

Intelligent Information Access to Linked Data

Weaving the Cultural Heritage Web

Inaugural-Dissertation
zur Erlangung des Doktorgrades
der Philosophischen Fakultät
der Universität zu Köln
im Fach Informationsverarbeitung

vorgelegt von
Robert Kummer
aus Frechen

Köln, 30. April 2012

Datum der Disputation am 11. Juli 2012

Erster Referent: Prof. Dr. Manfred Thaller

Zweiter Referent: Prof. Dr. Reinhard Förtsch

Abstract

The subject of the dissertation is an information alignment experiment of two cultural heritage information systems (ALAP): The Perseus Digital Library and Arachne. In modern societies, information integration is gaining importance for many tasks such as business decision making or even catastrophe management. It is beyond doubt that the information available in digital form can offer users new ways of interaction. Also, in the humanities and cultural heritage communities, more and more information is being published online. But in many situations the way that information has been made publicly available is disruptive to the research process due to its heterogeneity and distribution. Therefore integrated information will be a key factor to pursue successful research, and the need for information alignment is widely recognized.

ALAP is an attempt to integrate information from Perseus and Arachne, not only on a schema level, but to also perform entity resolution. To that end, technical peculiarities and philosophical implications of the concepts of identity and co-reference are discussed. Multiple approaches to information integration and entity resolution are discussed and evaluated. The methodology that is used to implement ALAP is mainly rooted in the fields of information retrieval and knowledge discovery.

First, an exploratory analysis was performed on both information systems to get a first impression of the data. After that, (semi-)structured information from both systems was extracted and normalized. Then, a clustering algorithm was used to reduce the number of needed entity comparisons. Finally, a thorough matching was performed on the different clusters. ALAP helped with identifying challenges and highlighted the opportunities that arise during the attempt to align cultural heritage information systems.

Contents

Preface	5
1 Introduction	6
1.1 Research Context	6
1.2 Scientific Interest	9
1.3 Approach Chosen	10
1.4 Thesis Contribution	12
1.5 Thesis Outline	13
2 Interoperability in the Humanities	15
2.1 Aspects of Interoperability	15
2.1.1 Views on Interoperability	16
2.1.2 Interoperability Research	17
2.2 Interoperability and Digital Humanities	20
2.3 An Example Scenario	22
2.3.1 Historical Sources and Archaeological Finds	23
2.3.2 Towards Use Cases and Applications	28
2.3.3 Possible Requirements	30
2.4 Interoperability and Formality	32
2.5 Related Projects and Organizations	34
3 Information Integration	38
3.1 Data, Information and Knowledge	38
3.2 Distribution and Heterogeneity	41
3.3 Integration of Entity Descriptions	43
3.3.1 Semantic Heterogeneity	43
3.3.2 Mapping and Matching	44
3.3.3 Semantic Information Integration	47
3.4 Integration Architecture and Infrastructure	50
4 Semantic Web Research	54
4.1 Fundamental Concepts	57
4.1.1 Modeling Knowledge	59
4.1.2 Calculating with Information	62
4.2 Semantic Web Information Integration	65
4.3 Recent Developments and Linked Data	68
4.4 Flagship Projects	71
4.5 Semantic Web Challenges	73
4.6 Knowledge Representation Alternatives	75

5	The CIDOC CRM	79
5.1	General Structure of the CRM	79
5.2	Information Integration and the CRM	81
5.3	Implementation Considerations and Applications	84
6	Entity Resolution as Knowledge Discovery	87
6.1	Information Retrieval	89
6.1.1	Information Extraction	90
6.1.2	The TF-IDF Model	92
6.1.3	Latent Semantic Analysis	95
6.2	Data Mining	98
6.2.1	Exploratory Data Mining	99
6.2.2	Prediction of Coreferences as Classification	102
6.2.3	Cluster Analysis of Vast Data Sets	107
7	A Theory of Similarity and Identity	111
7.1	Philosophical Implications	111
7.2	Methodological Considerations	117
7.3	(String) Distance Metrics	119
7.4	Automatizing the Decision Process	124
8	Towards an Entity Resolution Framework	129
8.1	A Real-World Example	130
8.2	A Theory of Entity Resolution	135
8.3	Entity Resolution Frameworks	138
8.3.1	Blocking and Resolving Entities	142
8.3.2	Matching Approaches	145
8.4	Evaluating the Success of Entity Resolution	151
8.5	Complementary Approaches	155
8.5.1	Controlled Vocabularies	156
8.5.2	Other Resources	158
9	An Alignment Experiment	161
9.1	Experiment Design and Software Development	162
9.2	Exploratory Analysis of Information Sources	170
9.2.1	Information Sources Overview	170
9.2.2	Initial Training Data Generation	173
9.2.3	Evaluating Extraction Success	176
9.2.4	Data Extraction, Cleaning and Normalization	178
9.3	Discussion of Considered Entity Description Aspects	181
9.3.1	Bibliographic Information	181

9.3.2	Collection and Depository Information	184
9.3.3	Findspot Information	187
9.3.4	Accession Numbers	189
9.3.5	Object Dimensions	191
9.3.6	Short Entity Description	194
9.4	Entity Resolution and Management	197
9.4.1	Partitioning of Datasets	197
9.4.2	Comparison of Entity Descriptions	199
9.4.3	Deciding on Matches and Non-Matches	202
9.4.4	Visual Representation and User Interaction	206
9.5	Discussion and Future Development	211
10	Beyond Entity Resolution	216
11	Summary and Outlook	221

Preface

I would like to express my gratitude to my advisor, Prof. Dr. Manfred Thaller, for his guidance and support during my dissertation project. I would like to thank Prof. Dr. Gregory Crane and Prof. Dr. Reinhard Förtsch for providing me with excellent research conditions and rendering possible an intensive collaboration with the Perseus Digital Library. I would also like to thank Prof. Dr. Ortwin Dally for assuring the support of the German Archaeological Institute.

Köln, September 2013

1 Introduction

1.1 Research Context

Bringing together information from disparate but related information resources for joint analysis is recognized as a central component of business decision making. By integrating data that is structured differently from different business areas, useful information is available to facilitate complex decision making. In a number of cases, additional summaries have been compiled from the information that has been joined in such data warehouses. And techniques that go beyond compiling simple summaries may also be applied to discover hidden connections and relationships among data objects. These techniques have been researched and elaborated for a long time under the notion “knowledge discovery”.

The methods and techniques elaborated in the fields of data warehousing and knowledge discovery may also be beneficial for processing information for the humanities and cultural heritage. In particular, the domain of digital cultural heritage has seen worldwide endeavors towards integrating cultural heritage data of different information systems.¹ A prominent metaphor used is the digital library, which electronically provides collections that are accessible and processable not only by humans but also by computers. As opposed to traditional libraries, they allow for location independent access to primary and secondary sources. Simultaneously, computers are able to exploit the same data to create added value by methods of processing like statistical analysis.² The generated information should be accessible along with the original resources, for example, by producing explicit relations between existing information objects.

A number of projects have tested the feasibility of models that are more or less standardized for sharing information in the humanities and cultural heritage [10]. Standards for electronically representing and sharing cultural heritage data have been discussed and published for a long time. For example, the Open Archives Initiative has developed the Protocol for Metadata Harvesting [81] for electronically distributing information in a controlled manner. And the International Committee on Documentation (CIDOC) published the CIDOC Conceptual Reference Model, a standard that facilitates the representation of cultural heritage data for processing and transmission [71]. The CIDOC Conceptual Reference Model mediates different

¹See for example the TextGrid project [116] as a larger German effort, the Europeana project [101] for European cultural heritage material and the reflections of the American Council of Learned Societies [4].

²The title of this publication refers to a definition by Chen, Li and Xuan [54]: “[...] Intelligent Information Access (IIA) refers to technologies that makes[sic!] use of human knowledge or human-like intelligence to provide effective and efficient access to large, distributed, heterogeneous and multilingual (and at this time mainly text-based) information resources and to satisfy users’ information needs.”

representations of cultural heritage information by imposing a certain structure for information organization. However, there is still no automated means for finding relations between information objects that are represented in a standardized way.

In this context, two parties from classical philology and archaeology have begun to study the peculiarities of information integration for material cultural heritage objects: The Perseus Project [68, 67] and Arachne [108]. The Perseus Project is a digital library currently hosted at Tufts University that provides humanities resources in digital form. It focuses on classical philology, but also considers early modern and even contemporary material. Arachne is the central database for archaeological objects of the German Archaeological Institute (DAI) [75] and the Cologne Digital Archaeology Lab (CodArchLab) [109] at the University of Cologne. DAI and CodArchLab joined their efforts in developing Arachne as a free tool for archaeological internet research. The results of this project have been documented as a master's thesis accepted by the University of Cologne [163]. These efforts are currently being advanced by different endeavors.³

Information systems that have been crafted to support scientists in the humanities and cultural heritage usually describe material and immaterial entities of that domain. These entities bear a certain set of features (i.e. the information that a sculpture is made of marble) and have relations to each other (i.e. the information that it has been found at a certain site). However, not all relevant features and not all significant relations have been documented for obvious reasons and thus do not become part of the entity description. But the methods and techniques of knowledge discovery referred to above could help by automatically enriching represented information with additional descriptions of features and relations.

In many situations, entity descriptions that refer to the same entity are represented in more than one information system and different aspects of the entity are described in each system. Information systems in the humanities and cultural heritage area are built with individual usage scenarios in mind, resulting in a multitude of scopes and perspectives. By discovering entity descriptions in different information systems that refer to the same entity, information about particular entities can be brought together. It turns out that the methodology of knowledge discovery and data mining provides the central toolset for aligning entities in the humanities.

The activities of mining and documenting links between “real-world” entities as well as finding relations between descriptions of entities (i.e. coreference) seem to afford further research opportunities. Thus, an information alignment experiment is elaborated and documented in the course of the following chapters, which focus

³Among others as part of a collaboration with the project CLAROS [17] hosted at the Beazley Archive in Oxford. The project Hellespont [9] focuses more on information about material cultural heritage that is encoded as text.

on a central aspect of information integration: aligning descriptions of entities that are represented in different information systems (Arachne and Perseus). Research in the field of cultural heritage often depends on describing complex settings that material objects are embedded in. And this requirement may result in complex entity descriptions. Since Arachne and Perseus use different national languages for entity descriptions, those knowledge discovery techniques that are robust and effective in this situation should be preferred.

This alignment experiment is implemented by using data from Arachne and Perseus with two major topics in mind: methodology and data. It aims to carry out the work that started as the above-mentioned collaboration, and it also aims to place clear emphasis on aligning entity descriptions. On the one hand, the methodology that is helpful and necessary for aligning cultural heritage information is identified and applied to real information. On the other hand, the alignment performance achieved with this particular kind of information is assessed. The results of this experiment will be documented and critically discussed to derive recommendations for future research in this area.

To put it in a nutshell, the humanities and the cultural heritage community in particular have recognized the importance of information integration in their domain. Aligning information about cultural heritage entities is vital for many subsequent forms of processing that generate significant value for scientists. But at the same time, information integration has been described as being extremely difficult and laborious because of the involved heterogeneity and complexity. Various forms of information organization and representation are often used by different information systems. In such situations, information integration projects should follow well-defined objectives that are developed from a user perspective.

The information alignment experiment for Arachne and Perseus (hereinafter ALAP) is designed to foster the understanding of this dilemma. Significant steps of the implemented workflow will be discussed in more detail by analyzing and interpreting the data that has been produced. It would be beyond the scope of this experiment to strive for holistic information integration and to make exclusive use of cutting-edge methodology. Therefore, a significant partition of the data provided by the information systems has been selected to study the alignment workflow. And the methods chosen are those that could be implemented with reasonable effort.

The discussion of these methods and techniques also comprises a reflection about ways of enhancing the entity resolution quality in the future by either using information in a different way or replacing particular techniques. Often, these techniques require a deeper understanding of the information to be aligned and need to be parameterized by optimization techniques. However, a considerable amount of entity descriptions has already been aligned by applying the techniques that

will be described in the following chapters. Opportunities for further research are derived and documented by analyzing the shortcomings of the current framework. In summary, considerable potential that is still unused could be unleashed by introducing the described methodology to the field of cultural heritage information integration.

1.2 Scientific Interest

An intuitive and reasonable approach to information integration in the field of cultural heritage is to make extensive use of resources that provide background knowledge. These comprise for example controlled and structured vocabularies that could help with resolving different names for one thing to a common identifier. Exploiting background knowledge can be extremely helpful if highly heterogeneous information needs to be aligned, in particular in international environments. But these vocabularies also have their drawbacks. They are expensive to compile and difficult to maintain. Therefore, it is difficult for information integration projects to find vocabularies that are adequate for the alignment of information and, due to costs, they are often unable to be compiled in the first place.

While these vocabularies will certainly be inevitable for information integration it is worth investigating alternatives. These alternatives should compensate for the mentioned weaknesses by making use of and contributing to different forms of background knowledge. Traditionally, structured vocabularies have been compiled in a top-down approach by dedicated authorities following a well-defined process. The entity resolution framework for Perseus and Arachne has served, among other things, to reflect on the various ways to complement traditional approaches with bottom-up methods.

Halpin, Robu and Shepherd [208] have observed that coherent categorization schemes can emerge from the unsupervised tagging activities of users. But archaeological information systems that are controlled by established institutions can also be seen as a valuable source for deriving organized forms of background knowledge. It would be beneficial if the information extracted from these information systems could be combined and used to generate background knowledge that is either implicit or explicit. Thus, ALAP is based on the hypothesis that entity resolution can be significantly improved by generating background knowledge from information that has already been aligned, to a certain extent, aligned.

Prior efforts to integrate information in Arachne and Perseus reveal the need to establish a shared data model and schema. The CIDOC CRM in combination with Semantic Web concepts has been chosen as a foundation for making internal information available to the public. At the same time, it has been recognized that aligning entity descriptions that are organized by these means should be a central concern of future research. The shared and structured vocabularies that

are related to these efforts, but which are still missing, could play an important role in future entity resolution frameworks.

Thus, the problem with structured vocabularies can be addressed by linking the content from different cultural heritage information systems in a meaningful way. These vocabularies are relevant both on the schema and instance level, but they are resource intensive to build and maintain. Therefore, ALAP strives to discover significant information within the data so that the factors with a strong positive or negative influence on this challenge can be determined. Immediately related to these efforts is whether the information that can be found and utilized within Arachne and Perseus is enough to significantly support future integration projects. This information could be used to bootstrap implicit and explicit forms of background knowledge, and to exploit this knowledge for resolving additional entities at the same time.

1.3 Approach Chosen

The predominant methodological approach that has been chosen to align the information in Arachne and Perseus stems from software development. The implementation of the experiment begins with reflecting on what interoperability could mean in an environment dealing with data from archaeology and classics. Thus, a user scenario has been elaborated to address vital requirements of information systems that strive to support the work of scientists. Besides motivating information integration itself, the scenario also defines the exact scope of the alignment framework to be implemented. To that end, software libraries are used or implemented to help with establishing vital components. These components are combined in a way that allows for effective and efficient entity resolution. The entity resolution process itself is developed in an iterative manner by interpreting the results and adjusting the implementation decisions and component configurations.

In the course of designing this entity resolution framework for Arachne and Perseus, different methodological approaches are considered. The main concepts and methods that support the above-mentioned tasks will be identified, discussed and documented in the course of the following sections. These comprise research that is rooted in different communities like database information integration, the Semantic Web / CIDOC CRM and knowledge discovery. Essentially, these focus on but are not restricted to harmonizing the way that entity descriptions are organized by data models and schematic structures. Since these challenges are currently being addressed by other projects like the Berlin Sculpture Network [8], they are not considered as part of ALAP.

The task of resolving entity descriptions is implemented by drawing from the paradigms of knowledge discovery and data mining. To that end, the workflow begins by extracting data from Arachne and Perseus, and it ends by aligning infor-

mation as well as by making generated background knowledge accessible for later processing. Preliminary background knowledge is generated in the form of trained machine learning models, semantic spaces and structured vocabularies. This information is then exploited by the alignment framework itself, and should be made available for other projects that deal with information retrieval and knowledge discovery in the future.

Once entity descriptions that refer to the same entity have been identified, they need to be aligned by some mechanism so that they are explicitly linked. Research that is currently being pursued in the Semantic Web community has suggested using Uniform Resource Identifiers to explicitly express sameness. Large projects have been studying ways of reducing the number of identifiers that refer to the same material or immaterial entity [44]. However, it seems that the mechanisms that have been originally introduced by the Semantic Web community are rather rigid and would lead to misleading or false representations. A number of philosophical implications that help with understanding this situation will be discussed and possible approaches will be suggested.

However, current developments indicate that the application of Semantic Web concepts in combination with the aforementioned CIDOC CRM is gaining momentum. Notable projects that strive to align information in the field of cultural heritage and archaeology in particular have been exploring Semantic Web concepts and / or the CIDOC CRM [8, 9, 15]. Therefore, the role of Semantic Web research for ALAP will be discussed, and its concepts will be evaluated. The concepts will be particularly interesting for ingesting information from different sources and sharing generated alignment information as well as explicit background knowledge.

Both Arachne and Perseus strive to make their information available according to the CIDOC CRM by making use of Semantic Web concepts. But for ALAP, information had to be extracted directly from each information system because the required endpoints were not available at the time of implementation. Additionally, the used techniques rely on information that is represented in a rather granular form. This kind of information could be extracted in a straightforward manner for the information objects that have been considered for ALAP. However, it will be possible to adjust the information extraction components with reasonable effort so that other forms of serialization can be consumed.

In summary, the overarching method used is a variation of a case study. A framework for aligning cultural heritage information with a strong focus on entity resolution is iteratively designed and implemented. Relevant methodology is identified, discussed and implemented as a kind of knowledge discovery workflow. Significant results generated at different steps of this workflow are analyzed and interpreted with the aim of enhancing effectiveness and efficiency. Opportunities and challenges are recognized at each step and for the overall framework. These

insights are accompanied by recommendations for future research.

1.4 Thesis Contribution

It has been argued that information alignment is vital for cultural heritage research. Information alignment provides advanced methods for accessing cultural heritage material and empowers scientists to tackle new and innovative questions. The main motivation for designing and planning the mentioned alignment experiment is to foster the understanding of necessary requirements, arising challenges and research opportunities. Both the identified methodology and the primary (alignment) and secondary (background) information that is generated will be beneficial for the humanities. In order to tackle this main scientific objective, a number of secondary topics must be addressed.

Archaeology and classical studies are domains that seem to be too small for developing and hosting huge IT infrastructures for themselves. Thus, ALAP is designed to not extensively rely on an external service infrastructure. Information is represented in a way that can easily be transmitted and reused in different contexts. This should enable projects to manage the needed background information as part of established infrastructure components like vocabulary services and gazetteers.

Information integration is resource intensive and, therefore, a rather expensive endeavor. Depending on the specifics of the underlying data, the amount of processing steps that can be (partially) automatized is different for each project. Therefore, ALAP should also help to discover opportunities for automatization and to minimize the need for allocating expensive resources to matching tasks. The concepts that need to be learned and the technology applied requires a certain amount of expertise. A compromise has been found between effective and efficient methodology and models that can be understood and interpreted with reasonable effort.

In order to maintain an overview, complex software systems are crafted according to certain architectural decisions. Designs that turn out to be helpful in different contexts qualify for being implemented in the form of reusable frameworks. The architectural design of ALAP uses foundational components of established entity resolution architectures. The way of combining these components and the flow of information between these is studied, and the results are documented in a way that should foster the implementation of entity resolution frameworks for cultural heritage in general.

Information integration projects need to address questions of existing infrastructure to communicate with information sources and to access background knowledge. But using and establishing complex and flexible infrastructures requires time and effort. Additionally, distributed infrastructures introduce additional risks like

vital components becoming unavailable as well as issues with latency and performance. These issues are discussed, and it is concluded that information integration projects should carefully assess the risks and opportunities of complex infrastructures. In the following chapters, needed infrastructural components will be identified, and the risks and opportunities of distributed infrastructures will be discussed.

Fensel [105] observes that ontologies in particular and the Semantic Web in general have been proposed as a “silver bullet” for information integration. It has been mentioned above that the concepts revolving around Semantic Web research are taking root in humanities information science and cultural heritage information integration. Therefore, the methodology and techniques that have been elaborated in the Semantic Web community will be examined more closely. In particular, the suitability of single concepts both for representing and processing data for information integration is discussed and assessed. The aim is to determine whether concepts contribute to the effectiveness and efficiency of the entity resolution process.

The methodology used in ALAP originates in different disciplines. Additionally, the architectures and infrastructures necessary for effective and efficient information alignment are rather diverse. Therefore, the principal aim of aligning information for Arachne and Perseus fans out into the topics that have been addressed above. By interpreting the results of ALAP, a better understanding of methodology, architecture and data should be fostered. Opportunities for enhancements and strands of future research will be highlighted for each of the above-mentioned areas.

1.5 Thesis Outline

Deriving functional requirements, surveying state-of-the-art methodology, implementation and documentation of ALAP are treated separately in the course of the argumentation. Due to the interdisciplinary nature of ALAP, many cross-references exist between the parts. Therefore, it is helpful to describe the overall structure of the chosen approach and to explain how the parts are related. Each part focuses on one overarching topic and contains several chapters that elaborate on the various aspects concerning this topic.

The first part (chapters two, three, four and five) deals with questions of interoperability and the role of information integration in general. It begins with an examination from a birds-eye perspective, motivates information integration and introduces important strands of research and foundational methodology. To that end, aspects of interoperability have been analyzed and functional requirements for information integration have been derived by elaborating a user scenario. Additionally, different approaches for establishing interoperability by implementing

an infrastructure that supports information integration will be considered.

The second chapter starts with an overview of interoperability challenges related to cultural heritage information systems by elaborating and discussing a user scenario. State-of-the-art approaches in the field of information integration are discussed in the third chapter with respect to their suitability for ALAP. Concepts that have been developed in the field of Semantic Web research are described in the fourth chapter, because their applicability to information integration has recently gained attention. The CIDOC CRM, which is related to the concepts of the Semantic Web and focused on cultural heritage information, is introduced in the fifth chapter.

The second part (chapter six, seven and eight) focuses on a central aspect of information integration: entity resolution. Although entity resolution is usually seen as a problem of data cleaning, it seems that dealing with it from a knowledge discovery perspective is helpful. Different ways of designing entity resolution frameworks are discussed with respect to their suitability for the interoperability of Arachne and Perseus.

Therefore, Chapter six introduces entity resolution as a problem of knowledge discovery and discusses appropriate methodology from data mining and machine learning that has been considered for ALAP. Philosophical implications related to the notions of similarity and identity are elaborated in chapter seven in so far as they result in additional requirements for the entity resolution framework. On the foundation of traditional approaches for entity resolution, the needed components and the way that they should interact is described in chapter eight.

The third part (chapters nine and ten) deals with the actual implementation of the information alignment experiment. To that end, software components are implemented and combined, making the extraction and alignment of information from Arachne and Perseus possible. The preliminary design of the framework is guided by an exploratory analysis of both information systems. Next, both the components and the way they are combined is iteratively elaborated in an analysis and interpretation of the generated results.

Chapter nine documents important aspects and significant intermediate steps of this development process. Those aspects of the described entities that have been intuitively selected for driving entity resolution are analyzed in greater detail. Both the maturity of the entity resolution approach that has been reached and recommendations for future research are addressed. Chapter 10 discusses the benefits of the generated alignment information and background knowledge as well as pointing out future research opportunities.

2 Interoperability in the Humanities

In many situations, collaborative work involving multiple entities is an approach that often proves to be more successful than the efforts of single monolithic entities. These entities comprise people, systems, organizations and so on. Collaborative architectures have a higher probability of producing high value and quality content. If single entities are to collaborate on a certain goal, a number of preconditions must be met. For example, interoperable infrastructures are necessary for telecommunication, software development and the medial industry. Networked infrastructures are also becoming increasingly relevant for research in the humanities. In addition to these interoperable infrastructures, the question of interoperability must be addressed. Over the years, different methods and techniques have been proposed to develop systems that are syntactically and semantically interoperable. The relevant aspects of interoperability are elaborated in this section.

2.1 Aspects of Interoperability

The term *interoperability* usually describes certain features of software systems in software development contexts. It refers to the ability of these systems to collaborate with other systems to work towards a defined goal. Systems that interoperate benefit from shared resources, which allow for a better division of labor. The term is also used in a broader sense to convey other influencing aspects, such as the social and organizational factors that form the environment of these systems. Thus, the non-technical entities that become part of a system infrastructure are considered. Two types of interoperability are distinguished here: syntactic and semantic. Although even more aspects could be distinguished, these two are central and the most difficult to establish.

Syntactic interoperability deals with the ways that systems can communicate to exchange data. Different hurdles must be overcome if syntactic interoperability is to be established. A certain network infrastructure that allows for the communication of different connected systems needs to be in place. Several technical obstacles, such as the creation of suitable application programming interfaces, must be resolved. An appropriate way to encode characters, which is usually the Unicode standard, must be agreed on. Also, the data structures that organize the data exchanged should be defined. If two systems are capable of communicating and exchanging data, they have established syntactic interoperability. But being able to bring data together at one place does not guarantee that the data can be processed according to its meaning. This comprises topics studied under the notion of semantic interoperability.

Syntactic interoperability is a precondition for establishing semantic interoperability. It is the information system's ability to joint processing information

according to its intended meaning. Thus, each information system needs to be in a position to properly interpret the information. This is important to enrich data and data structures with explicit meaning, making the data's results of reasoning predictable. Thus, semantic interoperability has been established if multiple systems are capable of automatically interpreting the data and processing it in a meaningful way. To achieve semantic interoperability, all parties need to commit to a set of shared semantics. Semantic Web technology, which is dealt with in more detail later, strives to establish semantic interoperability beyond the defined set of systems.

Syntactic and semantic interoperability seem to address how different systems can jointly achieve helpful results in a satisfactory manner. The following sections reflect on different aspects of interoperability with respect to its relevance for research in the humanities and in the cultural heritage domain in particular. This includes looking at the different stakeholders and the historical perspective. Additionally, a user scenario is elaborated to derive functional requirements for a system that integrates information from different cultural heritage information systems.

2.1.1 Views on Interoperability

It has already been emphasized that interoperability has many facets and opens up a complex problem domain. Not all of these will be relevant for the entity alignment experiment that is described later. However, to provide an overview, some helpful aspects are discussed in this section. Different views on interoperability are also presented to illuminate the subject from different perspectives. For example, Gradmann [120] enumerates a number of concepts that are relevant for analyzing interoperable architectures and infrastructures for digital libraries. The following overview introduces a number of concepts and stakeholders that are relevant for the user scenario. It does not claim to be exhaustive, but seeks to foster an awareness of the complexity.

Entities of almost any organization need to inter-operate in some way or the other. For example, it is a common practice for organizations to inform fiscal authorities about their earnings and losses so that they can be taxed. To that end, these entities need to exchange messages that are governed by the law. Organizations can also influence the emergence of interoperability. For example, funding organizations can equip cultural heritage entities with additional resources to establish an infrastructure that facilitates interoperability.

End users may be either people or systems, and are the main benefactors of interoperability. The means that establish interoperability should be transparent. In many cases, interoperability enables end users to achieve things that have not been achievable before, and, in the least, end users should perceive the benefits

of interoperability in a reduction of workload. Usually, the content of exchanged messages is formulated in a way that can be understood by all participants. It will differ in each discipline and national languages are problematic if messages need to be understood in an international environment. The analysis of knowledge in different disciplines is very important for establishing interoperability. In addition, in many areas, content is subject to copyright laws that need to be considered.

Another aspect of interoperability is the technological domain, which is manifold. Technical infrastructures are present in almost all larger organization to allow for interoperability. For example, communication processes in organizations and between organizations are frequently supported by information technology. Software components run on hardware that is connected by a networked infrastructure. These components usually interact with each other so that distributed computing can take place. Different functional requirements can be implemented on different hardware and be physically and logically distributed. Appropriate calling conventions must be in place to make use of this distributed functionality and to retrieve results.

The previous discussion exhibits a large number of aspects that are relevant for establishing interoperability. Large projects that do not consider these aspects risk serious architectural and infrastructural deficits. However, smaller information alignment projects should not aim to create holistic interoperability infrastructures. Rather, they should focus on selected technical aspects of an interoperability framework and restrict themselves to formulating desiderata for other aspects. The implemented technical means would benefit from background knowledge if it were in place. Background knowledge thus needs to be generated as well as maintained.

Many factors that cannot be clearly separated influence the problem area of interoperability research. At every stage of organizational information processing, humans are heavily involved in the communication process. They take part in exchanging messages, interpreting messages and deriving new and useful information to get a specific task done. In addition, recent digitization projects publish more information online than humans can process. Nevertheless, it is useful information that is more valuable if it is semantically integrated and interpreted. Thus, an alignment experiment is implemented to help identify the fundamental challenges and to reflect on ways to approach these problems.

2.1.2 Interoperability Research

How to make organizational knowledge accessible to different stakeholders is a pressing problem and has stimulated research that is strongly related to interoperability. The following paragraphs focus on research in the area of hypermedia and database systems. Work in these areas is driven by the idea of using technical means that support and extend the mental capabilities of humans. The idea

of hypermedia can be seen in the vicinity of interoperability research because it emphasizes the meaning of links between pieces of information that form a knowledge base. And database systems make large amounts of information accessible for querying and further processing. Recent developments attempting to integrate the functionality of both hypermedia and database systems will be illustrated in this section.

In 1945, Bush [48] illustrated his thoughts on how a machine should be constructed to help humans think, i.e., the MEMEX (Memory Extender). He imagined a machine made to support human memory and associative thinking. A researcher would be able to store his personal information with annotations and to create links as associative trails. With the help of this mechanism, researchers can encode thought processes by generating linear sequences of information. In addition, a potential MEMEX supports exporting information content from one machine and reusing it in other machines. The idea of the MEMEX heavily influenced future research and predicted the development of hypermedia systems.

In 1960 Ted Nelson began working on a project called Xanadu to research Hypertext concepts.[186] Xanadu is conceived of as a de-central storage system for interlinked documents, where each document has its own unique identifier that is independent of its physical place. Additionally, references to pieces of information are very granular and cover single characters. Although Xanadu has never been fully implemented, its main ideas have influenced the development of the World Wide Web as we use it today.

Douglas C. Engelbart was very impressed by the ideas published by Bush. In 1962, he defined his position on “augmenting human intellect”. He claimed he wanted to “increas[e] the capability of a man to approach a complex situation, to gain comprehension to suit his particular need, and to derive solutions to problems [99].” Solutions to problems usually require complex and complementary mental operations. Finding solutions involves understanding logical induction and deduction, but it also requires creativity and intuition. According to this paradigm, technology is useful if it provides fast and seamless access to information that can address problems as well as discover and elaborate new ones. In the context of this dissertation project, one step towards this objective would be to establish environments that facilitate the exchange and sharing of information among multiple cultural heritage information systems.

In 1989, Berners-Lee [20] proposed a new way of managing information at CERN by using the hypertext concept. One of the objectives of Berners-Lee has been to establish a straightforward way to exchange research results within the organization. Although the World Wide Web relies on the ideas that have been formulated within the hypertext community, it introduces a number of substantial changes. Perhaps the most fundamental deviation is that links do not need to

be bidirectional. Unidirectional links can be created by anyone without the need for communication with the authority of the link target. Along with the decision to open protocols and standards, this may be the most prominent reason for the success of the World Wide Web. However, the WWW has been established as a system that makes textual information accessible in an interlinked form, not machine actionable data.

Traditionally, the topic of machine actionable data has been examined under the notion database management system. These systems have been used to store structured data for querying and complex processing. Appropriate means for managing information are needed by almost every information system. Particularly, efficient ways to manage information need to be found to handle large amounts of data that does not fit into the main memory. In 1970, Codd [62] described the foundations of the relational data model, which can be considered today's industry standard. However, the need for information integration from distributed resources has also been identified in this area. Özsu[194] has systematized the area of distributed database system and reflected on the challenges that need to be overcome. In 1969, Fellegi and Sunter [104] came up with a theoretical framework for the problem of record linkage that is central for information integration.

Information that is publicly available on the World Wide Web usually lacks explicit structure for complex automatic processing. The highly structured information that is organized in database management systems is often not publicly available via the World Wide Web. Bergman [19] coined the notion of the “deep web” or “invisible web” for this situation. Some information hosted in databases is made available online by an intermediate layer that generates web pages on the fly. But a large amount of information is controlled by single applications that strongly determine the ways that others can use that data. Additionally, the syntax and semantics of data elements and the way they are organized is not public, making it difficult for third-party systems to interpret the data. In summary, one could argue that either the available data cannot be processed, or that the data which could be processed is not available. Since data is controlled by applications, information remains spread all over the world in a fragmented manner.

A development that is tightly related to the development of the World Wide Web is the implementation of markup languages. These have been developed to make the structure of documents explicit. A rather popular and successful markup language is XML, which was developed under the auspices of the World Wide Web Consortium (W3C). XML 1.0 was recommended by the W3C in 1998 and is the foundation for many other languages. Some of these languages, like HTML, focus on how a document is presented for displaying. Other languages that can be expressed by extensions of XML, like RDF, deal with the annotation of explicit semantics.

This form of semantic markup is one of the foundations of Semantic Web technology that will be dealt with in more detail later. In 2001, Berners Lee [27] published his thoughts on how the World Wide Web content should be organized in a way that is meaningful for computers. A suite of standards and techniques has been developed under the auspices of the W3C to deal with semantically storing and processing data. If many actors make their information available according to the concepts of Semantic Web research, then it should be integrated as well. And, it turns out that the same problems that the database community has been examining for years also arise in Semantic Web research. Thus, it would be helpful for communities, database research and web research to join efforts.

If the content of different information systems is brought together by syntactic and semantic integration, processing can be performed on more complete information and better results can be expected. A data element becomes information if it is associated with contextual data, and it becomes a more powerful research resource as it is added and linked to more data. In particular, Semantic Web research has adopted methods from research in the field of artificial intelligence to focus on how data is being processed. Formal semantics have been elaborated for RDF and for further concepts that build on RDF, which makes the deterministic processing of information possible. In the sense of Engelbart, the human intellect should be augmented by automatic inferencing on information.

To give a résumé, many areas of research try to tackle the problem of interoperability from different perspectives. Successful projects, like the World Wide Web, stimulate the creation of a comprehensive infrastructure that allows computer systems to interact. Reliance on open and standardized protocols like HTTP foster the development of ubiquitous information. But the whole only becomes more than the sum of its parts if structured information is integrated in a machine actionable way. The Web community would benefit from the years of past research that has been performed on the database community. ALAP makes use of methods and techniques that originate in the database and the Semantic Web communities.

2.2 Interoperability and Digital Humanities

Different aspects of interoperability have already been elaborated. The developments that have been discussed are a result of a pressing need to link and integrate information in particular communities. Architectures are planned, infrastructures are established and information systems are developed to approach this challenge. In the humanities, information science and cultural heritage in particular, more and more projects are publishing huge amounts of information. For example, large digitization projects are putting images of manuscripts and early prints along with metadata online, and archaeological information systems are producing data about archaeological finds. But how can this information be leveraged so that the spe-

cific needs of scientists in the humanities are met? Although this section is not intended to dive deep into the philosophy of science, it will develop basic ideas for how to structure and integrate information in the described context.

Humanities information science strives to support research in different disciplines by applying and developing information technology methods. To that end, information systems have been built to support certain aspects of research in the humanities. An information system that integrates data from different autonomous sources will support scientists with their research. If data is integrated in a form that can be understood by machines, new knowledge can be automatically derived. Therefore, information that has been syntactically and semantically integrated enables scientists to address more and different research questions.

It is certainly helpful for researchers to have cultural heritage information published online. Many situations, such as costly traveling, can be avoided if information is available on the internet. But this potential availability does not mean that everybody who needs the information can find it. Portals that list links try to attenuate this situation by providing low level finding aids for specialized information systems. The field of information retrieval tries to tackle this problem by providing search facilities that include the content itself (for example metasearch engines). Still, these engines make use of retrieval models that do not consider the deep semantics of information sources.

This is due to the fact that information is controlled by each information system independently. End users are forced to organize the relevant information into a proprietary analog fashion or in desktop information systems. Because of the homogeneity of bibliographic records, many problems have already been solved in this area. A number of standards like *bibtex* have been elaborated and are being actively used in different communities. But, due to heterogeneity, the situation is much worse for historical entities like archaeological objects or ancient buildings. A number of scientific questions cannot be optimally addressed without access to structured and integrated data.

Different disciplines in the humanities employ distinct methods and techniques in their research. Two well-known methods being used in a number of disciplines are hermeneutics and source criticism. These are particularly interesting to motivate interoperability because they rely heavily on information from various sources. Additionally, a high level of interaction between the human researcher and the information system is necessary to derive results.

In the field of literature studies, scholars must often prepare interpretations of literary works. One common method of interpretation is hermeneutics. A foundational concept of this interpretation method is the hermeneutic circle. It describes the process of understanding a text in iterative circles of mental processing. It also takes into consideration the cultural, historical and literary context. According to

the concept of hermeneutics, the meaning of a text can only be reconstructed if these additional contexts are included. Therefore, the text as an entity should be linked to further contextual information, which may be provided by additional information systems.

A recurring task for many scholars is to assess the authenticity and credibility of a historical source. The kinds of (historical) sources and the methods of criticism are explored in a number of auxiliary sciences of history. For example, research in diplomatics can support hypotheses put forward by medievalists. One way to criticize a source is to position it in a relationship with other sources or findings. The next section elaborates on this topic in the fields of ancient history and archaeology. One common user scenario is the historian who needs to verify a passage from an ancient text. He must collect information from a number of information sources that have not been explicitly related to each other, even if they are available on the Web.

Research in the humanities and cultural heritage certainly benefits from information that is available online. Additional information can be generated if information is structured and related to each other. This creates new ways to browse and query information on a more complete, high quality knowledge base. In turn, researchers are in a position to explore many more questions than they ever could in the past. However, structuring and linking information is an expensive endeavor. Entities that are the subject of research in the humanities tend to be very heterogeneous unlike information about customers or suppliers in economic contexts. Therefore, relevant scenarios and examples of past uses should be elaborated in order to bring about information integration.

Researchers in the humanities have been trained to find and exploit information that is relevant for their research problem. By explicitly linking relevant contextual knowledge to a piece of information, this research process can be further supported and improved. Historians or literary scholars may even find new or unexpected information that can be incorporated into their research. A lot of potential could be unleashed by integrating information from different sources in the humanities. However, this process should be goal-driven because the untargeted linking of data would certainly not create satisfactory results.

2.3 An Example Scenario

As mentioned, a user scenario should be developed to help understand the research interest of ALAP. Alexander [1] explores how user scenarios can support the development process and how they are employed in different contexts. These scenarios are frequently used as a tool in software development to facilitate a shared problem understanding by the different stakeholders (programmers, managers and so on). A user scenario is a narrative that describes how users interact with software

systems. Thereby, they are tools for reflecting on goals, expectations, motivations, actions and reactions that users may have, perform or encounter.

The methods and techniques explored in the fields of data mining and machine learning are central for the alignment of information. Herzog, Scheuren and Winkler [138] argue that for any data mining or analysis task, one should have a good understanding of how the discovered information is going to be used. And in the context of ALAP, the user scenario should help infer the requirements that form the basis for studying its feasibility. The user scenario helps to define the scope of the endeavor and is the foundation for a clear project definition.

2.3.1 Historical Sources and Archaeological Finds

The following paragraphs introduce a scenario that is set in the context of research in ancient history and archaeology. It emphasizes how information from different sources drives research on a certain topic. In this case, the scenario involves the development of ancient Pergamum. This case functions as a foundation for further discussion on the requirements and limits of information integration; it also serves as a good example of a possible use case, and helps future researchers envision appropriate architectures and infrastructures. In the following scenario, the main person is an imaginative historian who is working on the history of Pergamum.

A software system that is supposed to help this researcher should be able to gather as much information as possible on this specific research topic. A simple approach with full-text searching would probably not return all occurrences of this historical site because the name Pergamum is notoriously different in sources and editions. Additionally, multiple places could share the same name, another fact that is not dealt with by traditional full-text indexing and searching. Therefore, both precision and recall will not be very high. Precision is the number of search results that are relevant, and recall is the percentage of all relevant documents that are shown to the user. Information on ancient Pergamum can be various and come from various sources.

The Perseus Project is a digital library at Tufts University that assembles digital collections of humanities resources. The historian in our scenario uses Perseus to find references to Pergamum in ancient textual sources. After submitting a full-text query for the word “Pergamum” and browsing the search results, an interesting passage in a text by Strabo is found. Figure 1 shows the Perseus reading environment displaying this passage of the thirteenth book of Strabo’s Geography. The highlighted text is saying that Eumenes II, king of Pergamon, built up the city and that his successors added sacred buildings and libraries. The exact quotations is that “he built up the city and planted Nicephorium with a grove, and the other elder brother, from love of splendor, added sacred buildings and libraries and raised the settlement of Pergamon to what it now is.” Now, the historian

wants to look for further evidence that supports the credibility of this passage.

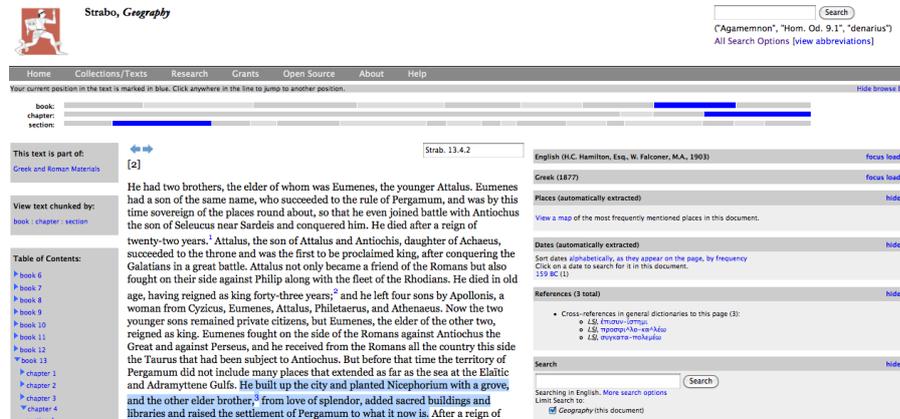


Figure 1: The Perseus reading environment displaying a text passage by Strabo that deals with the expansion of Pergamum (Screenshot taken from The Perseus Digital Library).

Since the historian has a strong archaeological background, he is aware of “Arachne”, the central database for archaeological objects at the German Archaeological Institute. Again, he submits a full-text query to search descriptions of archaeological entities for the word “Eumenes”. Then, he narrows down the search results by filtering only entities that are topographic sites. He realizes that several entity descriptions refer to a person with the name “Eumenes”, for example the main road of the arsenal. One entity that is associated with the short description “sogenannte Eumenische Stadtmauer”, which is the city wall of ancient Pergamum, attracts his attention. Since he wants to see more information, he clicks on the thumbnail to refer to the full entity description. Figure 2 shows how Arachne displays an entity description that refers to a topographical site. The historian can now browse through a set of information about the city wall and have a look at high resolution images.

Additionally, Arachne provides a tool that visualizes how an entity relates to other entities. This tool helps users obtain an overview of the different contexts that a material or immaterial entity is embedded in. Figure 3 shows how a node of a graph that represents a database record is situated in the middle of two concentric circles. While the inner circle shows database records stemming from different contexts that are immediate neighbors, the outer circle shows indirect neighbors. This visualization has been made possible by explicitly linking objects that are related within Arachne. By using this tool, historians can navigate a set of data that comes with rich structure, as opposed to merely navigating traditional, flat browsing paradigms.

8003191: sog. Eumenische Stadtmauer
Pergamon, Bergama

Vorhandene Bilder: 9
[alle Bilder anzeigen](#)

kein Eintrag D-DAI-ATH-Pergamon-0029

kein Eintrag D-DAI-ATH-Pergamon-0105

kein Eintrag D-DAI-ATH-Pergamon-0106

Informationen zur Topographie dieses Ortes

Lokalisierung:
Bergama, Antiker Ortsname: Pergamon, Türkei.
Ausdehnung: Mauerring von über 4 km Länge
antike Landschaft: Mysien
römische Provinz: Asia

Beschreibung:
Kategorie: Befestigung
freie Beschreibung: die an dem kyklopisch-polygonalen Mauerzug auf der Ostseite des Stadtberges ansetzende Mauer (Breite 2,2-2,7 m) zog in sägezahnartig gebrochener Strecke bis fast hinunter zum Keios-Fluss, um dann mit einem Eckturn nach Süden umzubiegen; in mehrfach geknickter Führung erreichte sie das Haupttor der Stadt (Eumenisches Tor); sie zog dann in westlicher Richtung auf den Selinofluss zu, um dessen Uferböschung dann nach Norden zu folgen; nach dem Verlassen des Flusslaufes wendet sich die Mauer nach Norden und dann nach Nordosten, um in Zick-Zack-Linie, unterbrochen von mehreren Türmen und kleineren Toren, die Arsenalspitze zu erreichen. Die traditionelle Zuschreibung dieses Mauerrings an Eumenes II. ist jüngst durch Keramikfunde einiger Sondagen bestätigt worden (Pirson a. O.).

Datierung:
Topographisches Objekt: hellenistisch, 1. Hälfte 2. Jh. v. Chr. (eumenisch).

Literatur:
M. Klinkott, Die Stadtmauern I. Die Byzantinischen Befestigungsanlagen von Pergamon, AvP XVI 1 (2001) 94-96;
W. Radt, Pergamon. Geschichte und Bauten einer antiken Metropole (Darmstadt 1999) 57-59;
A. Conze - O. Berjet - A. Philippon - C. Schuchhardt - F. Gräber, Stadt und Landschaft, AvP 1, 2 (Berlin 1913) 185-214;
W. Raack, IstMitt 54, 2004, 28 f.;
M. Klinkott, IstMitt 54, 2004, 147-159;
F. Pirson, AA 2007/2, 32-34;

Beschreibung der Datensatzanzeige

Auf der linken Seite sehen Sie detaillierte Informationen zum ausgewählten Datensatz.

Die Kopfzeile besteht aus der eindeutigen Seriennummer des Datensatzes in Arachne, gefolgt von einer Kurzbeschreibung. Darunter befinden sich die in der Datenbank vorhandenen Informationen zum Datensatz.

Falls Bilder zum Datensatz vorhanden sind erscheinen sie als Vorschaubilder, unterschrieben mit der jeweiligen Negativnummer, links neben den textuellen Informationen. Die Bilder können durch Anklicken des Vorschaubildes vergrößert werden. Mit einem Klick auf die Negativnummer gelangen Sie zur Ansicht des Bilddatensatzes, die zusätzliche Informationen zum Bild anzeigt.

In der grauen Navigationsleiste kann außerdem die Anzeige von Bezuehugnen des Datensatzes zu anderen Datensätzen in Arachne ausgewählt werden.

Figure 2: The Arachne single record view displaying topographic information about the city wall of Pergamum (Screenshot taken from Arachne).

8003191: sog. Eumenische Stadtmauer
Pergamon, Bergama

Vorhandene Bilder: 9
[alle Bilder anzeigen](#)

kein Eintrag D-DAI-ATH-Pergamon-0029

kein Eintrag D-DAI-ATH-Pergamon-0105

kein Eintrag D-DAI-ATH-Pergamon-0106

[Zum alten Kontextbrowser](#)

Wohnhaus...
abschnitt der sog...
Kurtine der phille...
Mauerabschnitt de...
Mauerabschnitt de...
Mauerabschnitt de...
sog. Philistalisc...
Mauerabschnitt de...
sog. Philistalisc...
Streckenabschnitt...
sog. Eumenische...
Agora Depot, ehem...
sog. Eumenische...
Ecksteigbinder...
Flügelfragment vo...
Fragment einer St...
18 Bauwerke
Relieffragment mit...
Fragment einer St...
Untere Agora
Museum Bergama

Beschreibung der Datensatzanzeige

Auf der linken Seite sehen Sie detaillierte Informationen zum ausgewählten Datensatz.

Die Kopfzeile besteht aus der eindeutigen Seriennummer des Datensatzes in Arachne, gefolgt von einer Kurzbeschreibung. Darunter befinden sich die in der Datenbank vorhandenen Informationen zum Datensatz.

Falls Bilder zum Datensatz vorhanden sind erscheinen sie als Vorschaubilder, unterschrieben mit der jeweiligen Negativnummer, links neben den textuellen Informationen. Die Bilder können durch Anklicken des Vorschaubildes vergrößert werden. Mit einem Klick auf die Negativnummer gelangen Sie zur Ansicht des Bilddatensatzes, die zusätzliche Informationen zum Bild anzeigt.

In der grauen Navigationsleiste kann außerdem die Anzeige von Bezuehugnen des Datensatzes zu anderen Datensätzen in Arachne ausgewählt werden.

Figure 3: The Arachne context browser displaying contextual information about the city wall of Pergamum (Screenshot taken from Arachne).

The description of entities in Arachne comprises information about bibliographic entities that refer to a specific entity (in this case, the topographic unit). Figure 2 shows that the most recent publication is by Pirson and appeared in the “Archäologischer Anzeiger”. After consulting “Zenon”, an online catalog that integrates several bibliographic databases, the historian realizes that this is a reference to a recent excavation report of Pergamum. More information about the archaeological excavation in Pergamum can be found in iDAI.field, a modular database for comprehensive documentation of field research projects. Under the auspices of the German Archaeological Institute, ongoing excavations in the region of the ancient Pergamum are documented in iDAI.field. Figure 4 shows a screenshot of iDAI.field, displaying information about archaeological findings near the southern gate of the city wall. Additionally, images are provided that have been taken from specific excavation sites.

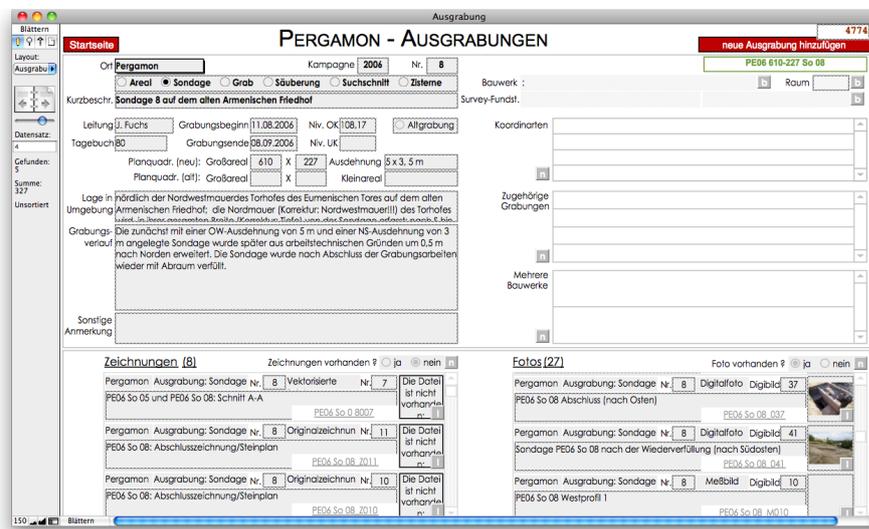


Figure 4: Documentation of a trenching at the city wall of Pergamon (Image taken from iDAI.field).

Figure 5 gives a glance at trenching number 8, which was performed in 2006 at the southern gate of the city wall of Pergamon. In a depth of 1.2 meters, the excavators discovered a pavement made of plates that flushed with the wall. The image also shows two pipelines that run parallel to the city wall. The undermost layer reveals debris of ceramic that were used to build the city wall of Pergamon. The ceramic fragments date back to hellenistic times.⁴ This fact supports the text passage of Strabo, where he claims that Eumenes II extended the settlement of

⁴Refer to [199] for a detailed account of the Pergamon excavations.

Pergamon. Construction activity related to the city wall is a strong indication the city itself was extended.



Figure 5: Image of a trenching at the city wall of Pergamon (Copyright by DAI Istanbul, Pergamon excavations).

The historian may be interested in information on Eumenes II from additional sources. For example, descriptions of material entities often contain information about inscriptions. A number of information systems that focus on inscriptions could provide additional information and should, therefore, be explicitly linked. Admittedly, the user scenario elaborated here is artificial, but it helps to improve our understanding of how information systems can support the work of researchers in the humanities. One can only imagine how powerful information systems could be if they provided links to additional information or integrated information from different sources. Additionally, the scenario emphasizes that this endeavor needs to be goal driven and not random.

Historians must collect information from different sources to obtain a complete (as possible) picture of historical activities. Luckily, they can rely on a growing number of projects that publish historical information online. These information sources can provide additional value if they start talking to each other and enable links and integration. In the course of the user scenario, the historian needs to deal with a number of different interaction metaphors and different national languages. By providing a single point of entry for information provision, the research quality

was significantly enhanced.

2.3.2 Towards Use Cases and Applications

The scenario emphasizes how integrated information can support the work of scientists in the humanities in a number of ways. Both complex applications and more concrete use cases can be derived from the scenario if the crucial points of user interaction are elaborated. However, a number of additional use cases could be established if integrated information was available. This section reflects on these applications and illustrates how they could fit into the daily work of a historian.

At several steps of the research process, the fictional historian had to rely on his knowledge to seek out the relevant information sources. After submitting full-text queries to different information systems, he had to sort out the relevant search results. State-of-the-art information retrieval systems allow users to put keywords in a single slot for submitting a query. The information retrieval model tries to produce content that seems to be relevant to the user. Obviously, it would be really helpful to end-users if these searching capabilities were applied to a collection of integrated information.

However, users still need to sort through the results to identify relevant pieces of information. This is due to the fact that information retrieval models cannot correctly interpret the semantics of documents. Recent research in the Semantic Web community is trying to tackle both problems: information integration and annotating rich semantics. This should not only allow for the formulation of more precise queries and higher precision and recall but also the automatic inference of additional information. Although this comes with the drawback of additional complexity, it is certainly an interesting development.

Both the user interfaces of Arachne and the Perseus reading environment exhibit functionality for browsing. Perseus uses these browsing features to convey additional information that is relevant to a specific passage of text, e.g., dictionary references or links to places that are mentioned in the text. Arachne uses a context browser to make the context of an archaeological entity explicit. However, only information that comes from within each system is presented to the user. An important step would be to make information from external systems visible.

Large amounts of data objects often feature useful information, but that information is neither explicit nor easily derived by manual inspection. Algorithms have therefore been elaborated in the field of data mining to automatically or semi-automatically analyze large amounts of information for meaningful structures or patterns. To that end, methods from a number of disciplines like artificial intelligence, machine learning and statistics are combined. As for searching and browsing, these techniques are improved upon if data is available in a comparable form, which increases the overall data for analysis.

Every user interface should present content to users in a way that they can understand and that is helpful. Graphical user interfaces are often divided into different parts that enable the display and navigation of content. Since the information that has to be conveyed to the user is different in each discipline and situation, there is usually a plethora of metaphors in use. Common user interfaces use simple graphical means to present data or information like text, pictures, lists, tables or more complex diagrams. A strain of research that is related to data mining, information visualization, examines ways to visually represent large amounts of data, so that users can deal with it.

For example, in the field of cultural heritage temporal and spatial coherences are often very important. Timelines can give a visual overview of how archaeological findings are distributed over time. Geographical distributions can be illustrated by putting a dot on a map for each archaeological object. Advanced forms of visual illustration and interaction that support reasoning and decision making have been discussed under the notion visual analytics. These techniques support thought and reasoning processes by exploiting information other than text. These visualizations often make implicit information explicit. Simoff compiled several contributions on how visualization and data mining relate to each other [227].

Another field has already been mentioned as being related to data mining: artificial intelligence. It strives to simulate the rational behavior of human beings, and is traditionally described as the ability to derive facts from existing knowledge. Artificial intelligence is promising insofar as it has been useful for deriving new information from existing knowledge bases. This can either be done by formal reasoning or by statistical processing of the data, thus enabling end users to benefit from additional relevant information. The tools that have been developed to process Semantic Web data seem to focus more on formal reasoning techniques. However, both ways of information generation will be evaluated for the aims of ALAP that will be described later.

As part of the research process, scientists collect and compile material that is relevant to their research problem. It should be possible to use this material as some kind of narrative that comprises a consecutive argumentation. Therefore, it is important to provide the means to organize information that has been discovered so that it can be used for further work. In this situation it is important for scientists to be able to manage information that has been drawn from different sources and to be able to make associations between the sources. Therefore, users want to be able to organize and annotate information that they have discovered in several information systems. They specifically want to create their own semantic links and group their information according to subject areas. At the end of this process, a way of seamlessly publishing the research results would be a very helpful feature.

A number of applications like searching and browsing can be directly derived

from the user scenario. But users also benefit from more advanced forms of knowledge discovery, information visualization and information management. Almost all applications produce better and more complete results if integrated information from different information sources is available. On the other hand, almost all mentioned techniques can be used to create collections of integrated information in the first place. In particular, data mining that makes use of machine learning techniques will be discussed in more detail later. It provides a helpful set of methods and techniques for information alignment.

Research in the Semantic Web community explicitly strives to deal with the problem of information integration. This is mainly achieved by adding explicit semantic structure to information elements and by applying automatic reasoning. Structured information usually allows for better exploiting semantics encoded in the information. The methods and techniques of information integration have been successfully applied to structured information over a long period of time within the database community. Thus, the following sections evaluate ways of applying traditional approaches to generate information that is also useful in the context of the Semantic Web.

2.3.3 Possible Requirements

The preceding considerations explicated a general idea of functional requirements for a system that integrates cultural heritage information. It is evident that information needs to be integrated from various and very heterogeneous information sources. Both structured and unstructured resources have to be considered to fulfill the information needs of users. Because of the autonomy of most information systems, a high amount of heterogeneity needs to be dealt with. In such situations, it is helpful to differentiate requirements that relate to technical and infrastructural peculiarities from the requirements that relate to content.

Many software projects are envisioned by single persons or groups that have a pressing need. The software is then developed to implement means to satisfy these needs. And usually, the situation is rather complex because different stakeholders articulate different needs. Alexander et al. [1] introduce a set of cognitive tools that help all participants of a software project to discover, document and implement these needs. Of course, for ALAP, this process is less complex. However, this section gives a short survey of requirements that will be partially implemented later.

In order to provide means for different information systems to exchange messages, a technical infrastructure needs to be established. But exchanging messages does not mean that these messages can be understood by each system. Providing a technical infrastructure only means that a number of information systems can rely on a working network connection and protocols for information exchange. To-

day, most physical machines are able to communicate via the TCP/IP mechanism, usually by making use of higher level protocols like HTTP.

In many situations it is useful to distribute the functionalities of a system to several connected resources. Different information systems, like Arachne and Perseus, have different focuses with regard to their content. Therefore, it is useful to manage the information in different systems and to use different technical means. This is one reason why systems that strive for information integration require complex architectures and infrastructures. Data needs to be transferred to a central component for further processing and may even be redistributed for reasons of efficiency. Different architectures and infrastructures for information integration will be elaborated later.

Different participants of a communication process need to be able to establish a shared understanding of the meaning of exchanged information. This is usually done by adhering to certain standards that implement a data model and schema for transmitting needed information. In contrast to the internal representation of information for information systems, these standards are often referred to as metadata exchange formats. Recently, a reference model has become popular for encoding and exchanging cultural heritage information, the CIDOC CRM. It has been implemented with means that have been elaborated in the field of knowledge representation and the Semantic Web community.

The elaborated scenario considers different types of information that is very heterogeneous. Besides a considerable amount of information about archaeological objects, the Perseus Project focuses on textual information. As stated, information integration is more effective if structured information is available. Thus, one of the first steps would be to extract relevant and structured information from different information systems. McCallum [176] provides a short overview of methods that can be used for information extraction from (semi-)structured data sources.

Of course, the activity of information integration is a fundamental requirement for integrated information systems. It is vital to link the information that has not yet been explicitly related. The extracted information not only needs to be matched and mapped on a schematic level but also on a data level. If the data is not properly integrated, most of the applications that have been described so far will generate unsatisfactory results. State-of-the-art methods and techniques of information integration will be elaborated later.

A couple of requirements and applications follow from the user scenario. Not all are relevant for the scientific interest of ALAP and others go beyond its scope. Thus, the topic needs to be further narrowed down to effectively and efficiently implement means for information alignment. Information that is considered for ALAP comprises parts of the material artifacts of Arachne and parts of the archaeological collection of the Perseus project. Thus, the information considered is

at least partially structured and not mined from full-text. The aim of ALAP is not to establish a comprehensive architecture and infrastructure for information integration. Furthermore, at least for the time being, information is extracted directly from the information systems without relying on exchange formats.

Thus, although a wide range of topics can be derived from the user scenario by envisioning applications, the subject needs to be further narrowed down. Functional requirements have been derived from these applications that are beyond the scope of this project. Therefore, the process of software development needs to focus on implementing means that are absolutely necessary. However, this minimal architecture should be implemented in a way that is modular and extendable.

2.4 Interoperability and Formality

The functional requirements needed to establish an interoperable system architecture have already been introduced. To establish a useful level of interoperability for end users, information must be structured and annotated with explicit semantics. This is important because internal data models and database schemata need to be semantically mapped onto external standards like metadata exchange formats. Structured and formalized information should also be accessible to automatic information manipulation by computers that go beyond full-text processing. At the same time, the structuring and formalizing of the information demands from software developers and end users to deal with more rigid formalisms that require additional overhead.

Everybody will at some time or other have entered informal information into a computer by using a word processor. Computers store that information and add certain internal structural information about paragraphs or page breaks. Additionally, it is possible to count the characters and words. But it is difficult to access additional structure and meaning that is implicitly contained in the text. Although the methods that have been explored in text mining are able to extract some structure and meaning from certain texts, this is a complex endeavor.

Besides this semi-formal way of making information accessible, there are everyday examples where more formality is required. If a user writes an email, there are certain fields that need to be filled in a strict way so that the email can reach its destination. Formal structures have also been used to create complex knowledge bases that can be used to perform automatic reasoning. For that purpose, medical data has been structured according to a formal syntax and semantics so that users can do useful things with it. By performing automated reasoning, which imitates human thought, these systems could come up with new and useful knowledge [112].

Shipman and Marshall [222] report how formalisms that have been embedded in computer systems can pose serious challenges to several stakeholders. To meet the criteria of information systems that work with strict formalisms, users need

to perform steps that are not inherently part of the task that they are trying to solve. These systems require users to break information up into meaningful chunks and then to characterize the structured content with names or keywords. Other systems may ask users to characterize information according to certain criteria or to specify how pieces of information are related to make implicit semantics explicit.

For example, the simple sentence “Plato is a student of Socrates .” needs to be chunked so that two individuals are explicitly related to each other: scholarOf(Socrates, Plato), beingScholarOf(Socrates)(Plato). Starting from this situation, Shipman and marshall state that the “[...] creators of systems that support intellectual work like design, writing, or organizing and interpreting information are particularly at risk of expecting too great a level of formality from their users.” Additionally, users often cannot express what they want or need to express because the terms and the ways that they can be composed are restricted, enforcing a structure that does not suit the needs of users is enforced.

Additionally, in most situations this requires more effort to enter formalized information into a system than free text. Sometimes this additional overhead can negate the usefulness of powerful means of information manipulation. And system developers also must provide the means to deal with this additional structure and meaning. This can drive up the costs for additional development and make the operation of such information systems ineffective. Therefore, a compromise needs to be found in most situations between the drawbacks and gained functionality.

Establishing a repository of integrated information from several cultural heritage information systems is the first step. And explicit syntax and semantics will allow for additional functionality, which can be conveyed to the end-user by developing appropriate user interfaces. These will not only allow for searching and browsing but also for submitting complex structured queries and new ways of navigation. But it is important to hide the complexity of the underlying data model and database schemata (i.e., which deals with explicit structure and semantics) from the users. This involves a lot of mediation between the back-end and the front-end, which results in additional costs for system development that increase with the degree of needed formality.

In many situations, establishing interoperability relies on introducing additional formality. This is important because only syntactical structures with explicit semantics can be related to each other so that joint processing becomes possible. Because the extraction of relevant information and structuring of it according to the needs of certain applications is resource intensive, a planned approach seems to be sensible. Here, a balance needs to be found in which the costs of introducing formality outweigh the benefits.

Creating well-structured content with defined syntax and semantics is resource intensive and can be problematic for end-users. However, structured content with

well-defined meaning is also the precondition for deep processing and advanced information manipulation, which helps with information integration. After having identified relevant aspects of entity descriptions in Arachne and Perseus, structured information was extracted from semi-structured database content where possible. This information was then used for matching and mapping – both of which involve user interaction. To this end, a rudimentary user interface was developed.

2.5 Related Projects and Organizations

The need for establishing interoperability is recognized in the digital humanities. A number of projects and organizations contribute to the aim of integrated information in the humanities and cultural heritage area. This section provides an incomplete overview of projects that strive for interoperability in the humanities and information integration in particular. It focuses on projects that have influenced the way that information alignment for Arachne and Perseus is pursued. The different concepts and standards developed in these projects are relevant to ALAP.

Major standardization efforts like the hardware architecture of computers or the foundational internet protocol TCP/IP rely on reference models. More specific models can be derived by reusing reference models to reduce implementation costs. But these reference model can also foster the interoperability of software systems. A reference model for the cultural heritage domain that has become rather popular is the CIDOC CRM (CIDOC Conceptual Reference Model), which is standard ISO 21127:2006 in 2006 [87]. A special interest group that comprises more than 50 member institutions that participate in the application and development of the CIDOC CRM has formed [88]. Since the CIDOC CRM has been used in combination with Semantic Web technology and is being used for exchanging cultural heritage information, it will be described in more detail later.

CLAROS (Classical Art Research Online Services) has used the CIDOC CRM for information integration in combination with Semantic Web technology [17]. The project strives to integrate content of several cultural heritage information systems to make it available to the public by a single browser. Additionally, image recognition algorithms have been developed so that new ways of accessing the contents can be provided in the future. A couple of ideas that have been considered for ALAP were developed during meetings and discussions with the project partners.

The CIDOC CRM has also been explored by the Berlin Sculpture Network. This is a project funded by the Federal Ministry of Education and Research [8]. The focus of this project is to reconstruct the original, ancient context of sculptures. This endeavor should form the basis to derive spatial, functional or substantial coherences of archaeological entities. A part of the project deals with

developing digital information of all sculptures, excavations and plaster casts in Arachne. Arachne provides an OAI-PMH interface, which publishes the data for a greater audience in CIDOC CRM. Thus, the project will unleash massive information in standardized form that can be used for information integration by others.

The Hellespont Project [9] is a joint effort of the German Archaeological Institute (DAI), the Cologne Digital Archaeology Lab (CoDArchLab) and the Perseus Digital Library. The project strives for aligning textual resources and object data from classical antiquity using the CIDOC CRM. As a starting point, Thucydides' Pentecontaetia - alongside extensive treebanking work - has been manually analyzed to annotate relevant entities, such as places and topographical entities, and people, groups and events. At the same time the project relies on the metadata mapping that is being created in the course of the aforementioned Berlin Sculpture Network. The combination of the created resources in the form of a virtual research environment should enrich research in this area.

The project STAR (Semantic Technologies for Archaeological Resources) has been funded by the Arts & Humanities Research Council since 2010. It applies knowledge-based technologies to the archaeology domain [247]. Together with English Heritage it has created controlled and structured vocabularies in combination with other means to help link digital archive databases. To that end, an extension of the CIDOC CRM has been elaborated, which helps to model information in the context of archaeological excavations. A research demonstrator was developed to demonstrate how information from several institutions can be integrated. Natural language processing is applied to unstructured textual information to extract key concepts.

ThoughtLab is a space within Europeana that highlights the various efforts of different project partners. It shows a number of topics that have been discussed as being vital to the establishment of interoperability. The ThoughtLab is also a search platform that collects information from European digital libraries. [100]. A semantic search engine for Europeana is being developed in an experiment as a future tool for accessing the content of Europeana. The user interface provides categorized auto-suggestions for searching and also groups search results into meaningful categories. Each object is annotated with additional structured vocabulary. Its relation to ALAP is its focus on metadata that is highly structured and semantically annotated.

The WissKI project (Wissenschaftliche Kommunikationsinfrastruktur) is funded by the German Research Foundation [260]. It brings together multiple project partners that strive for building an information system that allows researchers in memory institutions to “collect, store, manage and communicate knowledge.” To that end, the concept of a Wiki has been adopted so that scientists in these insti-

tutions can leverage their work with the system. The software in use is equipped with a semantic back-end that makes use of current Semantic Web technology. It makes use of controlled and structured vocabularies to organize and name content. The Erlangen CRM is an implementation of the CIDOC CRM using Semantic Web technology [192]. It aims at information integration and processing.

The TEI Consortium has published guidelines for digital text encoding and sharing [240]. Within the humanities, these guidelines are the standard for encoding printed works and linguistic information. Because texts often refer to persons (physical and legal), dates, events, places, objects, etc., there have been discussions about whether it would be helpful to annotate these items outside the text. This seems to be a good fit for the ongoing work in the field of knowledge representation, and a special interest group for the application of ontologies with TEI has been formed [96]. Not only has this group shown how TEI could be mapped to Semantic Web concepts but also how TEI tags relate to the definitions of the CIDOC CRM. This work seems to be interesting for bringing together textual information and descriptions of cultural heritage entities.

Information that has been integrated from different sources oftentimes refers to the same entity in the world. Recognizing and resolving these situations are key aspects of ALAP. Bouquet et al. [44] describe the “Entity Name System” that should foster the re-use of the names of entities. This system has been in development since 2010, as part of the OKKAM project, which seeks to help to reconcile the information that has been contributed by different sources. The project actively discusses topics such as entity matching and management.

PELAGIOS [15] (PELAGIOS: Enable Linked Ancient Geodata In Open Systems) is a project that strives to apply the concept of Linked Data. It focuses on information systems that deal with places in the ancient world. Pelagios is being organized by a group of project partners, and comprises Perseus, Arachne, CLAROS and many more. The Pelagios Explorer was recently introduced; it explores new ways of information discovery and visualization. Pelagios is using the Open Annotation Collaboration ontology to manage information about these places.

A couple of further humanities related projects strive for establishing interoperability. Hiebel, Hanke and Hayek [140] presented the project HIMAT, which applies a methodology that integrates the data that is organized along CIDOC CRM with spatial data. Dörr [90] describes the LVPA project [107] where several project partners agreed to create an integrated information system that links several cultural heritage sources. Stein et al. [236] describe “museumdat”, a format that has been optimized to search the publications of museum documentation. It generalizes the format CDWA Lite (Categories for the Description of Works of Art) [150], which was originally developed for art history. At the same time it

conforms with the definitions of the CIDOC CRM.

The projects introduced in this section are working towards establishing infrastructural elements for interoperability and information integration. The CIDOC CRM is used by many projects but in different manners, either as internal database schema or exchange format or both. Other projects like Pelagios pursue another approach by relying on the Open Annotation Collaboration ontology. A significant number of projects make use of concepts that are elaborated in the Semantic Web community like RDF and the Linked Data Principles.

Many of the projects described in this section deal with the development of infrastructural elements. The focus of ALAP is much narrower than the efforts that are described above. ALAP concentrates on resolving the different entity descriptions that refer to the same entities in the world. However, certain infrastructural elements need to be simulated to an extent that makes the experiment possible. This includes, for example, the process of extracting information and cleaning it so that a matcher can use it.

3 Information Integration

Different aspects of interoperability were discussed in the preceding chapter. These were derived from a user scenario that elaborated a realistic research setting. It is argued that information integration is a vital aspect of interoperability because it brings information together that is needed to approach a number of research questions. However, certain (functional) requirements must be met to enable the interoperability of information systems to deal with information from the humanities. Thus, this section addresses the challenges and approaches of information integration.

Different architectures have been proposed for implementing an integrated information systems, which must be evaluated and implemented to achieve a particular objective. A key decision is whether information elements should be replicated in an integrated system or if they should be distributed among several systems. Most steps of the information integration process depend on information that is accessible to algorithms, free of errors and of high quality. Therefore, problems of data quality must be dealt with to prepare the information for subsequent processing steps. Information extraction techniques can help with extracting structured and meaningful information from unstructured or semi-structured content.

However, the main obstacle for most information integration endeavors is the problem of semantic heterogeneity. Matching techniques must be applied to identify semantic correspondences between data models, database schemata and data elements. Methods and techniques that allow for discovering semantic correspondences between data elements are examined under the notion of *entity resolution*. This field of research aims to identify either the different names that refer to the same entity in the world or similar names that refer to different entities. The identified semantic correspondences can then be represented as a machine actionable mapping. Finally, it must be decided whether corresponding descriptions of entities should be fused or if only links between them should be maintained.

3.1 Data, Information and Knowledge

Information integration is simply defined as the integration of information from different sources. But what exactly is information? And how can it be distinguished from data, knowledge and wisdom? Different definitions are suggested to clarify the nature of information. It can be assumed that processing data and deriving information always involves the aspect of interpretation. This cannot be done by human effort alone if massive amounts of data are being considered. Therefore, software has been developed to imitate the ability of human interpretation. This section will approach the topic by contrasting the paradigms of information theory and (formal) semantics.

Shannon introduced information theory [218] as a field that examines the peculiarities of signal processing from an engineering perspective. This branch of research tries to optimize the transmission of symbols that form data and information over a channel. A key issue is establishing communication channels that provide optimal compression and transmission [66]. But this branch of research does not deal with the message that is the subject of the transmission. However, different aspects of information theory, such as the concept of entropy, have been used to build machine learning components, and will be used for the purposes of this thesis. Favre-Bulle [103] concludes that information theory does not consider information as the result of cognitive interpretation.

Semantics is another, much older, branch of research that focuses on the relation between content and container. Auxiliary to syntactics and pragmatics, it is a branch of semiotics and studies the notion of meaning. In linguistics, semantics deals with the study of the meaning of linguistic units like words, phrases, sentences and larger units of discourse like texts. One important aspect is the study of relations between semantic units like homonymy (identical names refer to different things) and synonymy (different names refer to identical things). In computer science, semantics usually refers to the meaning of formal languages. Additionally, the notion “semantic network”, originally introduced by Peirce [195], describes a data model that is characterized by the use of directed graphs. These graphs are made of vertices that denote entities in the world and of arcs that represent relationships between different entities.

It has been indicated that information is the result of interpreting data in a context. Schneider [216] defines data as “formations of marks or continuous functions, that represent information based on known or assumed agreements. These formations are being processed or form the result of prior processing.” Gumm and Sommer [125] describe the process of representing information as data in information systems. Information is represented as data in information systems, and encoded as sequences of bits that are joined to bytes. But unless an interpretation of these sequences is known, the represented information cannot be extracted.

The context seems to be vital for extracting information from data. Context refers to the elements of a communication situation that determine the comprehension of an utterance. Examples for these elements are things that have already been uttered or knowledge that is shared by all cognitive agents. Because information may be sparse or incomplete, cognitive agents may not be able to directly infer the meaning. But if contextual information is available, an interpretation may be still possible. Therefore it is important not only to integrate data but also to integrate contexts. The various ways contextual information can be made available to information integration will be discussed in the following.

Favre-Bulle [103] emphasizes the relevance of information flows and the role

of cognitive agents. These agents can be human beings or software artifacts that interpret data in particular contexts. The result of this interpretation activity is that the agent gets to the meaning of the formations of marks that are encoded as data. Confrontation with data triggers flows of information can lead to learning or a growth of knowledge. This flow of information comprises the interpretation of (combinations of) marks with respect to its syntactical (rules of combination), semantical (relation between signifier and signified) and pragmatological (interpretation in context) functions.

According to what has been mentioned above, knowledge seems to be something that enables interpretation by contributing context. Karl M. Wigg [257] presented a practical definition of knowledge that his group has developed over time: “Knowledge consists of facts, truths, and beliefs, perspectives and concepts, judgments and expectations, methodologies and know-how. Knowledge is accumulated and integrated and held over time to handle specific situations and challenges. Information consists of facts and data organized to describe a particular situation or condition. We use knowledge to determine what a specific situation means. Knowledge is applied to interpret information about the situation and to decide how to handle it.”

What does it mean to integrate information from different humanities information systems? Analyzing flows of information seem to be more effective than looking at information alone. Data that is comprised of formations of marks must be combined with/ integrated into the contextual information, which enables information to flow. The activity of interpretation takes place at different stages of the process – both automatically by software artifacts and by humans. In the course of Semantic Web research, methods and techniques have been proposed to make this process more seamless. Therefore, the role of Semantic Web technology for enriching data in its context by entity resolution and entity resolution is evaluated in more detail later.

The flows of information that emerge from the interpretation of syntactic, semantic, or pragmatic mark formations play a major role in information integration. Intermediate systems act as cognitive agents: they convey information according to its meaning and help humans to maximize the extractable meaning. Contextual information is present in different forms. It is either produced by the assumptions of software developers, which materialize as algorithms, or is auxiliary structured information that has been semantically annotated. In summary, data must be integrated in a way that enables the joint interpretation and abstraction of information by human beings and information systems.

3.2 Distribution and Heterogeneity

Both distribution and heterogeneity are major obstacles for establishing seamless interoperability. But, at the same time, it would not be effective to approach these challenges by enforcing centralization and homogeneity. Therefore, how architectural and infrastructural problems can be approached without ending up with an inflexible and monolithic architecture must be explored. Distributed information access must respect the autonomy of single information systems while creating a foundation for information sharing; it must make effective use of computer networks and overcome semantic conflicts.

Özsu and Valduriez [194] have observed that distributed database system (DDBS) technology tries to bring together approaches that appear to fundamentally contradict each other: database systems seem to contradict computer networks. Database systems deal with abstraction layers so that application developers do not have to worry about how data is physically and logically managed. Additionally, the centralization of data management is often mandatory for reasons of efficiency. Thus database technology strives for integration and centralization, but computer networks work towards distribution of data and applications. The authors find that the key to understanding this dichotomy is to avoid confusing integration with centralization.

It has been emphasized that network technology plays a major role in the integration of distributed information from different systems. However, making extensive use of network technology may introduce additional problems. Rotem-Gal-Oz [211] emphasizes a number of assumptions that designers of distributed systems are likely to make. Bad practices resulting from sticking to these fallacies may raise serious obstacles in the long run for designers and operators of large distributed systems. Networks are usually not reliable, introduce latency and have limited bandwidth. Certain applications require high standards of security that are difficult to enforce in networked environments. Additionally, heterogeneous systems are almost always connected via networks that are prone to frequent changes.

Leser and Naumann [168] emphasize how distribution, autonomy and heterogeneity need to be considered by information integration projects. The physical and logical distribution of information is often needed to enhance the performance of a system and to ensure that data is safe from loss. At the same time, problems of network communication, schema incompatibility and query optimization are introduced. Organizations strive to maximize their alternatives of action by remaining autonomous with respect to hardware, software and the approach of system development. Thus, the different groups that implement the humanities information systems start with different premises, which causes them to develop systems that are highly heterogeneous. One way to tackle this problem would be to delimit this autonomy by introducing standards.

The authors distinguish different forms of heterogeneity that cannot be clearly separated from each other. Technical heterogeneity is resolved if two communication systems can communicate with each other without understanding the exchanged messages. Syntactical heterogeneity deals with information representation by different character encodings like Unicode. Data model heterogeneity can be resolved if a standardized data model for information integration is introduced. Furthermore, structural heterogeneity can be approached by introducing database schemata. And finally, semantic heterogeneity is attenuated by consistent names and structures to denote the same meaning. Different systems can only process information according to its intended meaning if these forms of heterogeneity have been resolved.

Among these forms of heterogeneity, semantic conflicts are the hardest to resolve. According to Frege, [111] the meaning of a complex expression is determined by the meanings of its constituent expressions and their compositions. In particular, this is relevant for schema elements with syntax and grammar. These (complex) symbols need to be interpreted by algorithms that have access to contextual knowledge. As previously discussed, contextual information is needed to extract meaning from unprocessed data. Therefore, it is important to represent this contextual information in a machine actionable way. However, a lot of this information, like the documentation of an information system, has not been formalized and cannot be used by algorithms.

To establish information integration, different systems need to be combined and certain functional requirements need to be met. This raises the question of how to combine these different systems into an overall architecture. Özsu and Valduriez [194] have described different kinds of architectures for information integration. Architectural designs range from centralized, monolithic descriptions of systems to highly distributed suggestions. Both alternatives have assets and drawbacks. Integrated information system planners and developers must find a compromise between acceptable distribution and helpful centralization.

The topics introduced in this chapter illustrate the challenges and endeavors in the context of information integration. A number of issues can be dealt with by developing technical means for networked communication as well as introducing standards for character encoding and data modeling. But problems with semantic heterogeneity, which often lead to semantic conflicts, are the main obstacle. Semantic conflicts comprise the inconsistent or ambiguous use of names for things. These conflicts mainly stem from heterogenous naming schemes for schema and data elements that are employed by different information systems. ALAP focuses on resolving the semantic conflicts that concern data elements by performing entity resolution. The use of Semantic Web technology, along with its suite of standards, was also considered and evaluated. Semantic Web research is a major effort to

approach the described problems.

3.3 Integration of Entity Descriptions

3.3.1 Semantic Heterogeneity

Semantic heterogeneity is a serious obstacle for the joint processing of information by multiple information systems. There are a number of ways in which semantic heterogeneity may appear during the process of information integration. Therefore, different kinds of heterogeneity must be identified and dealt with by making use of appropriate means. This section elaborates the different forms of semantic heterogeneity and explores the multiple approaches that are being successfully used.

At first sight, a name in an information system is nothing more than an arbitrary sequence of characters. But this name symbolizes a representation of a concept in our minds and – at the same time – refers to a material or immaterial thing in the world. Figure 6 illustrates the coherence of symbols (names), concepts and objects in the world for a given context.⁵ The concept is termed the *intension of a name* and the set of objects in the world that is described by this concept is the *extension of a name*. In the classics, for example, the intension (Sinn) of *sculpture* is its concept, which is usually conveyed by a description. And the extension (Bedeutung) of *sculpture* is the set of all sculptures that fall under this definition. In this way, a name denotes both a concept and a set of objects that fall under this concept.

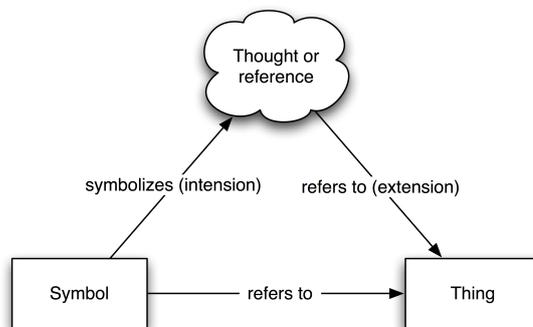


Figure 6: The semantic triangle.

However, this figure does not show how semantic relations with different names relate to each other. In particular, synonymy and homonymy are problematic.

⁵The figure follows the semantic triangle that has been introduced by Richards and Ogden [191].

They lead to situations where information retrieval systems either neglect relevant information or consider irrelevant information. Names are synonyms if they are different but have the same denotation (“student” and “pupil”). Names are homonyms if they are equal but have different denotations (“fluke” can denote a flatworm or the end parts of an anchor). In many cases the denotation of one name contains the denotation of another name. For example, the extension of “action film” is contained in the extension of “film”. In this case, the latter is a hyperonym of the former and the former is a hyponym of the latter.

Many of the semantic conflicts that have been described can be resolved by considering contextual information. Therefore, it is very difficult to find and resolve semantic conflicts between information system elements if contextual information is lacking. It has been emphasized that contextual information is not explicitly formalized so that it can be used to automatize the process of semantic conflict resolution. When developing an integrated software system, software developers often have limited information about database models, the schemata and a couple of database records. Many systems have their business logic hard coded as software components that are poorly documented. Few information systems come with proper documentation for their database schema and the software developer needs to infer the meaning of the elements he encounters.

Different information systems are built according to different design decisions. The causes of heterogeneity among information systems are described above. Semantic heterogeneity stands out because it is a major obstacle for information integration and there is no easy resolution for it. Both database schemas and – even more important – the content, which they organize and structure, are constantly changing. Manually resolving these conflicts may work for database schemas that are of reasonable size. But it is not efficient for the entities themselves (the actual content of an information system). Therefore, different means are explored that are suitable to approach this situation. Ways to explicitly formalize semantics and techniques to deal with unstructured information are introduced in information technology to automatically resolve these problems. These are applied in the course of ALAP.

3.3.2 Mapping and Matching

Although no clear distinction can be drawn in theory, the metadata schemas are often separated from actual data, which is organized along these schemas. The semantic conflicts, which are described above, are relevant for both metadata schemas and the data itself. Usually, metadata schemas remain constant over a certain period of time due to standardization efforts. But this does not hold true for the data elements that are subject to constant modification through change, addition or deletion of content. Therefore (semi-)automatic means are elaborated

to approach these problems.

Different database management systems are relying on different data models. There are very simple systems that can only store key-value pairs without imposing any explicit structure. Relational databases have introduced additional means to store sets of data objects with similar features in tables. Relations can be defined between tables to make structures explicit. Thus, the database schema determines how data is organized in a database management system along a certain data model. Although semantic conflicts may have been resolved among different database schemas in the first step, there will probably be an overlap of the actual data elements as well. Data elements that have been drawn from different information systems may describe or document the same object in the world.

Information integration relies heavily on discovering and documenting semantic correspondences between schema and data elements. Kashyap and Sheth [156] have introduced the concept of schema correspondences to describe the structural similarities of data elements. There can be very simple correspondences for one element from a schema that corresponds to exactly one element from another schema. However, these correspondences are often more complex, and one schema element corresponds to multiple elements of another schema. Additionally, the problems of hyperonyms and hyponyms are examples of more complex semantic relation. But semantic correspondences can also exist between data elements themselves. As with schema elements, semantic correspondences between data elements can be very simple but also rather complex. ALAP will focus on *coreference*, a rather frequent semantic correspondence between data elements. In this situation, two entity descriptions refer to the same thing in the world.

Schema mapping means to describe and document the semantic relationships of elements from multiple schemas. This documentation can be used to generate transformation algorithms that implement the mapping. These rules need to be implemented as a set of instructions that can transform data from one schema to another. Special query and transformation languages can be used for this. For example, XQuery and XSLT can be used for transformations on XML. Popa et al. [200] describe an approach for schema mapping of XML and relational data that has been used in the course of the Clio project. The Clio tool can automatically generate queries from the schema mapping that are consistent with the semantics of multiple information systems. In the context of knowledge representation systems automatic inference is used to answer queries without permanent transformation of data.

In situations where huge amount of schema and data elements need to be considered, manual mapping of elements may be prohibitive. If resources are scarce or the amount of elements is huge, it can be helpful to apply automatic processing techniques on these elements that can establish (preliminary) correspondences.

This approach can reduce manual mapping work significantly, even if it only produces preliminary results. Since the amount of data elements is usually much higher than the amount of schema elements, these techniques are vital for identifying correspondences. The process of identifying related information in different information systems can be compared to the identification of patterns in a large set of data. Zhao [266] has compiled a survey of schema and data matching approaches, which make extensive use of methods and techniques from data mining and machine learning. These approaches will be introduced later.

To begin, schema and data elements must be accessible and correct for information integration and the joint processing of integrated information. In this context accessibility means that the elements can be properly interpreted by humans and algorithms. Wang and Strong [251] have enumerated several criteria to evaluate the data quality of information sources. They emphasize the relevance of good data quality for the consumers of information. Figure 7 illustrates a classification of common data quality problems proposed by Rahm and Do [206]. The authors distinguish single-source and multi-source problems that can either arise on an instance level or on the schema level.

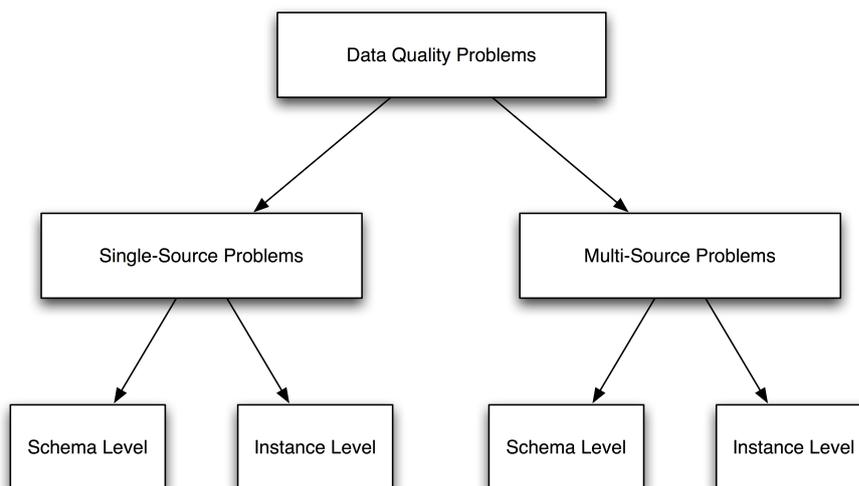


Figure 7: A classification of data quality problems in and among data sources.

The heterogenous data models, database schemas and overlapping data elements that have already been mentioned belong to the class of multi-source problems. Thus, ALAP strives to solve a data quality problem by identifying the entity descriptions that refer to the same entity in the world. The result of joint information processing will certainly be more useful if high quality data is available. Common data quality problems like different representations for values (multi-source / instance), erroneous data (single-source / instance) or fields with internal

structure (single-source / schema) are a hinderance for matching data elements. Thus, ALAP strives to enhance data quality but relies on a certain level of data quality itself.

Information integration often involves the physical or logical materialization of data in one place. If two information systems that need to be integrated are thematically related, the probability is high that both information systems host information about similar entities to a certain extent. In the best case, this can be an annoying situation because users are confronted with multiple representations of the same entity. But this situation can also be harmful if the automatic processing or inferencing produces wrong results. Thus, different descriptions of the same entity need to be identified and dealt with, for example, by fusing the information.

Data models determine how data elements can be organized by schemas. And semantic conflicts are relevant for both schema and data elements of information systems. Semantic correspondences must be discovered and documented in a machine actionable way to overcome these conflicts. ALAP, which is described in the following sections, focuses on matching data elements. It has already been mentioned that the parts of the schemas of Arachne and Perseus that are relevant for information integration are being manually mapped. ALAP relies on this preliminary work and focuses on identifying coreference relations between these data elements. Although schema matching and mapping are important for information integration in general, it is of minor relevance for this experiment.

3.3.3 Semantic Information Integration

In many projects, data mining and machine learning are successfully used to perform information integration. These tools make use of statistics and deal with uncertain and fuzzy information to a certain extent. More formal methods are also explored to approach the problem of information integration and joint processing. These approaches rely on formal logics and are extensively implemented as part of Semantic Web technology. Semantic information integration assumes that the structure of a domain can be modeled with description logic. This section introduces description logics as a particular approach to semantic information integration.

Helping human beings with problem solving is an important aspect of software development. Still, human involvement cannot be replaced in most situations. Research in artificial intelligence explores such situations where automatic problem solving is promising. The algorithms that manipulate data in a certain ways are usually the central elements of research in this area. Data is being created to model sections of the world, and computer programs are being developed to perform automatic reasoning on that data. To that end, computers rely on formal systems that work with formal semantics.

Researchers in the field of Artificial Intelligence are therefore exploring efficient ways to implement formal logics. Russel and Steward [188] have elaborated the concept of intelligent agents as software artifacts acting in useful and goal-driven ways. To that end, these artifacts must represent knowledge about the world that they are acting in. Additionally, they need to be able to draw conclusion on the basis of this knowledge – a precondition for acting in a useful manner. This knowledge is usually represented in knowledge bases that contain sentences expressed according to the syntax and semantics of a representation language. A logic will define the semantics of the representation language to determine the truth of a sentence with respect to a certain model. Different logics with different definitions for syntax and semantics have already been proposed; they differ mainly in their expressiveness.

Boole [43] laid the foundation for a rather simple formal logic. Boolean algebra became the foundation for propositional logic and was later developed by Frege in his “Begriffsschrift” [110]. Unfortunately, propositional logic is not powerful enough to represent complex domains. Frege’s contribution has also been an important step towards the development of first-order-logic. In FOL, means are introduced to represent objects and their relations in addition to facts. Higher order logics have been introduced so that assertions about relations rather than objects alone could be made. Modal logics have been introduced to augment first-order logic with operators like “necessity” and “possibility”. This enables reasoning not only about facts but also about knowledge about facts. Another extension, temporal logic, assumes facts to be true only in a particular time interval. Other approaches include probability theory, which can model beliefs and fuzzy logic in a way that represents degrees of truth.

It has been recognized that a consistent and shared vocabulary can foster communication in domains with a large number of participants. Therefore, research in the field of knowledge representation has focused on systems that enable concepts to be organized as taxonomies. While early research in Artificial Intelligence focused on understanding expert systems, more recent projects deal with creating so-called ontologies. For example, Lenat et al. [167] describe the CYC project, which created a knowledge base of ontology and common sense knowledge to enable agents to reason about common sense facts.

Ontologies and knowledge representation has also been adopted by researchers in the Semantic Web community and will be elaborated later. It is relevant because the explicit definition of semantics for formal languages allows for automatic inference. In standard artificial intelligence settings, intelligent agents can use this information to infer new information that is necessary for useful and goal-driven actions. In the field of information integration, automatic inference can be useful to represent the contents of an information system in a standardized formal lan-

guage with defined semantics. Inferencing on a set of shared concepts can then be used to retrieve relevant information from multiple information systems.

First-order logic is very powerful. It could be used to build an ontology and to model facts about the world. There is another branch of logics that has developed as an offspring of the previously mentioned semantic networks: description logics. Many description logics form a subset of first-order logic to allow for effective and efficient automatic inference. The attributive concept descriptions with complements (ALC), introduced by Schmidt-Schauß and Smolka, [215] specialize in defining concept hierarchies and roles. The Web Ontology Language (OWL) and its heavy uses of ALC concepts will be covered in more detail later.

Description logics have been applied in the field of information integration. Since categories and the relations of their members are an important feature in knowledge representation, they are also used in most database systems. For example, in a relational database tables can be considered as categories, and primary and foreign keys form relationships. If the contents of information systems can be translated into concepts of description logic, categorial inference can be used to build an integrated information system. Hebel et al. [135] describe different ways to relate the ontologies of different information systems. If the categories are mapped onto a global ontology, queries can be formulated as definitions of categories. These categories are then incorporated into the global ontology and the related concepts.

Formal logic and the formal languages associated with it can be used to represent facts about the world. These facts could be made into a formal language or transformed into a traditional database system. And the fact that description logics allow for categorial reasoning can be exploited for information integration. A number of practical applications of these concepts are implemented in the Semantic Web community. However, they do not explicitly allow for the discovery of coreference relations of data elements that are organized along an ontological vocabulary. Categorial reasoning with description logics relies on data that is available in a very formalized way. Moreover, it has been emphasized that introducing additional formality in information systems can be challenging.

Semantic Web technology has recently become rather popular for information integration in the humanities. Therefore, a couple of projects are examining its performance for information integration with varying results. For ALAP, Semantic Web technology will be important for ingesting and exporting information. For the actual processing of information, methods and techniques from data mining and machine learning still prevail. This is due to the fact that discovering entity descriptions that refer to the same entity in the world relies on methods that deal well with fuzzy and messy data.

3.4 Integration Architecture and Infrastructure

Information integration enables the usage of information in ways that have never been possible before. But information integration also introduces additional complexity due to its reliance on the complex transmission and manipulation of information itself. One way to approach the complexity of systems in the design phase is to adhere to certain architectural patterns and paradigms. At the same time, the architecture of a system should enable effective and efficient ways of problem solving. This section reflects on the role of architectural patterns in information integration.

Complexity has long been recognized as a problem of software development. For example, by isolating domain logic from the user interface in software development, a layer structure has been introduced. These layers of abstraction have been used in many fields like network protocol design but also in defining database and information system architectures. Leser and Naumann [168] distinguish six different architectural design for databases: Monolithic databases, distributed databases, multi-database systems, federated databases, mediator-based databases and peer-data-management systems. From the first to the last mentioned systems, the complexity of distribution, autonomy and heterogeneity increases.

Architectures of integrated information systems can be classified according to the features that have been elaborated so far: autonomy, heterogeneity and distribution. Figure 8 illustrates this coherence as a three dimensional coordinate system.⁶ Classical monolithic database management systems are located in the origin of the coordinate system because there is just a single database system. They are not distributed because the database software runs on only one computer; clients only communicate with this machine. On the other hand there are peer database management systems that stand out due to high values on all three axes.

Sheth and Larson [220] have presented the five-level architecture of a federated database system. This architecture has abstraction layers for all important information integration tasks that are illustrated in figure 9. The local database schema is expressed in a canonical data model by the component schema layer. Parts of this schema are made visible to a federating system that maps schema elements onto a federated schema. Finally, parts of the federated schema are made visible to external applications. Many architectural designs for information integration can be interpreted as variations of this five-level-architecture.

However, while the five-level architecture is useful for dealing with data model and schema design heterogeneity, it does not provide a means for overcoming instance data heterogeneity. Therefore, figure 9 can be seen as one specific view

⁶The illustration follows Tamer and Özsu [194].

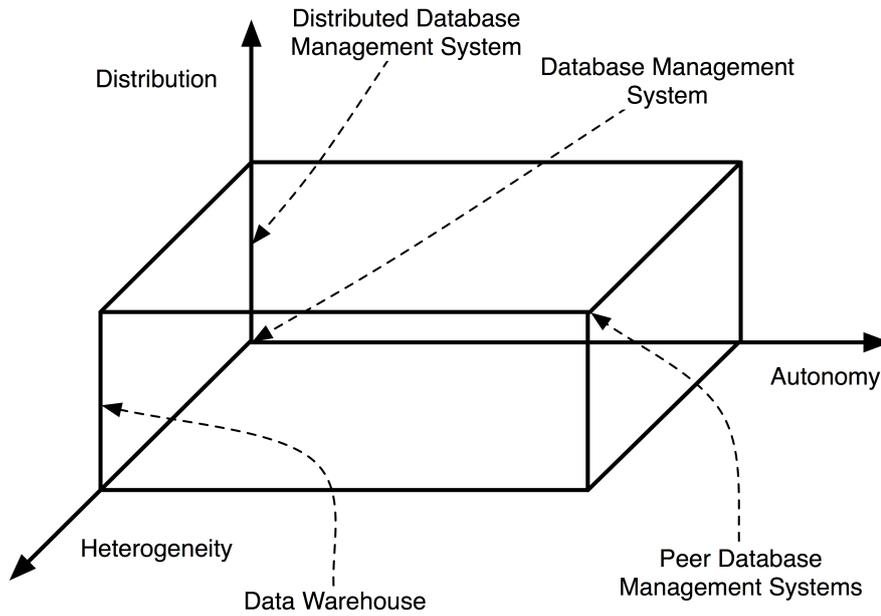


Figure 8: Architecture classification of integrated information systems.

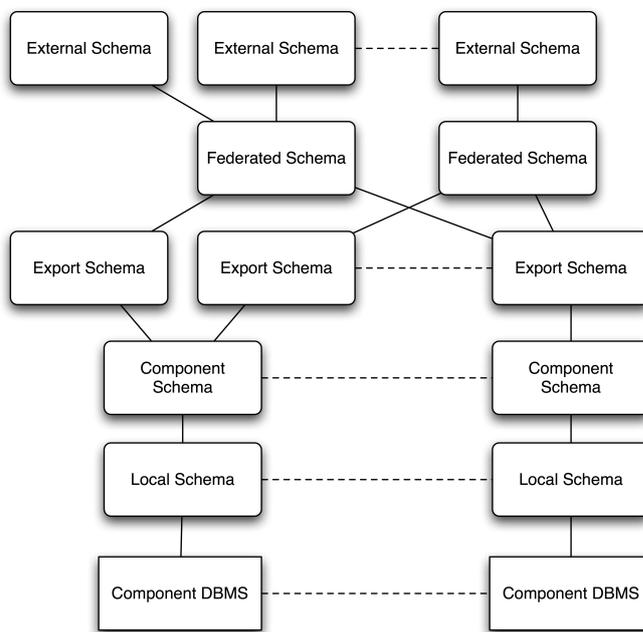


Figure 9: Five-level-architecture of a federated information system.

on a part of an information integration architecture. Additional components may be needed to perform caching or that physically materialize data for efficient processing. In principle, each layer and component of an architecture needs software components to be implemented. Thus, architectures describe the overall system from different perspectives and enumerate functional components that are necessary for solving a specific problem.

On the other hand, a certain infrastructure is needed for information integration. The term infrastructure refers to long-term physical and organizational structures that are needed for the operation of a (distributed) information system. Of course, infrastructural elements could be the physical computers located in a data center or the network hardware that enables physical communication. But infrastructure also alludes to software components that need to be implemented or the software components that implement the tasks of needed functional requirements. In this sense an infrastructure is the implementation of a certain architecture. Once an architecture has been implemented, it becomes extremely resource intensive to change the architectural decisions.

Even for software architectures that are not distributed, different parts of a system need to communicate with each other. Specifications are typically developed to enable software programs to communicate locally or over a network; this is referred to as application programming interface (API). Different entry points can be defined to call functions in software systems either locally or over a network. If software modules are distributed over a network, these entry points are called end points by convention, a term that has been coined in the field of Service Oriented Architectures. For example, Concordia, Gradman and Siebinga [64] describe the API design and its calling conventions together with use cases for accessing the functionality of Europeana.

The term middleware is often used to describe an application programming interface that enables different software components to connect in a centralized or distributed fashion. Middleware implementations can be described as software frameworks that support software developers with establishing complex software architectures. It strives to provide a software economy to establish interoperability between several information systems by abstracting certain resources. Many middleware systems provide complex messaging architectures that can be used for the unsupervised communication of different components that are distributed system-wide. Middleware is often based on standards like XML and SOAP, which form the building blocks of Web Services and Service Oriented Architectures.

Different infrastructural elements have been developed in the past to tackle problems of cultural heritage information integration. Particularly for bibliographic information exchange, Z39.50 has been developed under the auspices of the Library of Congress. Herbert [81] describes another notable approach that

has been put forward by the Open Archives Initiative: the OAI Protocol for Metadata Harvesting (OAI-PMH). Bizer, Berners-Lee and Heath [40] describe another rather new development under the notion of Linked Data, which attempts to make highly structured data available to a wide community. All three developments have certain features that make them useful in different contexts. While Z39.50 and OAI-PMH have established standards, Linked Data is recently becoming popular in the context of Semantic Web research. It will be dealt with in more detail later.

The advantages and disadvantages of different architectural and infrastructural decisions have been discussed in this section. These opportunities and drawbacks need to be considered and discussed by the planners, developers and users of such systems. Centralization results in a reduction of administrative tasks. In certain situations, it allows for high efficiency because the information does not need to be transmitted over networks. However, distribution enables the parallel processing of information and can be very efficient in other situations. Additionally, centralized approaches often do not meet the requirements of existing information system landscapes.

Algorithms that are developed for ALAP must materialize the information from different sources in local memory to determine similarities. Thus, the software components run on a single physical machine to achieve maximal efficiency. However, certain tasks of ALAP could be computed in a controlled, distributed fashion in the future. Infrastructural elements that allow for controlled information exchange over networks must be established in larger projects. But for the time being, they will be abstracted and implemented locally.

4 Semantic Web Research

I have a dream for the Web [in which computers] become capable of analyzing all the data on the Web – the content, links, and transactions between people and computers. A ‘Semantic Web’, which should make this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The ‘intelligent agents’ people have touted for ages will finally materialize [26].

The World Wide Web provides a global infrastructure that can be (and has been) exploited for large scale information integration. Semantic Web research builds on this infrastructure by creating concepts and tools. These concepts and tools should help to develop software systems that support the global integration and inferencing of information. An interesting aspect of Semantic Web research is its rootedness in the artificial intelligence community; it makes heavy use of rigid formalisms to represent and manipulate knowledge. The CIDOC CRM, a so-called ontology, is related to the research in this community. It has been developed to represent knowledge in the cultural heritage domain and to enable information integration and inferencing. This section introduces important concepts, fundamental research topics, opportunities and challenges of Semantic Web research.

Most concepts that are relevant for the Internet still rely on the concepts that have been developed by hypertext research. According to these concepts, documents can be related to each other by hyperlinks. These hyperlinks are machine-actionable; a user can click on a link in a hypertext browser and be referred to the target of the link. This works well because humans understand the information that these interlinked documents convey. Unfortunately, the same does not hold true for computers because of their obvious lack of formality.

In 2001, Berners-Lee, Hendler and Lasilla [27] presented their vision of the World Wide Web successor . They proposed extending the current World Wide Web by creating more formalized and standardized ways of knowledge representation to allow for automated information processing. In 2006, Berners-Lee, Hall and Shadbolt [217] then reflected on how research is progressing towards the presented vision. Unfortunately, they observed that this vision, which was formulated years ago, is still far from being implemented. This can be interpreted as an alarming situation or as an indication to reevaluate the proposed methods and techniques.

Due to the complex history and influencing disciplines, it is not easy to establish a definition for the term *Semantic Web*. Many concepts have been elaborated and standardized under the auspices of the World Wide Web Consortium.[249] A partial definition can be found there:

The Semantic Web is about two things. It is about common formats

for integration and combination of data drawn from diverse sources, where on the original Web mainly concentrated on the interchange of documents [sic!]. It is also about language for recording how the data relates to real world objects. That allows a person, or a machine, to start off in one database, and then move through an unending set of databases which are connected not by wires but by being about the same thing.[248]

This refers to fundamental concepts that have also been elaborated in the area of information integration research. Although a global network infrastructure – the Internet – has emerged, data is still controlled by single applications. Additionally, many websites do not provide structured data at all but unstructured documents that are connected by hyperlinks. Thus, additional components need to be implemented to make this vision into a reality. For example, textual data with only weak semantic annotations could be transformed into structured information. After this has been completed, the subsequent steps of the information integration process, which have already been elaborated, can help to relate information about the same thing in different information systems. Semantic Web technology achieves information integration by using formal semantics and reasoning about categorical information.

Hitzler, Krötzsch and Rudolph [141] classify Semantic Web concepts into three main areas: knowledge representation, knowledge sharing and calculating with knowledge. Although representing textual information in information systems is absolutely useful in many areas, it may not be useful for certain processing needs. If structured information is available in some form, or if it has been extracted from less-structured information, it can be represented in a form that allows for automatic processing according to a desired goal. In artificial intelligence research, the multiple methods of knowledge representation that allow for effective and efficient automatic reasoning (calculating with knowledge) have already been elaborated. The Semantic Web community has obviously been influenced by these developments. The third area of research focuses on sharing information; namely, it looks at how distributed information about the same thing can be effectively discovered and retrieved.

Thus, information integration is an integral part of research in the Semantic Web community. To that end, Semantic Web technology provides means for explicitly specifying how information from different provenances can relate to each other. For example, Semantic Web technology has elaborated representation languages that provide language elements that can associate different names referring to the same thing. However, in many cases it is not evident whether information from different sources is related in the first place. Although Semantic Web concepts can explicitly identify related information, they do not focus on discovering

these relations. If this associational knowledge has not already been explicitly modeled somewhere else, it needs to be discovered.

Discovery of related information from different information systems has been extensively studied in the database community. Concepts like similarity, probability and learning turned out to be helpful in situations with limited explicit correspondence information. Although these concepts are currently being incorporated in Semantic Web concepts, there is no suitable implementation so far. Therefore, it seems to be helpful to use algorithms that have been proposed by the information integration community on data that is represented according to Semantic Web concepts. However, the representation of probabilistic knowledge in a language that is based on description logic is still in an experimental state. Plus, there is no implementation to evaluate this information or to perform automated reasoning.

Software developers who need to implement an information system that makes use of Semantic Web technology can rely on prebuilt frameworks. According to Hebel et al. [135], a Semantic Web framework consists of three parts: storage, access and inference. A knowledge base constitutes the storage component; it stores information that has been encoded according to Semantic Web concepts. The underlying technology is often implemented as a graph data model, which is well-suited to manage highly structured information. Additionally, a query API is needed to provide access to the stored information. Some queries can be answered directly by referring to knowledge that is explicitly represented within the knowledge base. If the answer cannot be answered by the knowledge base alone, the query can be passed to a reasoner component that applies inference rules to infer the answer from explicit knowledge. Software developers can resort to the functionality of these frameworks by including computer libraries and running additional services that can be called via an API.

Although no killer application has been seen yet, end-users and software developers already have a set of helpful Semantic Web tools at their disposal. Furthermore, large research projects are using the ideas of the Semantic Web to envision and implement semantic user scenarios for the public and private sector. To this day, the number of Semantic Web projects has become so large that they cannot be covered exhaustively. The Linked Open Data Community Project has recently become popular and strives to publish freely available data sets as RDF. Guidelines have been proposed so that the data can be explored and re-used in other context [40]. The DBpedia project has been an important data provider for this project because it makes Wikipedia content available by using Semantic Web concepts [41]. A couple of projects that are relevant to ALAP will be described in more detail later.

The vision piece that has been elaborated by Berners-Lee, Hendler and Lassila

[27] introduces a compelling usage scenario. But by drawing on concepts from AI research and the World Wide Web, it also inherits the challenges and shortcomings that have been discovered in these disciplines. Because of its scope (the Internet), it will have to deal with problems of computational complexity on huge amounts of information. Many statements represented in textual web documents are either vague or uncertain. Therefore, Semantic Web concepts should strive to represent both and to provide adequate reasoning capabilities. Furthermore, if the environment cannot be centrally controlled, the information will be contradictory and inconsistent. Semantic Web concepts rely on strong formalization because it introduces additional problems when interaction with humans is needed. The challenges introduced by strong formalisms have already been mentioned.

Semantic Web research introduces interesting concepts for integration, discovery and processing of distributed information. It adopts and modulates concepts from different disciplines. In particular, AI research and the Web community contribute reasoning capabilities and infrastructure. The following sections introduce the most important concepts of Semantic Web research and estimate their potential for ALAP. The vagueness, uncertainty and formality of the methods and techniques are highlighted – all common themes in research on information integration. The introductory character of this chapter also fosters an understanding of the CIDOC CRM, a formalism to represent cultural heritage information.

4.1 Fundamental Concepts

The W3C published a set of standard recommendations after Tim Berners-Lee and others expressed their vision for extending the World Wide Web in the future. These recommendations should help to develop information systems that do not monopolize control over their data. Thus, the term *Semantic Web* is often used to refer to this set of published concepts. This section introduces the recommended standards by emphasizing their interaction, which has been illustrated as a layered model. At the same time, the Semantic Web's fundamental concepts should foster a basic understanding of it and further the discussion of its contribution to information integration. The definition of formal semantics for the suggested syntactical models is also highlighted because it seems to be rather important for information integration efforts.

Originally introduced by Tim Berners-Lee to illustrate how proposed Semantic Web concepts relate to and build upon each other, the Semantic Web Stack has been further elaborated since then. Figure 10 shows a recent version of this “layer cake” that refers to the most important concepts currently being discussed in the Semantic Web community. Very low-level concepts can be found at the bottom of the stack; these are used to define a standardized character set and to identify things in general. The middle layers introduce syntactic elements that are attached

to formalized semantics; these structure information and make assertions about things. The higher levels of the stack deal with exploiting these semantics by making intelligent use of represented information.

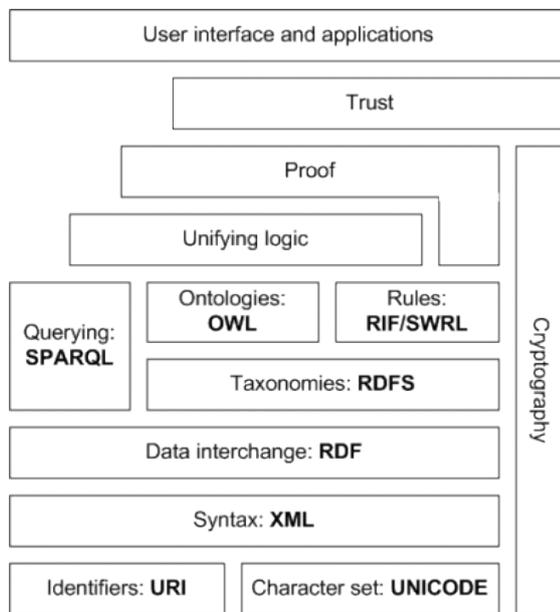


Figure 10: The Semantic Web “layercake” diagram.

The three foundational concepts mentioned above are realized when the Semantic Web Stack is implemented: modeling of knowledge, sharing of knowledge and calculating with information. Although these concepts are weighted differently in different communities, each is subject to active research. Regardless of its provenance, data is represented in a structured and formalized way that is meaningful to computers. Thus, the proposed formalisms provide expressive syntax and semantics to make assertions about things. This enables computers to automatically process the represented information in a meaningful way.

The most prominent example is of computers performing automated reasoning, which is the Achilles heel of Semantic Web technology because of its demanding concepts and involved computational complexity. Current approaches are, therefore, rather limited. But modeling and calculating with knowledge have been proposed to foster efforts in sharing information. In fact, the standards and technologies that have been proposed can help to share and integrate information. But the capabilities for discovering related information are rather limited due to the exact formalisms that are being used.

The concepts discussed in the following chapters have been elaborated over the last 12 years. The techniques, which are inspired by AI research, have an even

longer research tradition. Still, it seems that the adoption of these concepts under the notion Semantic Web is not well understood and needs further reflection. Some developments are rather new and recognize that the traditional goals of Semantic Web research are overambitious. Thus, the Semantic Web is still discussed controversially and is perceived differently by different communities. The following sections illustrate its means for modeling and calculating information.

4.1.1 Modeling Knowledge

Almost all information systems depend on some knowledge about the world, which they in turn must represent in a machine-actionable way. Therefore, digital surrogates of entities and their relations to each other in the world are managed in knowledge bases, and certain operations are defined for manipulation. Knowledge modeling and calculating are two sides of the same coin. Semantic Web research applies many of the recent developments in knowledge representation and reasoning. This section introduces the fundamental concepts that help developers to represent knowledge for their specific purposes.

Knowledge representation depends on mechanism to unambiguously identify entities that are part of an assertion. In first-order logic, the mechanism of interpretation can be used to refer to entities in the world by using non-logical symbols. But since these interpretation mechanisms are dependent on context, symbols usually refer to different entities in different contexts. For example, relational databases assign primary keys to a database record that is only unambiguous in the context of a certain database system. If the information from different information systems is integrated, this system of identification will fail if the other databases use the same combinations of table names and primary keys. An unambiguous association between the signifier and the signified is not possible in these situations.

Therefore, Berners-Lee proposes Uniform Resource Identifiers (URIs) for the identification of abstract and physical “resources”.⁷ A resource can be a document on the World Wide Web, or some other entity that needs to be identified on the Web, even if it cannot be accessed on the Web. URIs enable different communities to talk about identical entities. It is important to note that a URI always has the same interpretation, regardless of which information system is using it. The association between the identifier and the identified is globally fixed.

The Semantic Web has been built upon the Resource Description Framework (RDF), which was developed to represent knowledge about a certain domain. It is organized according to statements, each one forming a so-called triple that consists of a subject, a predicate and an object [47]. For example, imagine a database that

⁷URIs have been proposed by Berners-Lee et al. [25].

contains a table “sculpture” with an attribute “material”. For the database record that has the primary key “4711” the attribute has the value “marble”. At least three triples can be derived from this information: “4711”, “is a”, “man-made object”; “4711”, “has type”, “sculpture” and “4711”, “consists of”, “marble” (the vocabulary is based on the CIDOC CRM). Tools have been developed to map the internal structure of a relational database to RDF.[39]

Since RDF refers to conceptual and material entities by using URIs, the subject, predicate and object of these triples need to be rewritten to conform to the recommendation: <http://arachne.uni-koeln.de/item/4711> <http://arachne.uni-koeln.de/relation/consistsOf> <http://arachne.uni-koeln.de/material/marble>. Although RDF is introduced as a framework, there are various ways to serialize the knowledge, which have already been mentioned. The syntax of XML can be used to express a formal language for knowledge representation. Therefore, the W3C proposes using XML to serialize information modeled according to RDF.

Philosophers have been interested in the nature of reality for a long time. The branch of philosophy that is systematically researching this topic is called metaphysics. In particular, the question whether certain entities exist is discussed under the notion “ontology”. Furthermore, questions of grouping and hierarchical relations have been discussed under the notion ontology.[142] In information science, and particularly in artificial intelligence research, the term has been adopted to refer to a model of the world.

Ontologies have been incorporated in information systems to represent knowledge about the world and to enable different systems to share knowledge. Gruber [124] has established the term “ontology” in the information technology community: “An ontology is an explicit specification of a conceptualization.” Ontologies provide a representational machinery that models entities that are grouped into classes with restricted relations. Recently, both communities, philosophy and information technology, have drawn closer by exchanging research results.

Since ontologies have become the standard mechanism for knowledge representation, the RDF Vocabulary Description Language (RDF Schema or RDFS) has been elaborated as a semantic extension of RDF. RDFS provides means to describe groups of resources that are related and to specify the possible relationships between them.[47] By introducing these additional concepts, a particular form of knowledge representation by ontologies has been at least partially realized.

The W3C has introduced an additional formal language, the Web Ontology Language (OWL)[179]. OWL adds additional vocabulary, which allows for greater expressivity than RDF and RDFS. While it builds on the structures that describe classes of objects and their relations, it adds means like stating the equality of classes or introducing cardinality constraints (for example, a man has exactly two arms). Because of the involved computational complexity, OWL has been

split into three sublanguages: OWL Lite, OWL DL and OWL Full. OWL Lite provides a reduced vocabulary for building classification hierarchies with simple constraints. OWL DL is expressed in a way that corresponds with description logic. It ensures computability within a finite amount of time by ensuring computability and decidability.

OWL Full provides the greatest expressive power but cannot make any guarantees on the computational features of the knowledge base. A newer version of OWL has been proposed under the name OWL 2; it adds new functionality while maintaining backward compatibility [250]. All these developments aim to establish means to formally define the structure and meaning of the vocabularies used by information systems. The CIDOC CRM, which is dealt with in more detail later, has been formalized in OWL.

The expert systems that were studied in the 1980s made extensive use of facts and rules. These rules can be seen as a simplification of first-order logic, and they also allow for the representation of domain knowledge. By applying inference rules, syntactic structures can be transformed into new structures while preserving the truth of assertions. Meanwhile, many different rule systems have evolved since then and need to be unified somehow. This is why the W3C recommends the Rule Interchange Format (RIF) to exchange rules between different rule languages [157].

In many situations, it is useful to query a knowledge base for a certain set of assertions. Since RDF is a data model that represents knowledge as a directed, labeled graph, it would be useful to have a query language that could query for certain graph patterns. Therefore, the W3C published the SPARQL Protocol And RDF Query Language (SPARQL) to query graph patterns that have been constructed with RDF [202]. A SPARQL query can usually be formulated as one or more alternative graph patterns that represent a set of assertions. Then, the query API of a semantic framework can either refer directly to the knowledge base or pass the query to an inference engine.

The different standards and recommendations published by the W3C provide the means for knowledge representation in a wider community. Some concepts are adopted from artificial intelligence research, which has studied knowledge representation and reasoning for decades. Other concepts like URIs, which are integrated into the proposed knowledge representation formalisms, are being developed in the Web community. Although specialized tools for knowledge representation are available, it is obvious that substantial training is needed to acquire an understanding of the syntax and semantics of the addressed recommendations and standards.

4.1.2 Calculating with Information

If there are no formal semantics defined for RDF, the interpretation of represented assertions must be implemented intuitively. This leads to a situation where different software components with similar reasoning behavior come to different conclusions for the same queries. This problem can be addressed by defining the formal semantics for the Semantic Web languages. But, at the same time, this introduces questions of computational complexity, time and modeling effort. Therefore, this section introduces and discusses the benefits and shortcomings of formal semantics for the Semantic Web.

In the early days of the Semantic Web, Berners-Lee [22] reflected on the Semantic Web as a language of logic. He refers to the functional requirements of knowledge representation systems, as formulated by Crawford and Kuipers [69]:

In the broadest sense the study of knowledge-representation is the study of how to represent knowledge in such a way that the knowledge can be used by a machine. From this vague definition we can conclude that a knowledge-representation system must have the following properties:

1. It must have a reasonably compact syntax.
2. It must have a well defined semantics so that one can say precisely what is being represented.
3. It must have sufficient expressive power to represent human knowledge.
4. It must have an efficient, powerful and understandable reasoning mechanism.
5. It must be usable to build large knowledge-bases.

It has proved difficult, however, to achieve the third and fourth properties simultaneously.

In the same contribution, Berners-Lee emphasizes that the focus of Semantic Web languages should be on the third property, and any aspect of the fourth property that contradicts the third should be sacrificed. On the other hand, the features of the fourth property seem to be indispensable in many cases. Therefore, a large amount of research has been devoted to finding a compromise between expressive power, reasoning capabilities and efficiency. The following paragraphs will elaborate the issues that are associated with this dichotomy of expressive power and efficient reasoning.

To meet the second property of the aforementioned list, Hayes [133] has described the model of theoretic semantics for RDFS, OWL and SWRL. It has been

formally defined so that developers of semantic stores can implement a proper reasoning strategy. OWL is based on the description logic *Attributive Concept Language With Complements (ALC)*, which was introduced by Schmidt-Schauß and Smolka [215]. Additionally a couple of extensions have been introduced to ALC. Many description logics are decidable fragments of first-order logic. While OWL Lite and OWL DL are decidable, this is not the case for OWL Full.

An important backbone of the Semantic Web is the development of semantic stores that are both scalable and efficient. Despite the expressivity of RDF(S), it still does not allow for a reasoning system to be built in a way that compares to the efficiency of relational databases. Even OWL Lite, the least expressive sublanguage of OWL, is a description logic “without algorithms allowing for efficient inference and query answering over knowledge bases (KB) scaled to millions of facts (triples) [193].

A popular alternative to description logics is logical programming, which usually allows for more efficient inferencing and scaling of the underlying knowledge base. Grosz et al introduced OWL DLP as an intersection of OWL DL and logical programming [123]. This type of work aims to find a system of knowledge representation that can deal with billions of statements in an efficient way. Figure 11 illustrates the complexity of different OWL dialects and their affinity to logical programming and description logic.⁸

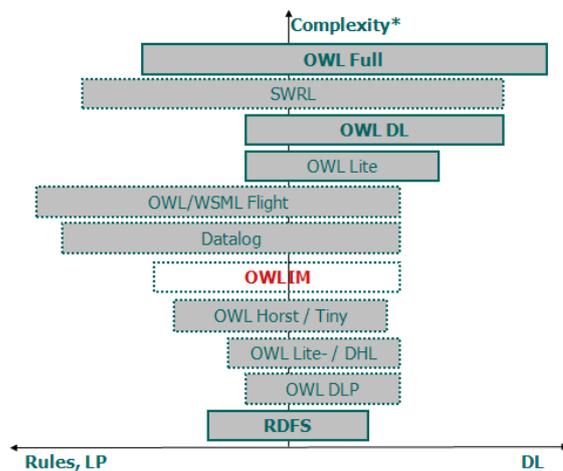


Figure 11: Complexity and relation to Rules, LP and DL of different OWL dialects.

The model theoretic semantics that have been elaborated for RDF(S) can be directly connected to real-world applications. To that end, a set of axiomatic statements and additional entailment rules must be defined. For example, table 1 shows an entailment rule for the subclass relation that is defined in RDFS. These

⁸The figure has been taken from [193].

syntactical rules can be used to implement reasoners for semantic stores that make use of certain reasoning strategies like forward chaining or backward chaining.⁹

Table 1: Entailment rule *rdfs11* that covers the subclass relationship in RDFS.

rdfs11	uuu rdfs:subClassOf vvv . vvv rdfs:subClassOf xxx .	uuu rdfs:subClassOf xxx .
--------	--	---------------------------

The semantic stores that use forward chaining can derive new facts by repetitively applying entailment rules to existing facts. By materializing the full closure of a knowledge base, queries can be answered very fast. However, the overhead for loading and changing the knowledge base is very high. This is because the closure must be recalculated every time facts are added or changed. Backward chaining starts with a fact to be proved or a query to be answered. The algorithm terminates directly if the fact is present in the knowledge base. If it is not, the algorithm starts searching for rules that can prove the fact and adds supporting facts to a tree of goals that also need to be proved. The algorithm stops if the fact that needs to be proved can be traced back, through the application of entailment rules, to existing facts. Backward chaining uses less memory, and the loading / changing of facts is very fast. However, certain queries may require a very long computing time.

It has been emphasized that information sharing is an important Semantic Web topic. It relies on information modeling and calculating. In fact, it has been proposed that the information represented according to Semantic Web concepts is easy to integrate. In this setting, information is integrated through reasoning. One precondition for Semantic Web information integration is that the information from different cultural heritage information systems must be available in a Semantic Web language like RDF(S) or OWL.

Afterwards, schematic heterogeneity can be resolved by adding facts to the knowledge base that make explicit how different schematic elements relate. These semantic relations can either be modeled in a bidirectional manner or in relation to a shared ontology. However, a rather difficult task is the automatic discovery of semantic correspondences between RDF resources that stem from different information systems. This is relevant for both schema elements and instances that are organized according to a schema. These challenges of semantic information integration will be dealt with in more detail later.

Calculating with information is a major feature that Semantic Web technology strives to support. In this section, the foundations of data modeling according

⁹Refer to [133] for a more comprehensive list of RDF entailment rules.

to the Semantic Web's standards and recommendations are introduced, followed by a discussion of the semantic foundations. Without these semantic foundations, developers of semantic stores would be forced to implement reasoning algorithms intuitively, and the different semantic stores would then derive different results from a knowledge base. But the rigid formalisms governing Semantic Web technology also introduce an amount of complexity that results in a steep learning curve.

4.2 Semantic Web Information Integration

Information sharing is a central concern of Semantic Web research. One of the most important tasks of information integration is to model semantic correspondences between (meta)data elements of different information systems. Thus, Semantic Web languages provide constructs that allow for explicit modeling of these semantic correspondences. Semantic stores can use their inference capabilities to jointly process information from different sources by processing these constructs. However, inferencing capabilities, which are needed to discover entity descriptions that refer to the same entity in the world, are not part of the proposed concepts. This section introduces the means for semantic information integration according to the Semantic Web and also refers to the shortcomings of this approach.

Semantic Web information integration relies on data that has been represented in RDF. Therefore, all information systems need to provide their data as RDF and its extending languages. Different approaches can be used to expose internal data structures as RDF data. Special mapping software has been developed to map a relational database onto an RDF graph. XSL Transformations can be used to transform an XML document to RDF. But in many cases, more complex approaches that make use of a higher programming language seem to be more adequate because they provide better means to manipulate data. For example, this is relevant in cases where data lacks explicit structure and needs to be extracted.

Even after data is transformed into RDF, it is still expressed according to an ontology that represents the domain vocabulary of an information system. If other information systems do not know the meaning of the classes and their relations, they cannot make use of information that has been published in this way. Hebel et al. [135] refer to this as the data source ontology. Since information systems manage information that is potentially interesting and useful to a broader audience, overarching ontologies have been crafted in many communities. The process of finding and documenting semantic correspondences between data source ontologies and / or domain ontologies is called ontology alignment. Ontology matching refers to methods that are suited to automatically discover semantic correspondences between elements of different ontologies.

Whether semantic correspondences have been discovered manually or with the

help of automatic methods, they need to be documented in a way that machines can use. This is why OWL provides properties as part of its language. The property `owl:equivalentClass` can be used to state that two URIs are referring to classes with exactly the same intension (the individuals that fall under the respective class description). The property `owl:equivalentProperty` can be used to state that two properties have the same property extension. Another example is `owl:subclassOf` to assert that the extension of one class is contained in the extension of another class. Furthermore, rules can be formulated in SWRL, RDF models can be manipulated by code, and even XSLT can be used to align the source ontology with the domain ontology (if the source ontologies have been serialized as RDF/XML).

Of course, ontology alignment is required for the subsequent steps of Semantic Web information integration. After the domain vocabularies have been aligned, the instances that are organized according to ontologies must be aligned as well. OWL has introduced several language elements that are relevant for the task of entity alignment. Many formal languages make the unique names assumption by presuming that different names refer to different things in the world. This assumption does not make sense in an environment that tries to deal with information that has been contributed by thousands of information systems. Many names will be used for the same thing in such an environment; therefore, the semantic of OWL does not enforce this assumption. However, several language constructs have been provided that allow for stating that different names refer to the same thing.

If two instances refer to an entity that is associated with a unique identifier, the property `owl:hasKey` can be used to make this circumstance explicit. The domain of this property is a class, and the range is a property that holds the key values for this class. If the value of this selected property is identical in two instances, they are treated as the same instance by semantic stores that implement the semantics of `owl:hasKey`.

Another set of language elements that enable the inference of the identity of instances are `owl:FunctionalProperty` and `owl:InverseFunctionalProperty`. If a property is a member of one of these classes, it can be defined as functional or inverse functional. Functional properties can only associate a unique value with an individual, so if this property is used more than once, the objects of the property are considered to be the same. For inverse functional properties, the object uniquely identifies the subject of the property. If the property is used with different URIs in the subject position and similar URIs in the object position, these subject URIs are considered as referring to the same entity.

In addition to these indirect means to make statements about the identity of instances, OWL provides the property `owl:sameAs`. This property can be used as a direct way to assert that two URIs refer to the same entity. Similarly, the

property `owl:differentFrom` and the class `owl:AllDifferent` can be used to explicitly state that two or more URIs do refer to different things.

The concepts that have been introduced above presuppose either shared, unique identifiers or explicit assertions of identity. If the instances brought together from more than one information system share some kind of unique identifier, a semantic store can use automatic deduction to resolve the identity of the two instances. If this is not the case, instances can be linked with the property `owl:sameAs` either by clerical work or with the aid of automated means. Methods and techniques that describe and implement the latter have been explored under the name entity matching.¹⁰

The problem of automatically matching different identifiers that refer to the same entity has been extensively studied in the database community. The developed methods and techniques make use of similarity measures, which determine the similarity of database values in the corresponding database fields. But similarity is a concept that introduces elements of vagueness, which is difficult for Semantic Web technology to deal with. Similarity metrics are used to assign degrees of similarities to pairs of entity descriptions. The description logics that form the foundation of Semantic Web languages cannot represent the concept of vagueness at this time.

Additionally, machine learning has been extensively used to determine the most relevant database fields for entity resolution. It is also used to decide whether two database records match or not. Most machine learning techniques implement different kinds of inductive reasoning, often based on the concept of entropy or other statistical models. The results of a set of similarity measurements form the input for these machine learning techniques. Most of these techniques learn by induction from previously flagged matches and non-matches. They make educated guesses on whether unknown pairs of instances match according to their similarity. Again, inductive reasoning on the basis of vague information has not yet been implemented for semantic stores that are used by the Semantic Web community.

This section introduces the features of RDF(S) and OWL that can make assertions about the identity of instances. Semantic stores can infer the identity of URIs by exploiting exact identifiers or by using facts which have been explicitly asserted (`owl:sameAs`). However, there are no means implemented to perform inductive inference, which would enable the identity coreference relations that are not explicitly stated. This is due to the fact that inductive inference for record linkage relies on concepts of vagueness (degree of similarity in contrast to binary decisions) and uncertainty (degree of belief, the real truth value is not known). Current Se-

¹⁰The process of resolving identifiers that refer to the same name has been given different names in different communities: record linkage, entity resolution, entity identification, coreference resolution, etc.

semantic Web implementations cannot deal with these concepts; this would require external tools to be added to the infrastructure. In some situations, the means to map discovered relations are inadequate. The instances modeled according to Semantic Web concepts refer to the entities in the world that are usually not persistent but change over time as well as in different situations (these are contextual effects).

4.3 Recent Developments and Linked Data

The amount and diversity of data published according to Semantic Web standards and recommendations is growing rapidly. Therefore, a community has been founded under the auspices of the World Wide Web Consortium (W3C): Linking Open Data. Berners-Lee came up with the idea to create semantic links between Semantic Web instances, which he uttered in a document laying out the Semantic Web roadmap [21] in 1998.

A set of principles has been formulated to guide institutions' efforts to share their data with a greater audience. The data published according to these principles can be considered part of the Semantic Web. The community project has been established, most importantly, to bootstrap the Semantic Web by providing it with large amounts of useful information. This initiative is closely connected to the database community, because both explore how to discover entity descriptions that refer to the same entity.

The term "Linked Data" has already been described as a way to exploit the infrastructure of the World Wide Web for the purposes of publishing, sharing and linking data. It makes extensive use of dereferenceable URIs to create links, and it envisions an architecture that is analogous to the WWW, but with data that has a much more explicit structure. Additionally, linked data connects related information in new and sometimes unanticipated ways.

Bizer, Berners-Lee and Heath [40] provide, along with a thorough introduction, a technical definition of the term: "Linked Data refers to data published on the Web in such a way that it is machine-readable, its meaning is explicitly defined, it is linked to other external data sets, and can in turn be linked to from external data sets." The relation between linked data and Semantic Web research, as well as its relevance for information integration, becomes obvious in this statement.

A few years earlier, Berners-Lee [23] reflected on a set of principles that may be helpful for publishing structured data on the Web:

- Use URIs to identify things.
- Use HTTP URIs so that these things can be referred to and looked up (dereferenced) by people and user agents.

- Provide useful information about the things when its URI is dereferenced, using standard formats such as RDF/XML.
- Include links to other, related URIs in the exposed data to improve discovery of other related information on the Web.

These four principles refer to methods, techniques and standards that have been developed in the Semantic Web community, but that still rely on the infrastructure that has been established for the Web. The usefulness of URIs for referring to things has already been discussed above. Therefore, the first principle recommends URIs for publishing structured data so that they can become part of the Semantic Web.

Additionally, the second principle of Linked Data recommends using HTTP URIs so that it can be dereferenced by user agents like common web browsers. But web servers can provide more information than HTML if the RDF data is transmitted to a caller. It is important to distinguish the two relationships that exist between the identifier and the identified: reference and access. While an HTTP URI does denote a thing in the world, it is also a way to access information that is hosted on a web server. Halpin and Hayes [134] point out this dichotomy and recommend keeping these two concepts separate.

The access mechanism of HTTP URIs is very important for the concept of Linked Data. The third principle deals with providing useful information about things that are identified by URIs. Since this data should be structured and machine-readable, it is helpful to use a standard format such as RDF/XML. Although user agents can use HTTP URIs to acquire additional information about an entity, they cannot influence how comprehensive the transmitted information will be. Thus, user agents may need to complete their knowledge by crawling additional URIs that are part of an entity description like Web bots of internet search engines.

The fourth principle partially results from the use of global HTTP URIs that can be dereferenced if the server is made responsible for the process of resolving. Regardless of how RDF/XML is represented in an information system (e.g., as simple text documents or as statements in a semantic store), a link to another information system can be established by using its URI scheme. This enables user agents to provide browsing functionality and crawlers to acquire semantic information. Both use cases require to explore a graph that is established by the extensive use of links. Local identifiers can still be used to refer to information in a single file or semantic store.

A certain amount of information has already been published according to these principles. The result is a very large network or graph that relates rather diverse information sources with each other: geographic locations, people, companies, books

should be put in a position to search and browse it. This is why search engines such as SWSE, which make explicit use of structured information, have been developed [143]. Browsers like Tabulator seek to provide users with a view of the underlying graph structure of data that has been represented as RDF [24]. Indices have also been created for Semantic Web agents who need to acquire more information about a certain resource. Projects like Watson provide these agents with APIs to gain better access to Semantic Web data [76].

While these use cases are still rather generic, smaller communities may want to develop more specialized applications that could benefit from structured and linked information. A use case for the domains of archaeology and classics was elaborated in chapter 2. Combining data that has been obtained from many sources, and using more than one online service to query and visualize the data is one possible application. It would be interesting to reflect on the more complex Semantic Web applications that provide functionality beyond searching and browsing. This could comprise reasoning on large amounts of information, complex graph analyses and visualizations.

Linking Open Data (LOD) is the most prominent community project for the Semantic Web. It was originally started to bootstrap the Semantic Web by providing the community with a sufficient amount of information for exploration. However, it seems that this project is the visible result of reevaluating the reasoning capabilities of semantic stores. Rather than focusing on the reasoning part of the Semantic Web, LOD provides structured data in a Web-like manner while making no recommendations on how the data should be processed.

The development of linked data also depends on link discovery and entity resolution. It was stated that this cannot be accomplished in a straightforward manner with Semantic Web technology. Link discovery has only recently become prominent in the context of evaluating information in social networks. Additionally, the entity resolution methods and techniques being explored in the database community need to be considered.

4.4 Flagship Projects

Shadbolt, Berners-Lee and Hall [217] emphasize the role of small communities, which feel a pressing need to apply Semantic Web technologies. Since 2006, many projects have started to experiment with the Semantic Web recommendations and tools that were developed under the auspices of the W3C. The Linking Open Data project is certainly one of the most important activities in the field of Semantic Web research. However, other projects, or flagship projects, are also reaching relevant communities and drawing more attention to the Semantic Web in general.

This section introduces a non-representative selection of projects that emphasize the role of Semantic Web technologies and show the progress of Semantic Web

information integration: DBpedia extracts structured information from Wikipedia and forms a major hub for the Linked data project [41]. The SIMILE project, which is hosted at the Massachusetts Institute of Technology, has been working on a suite of tools that should help to manage and use “semantic” data [175]. Finally, Europeana Connect is exploring semantic technologies for the integration of cultural heritage sources [11].

In 2007 a collaboration between the University of Leipzig, Freie Universität Berlin and Open Link Software was started to acquire structured data from Wikipedia. Since then the DBpedia project has been harvesting structured information from Wikipedia, mainly by evaluating the so-called *Infoboxes*. These formalized sections in Wikipedia hold structured information. They have been annotated by users as tables that can be easily extracted and represented in RDF. Since Wikipedia links pages in multiple national languages on the same topic, these Infoboxes attach multilingual information to a URI. This will certainly result in useful, multilingual vocabularies as well.

DBpedia publishes its information according to the principles of Linked Data. Therefore, its information is guaranteed to be aligned with other information from different institutions. Additionally, the project has implemented an API that allows for querying the data with SPARQL, the standard query language and protocol for RDF graphs. The emphasis of this project does not lie on processing acquired information, but rather on providing a large amount of information and creating as many useful links as possible.

SIMILE¹² is a research project hosted at MIT in collaboration with the W3C. The project is developing a set of tools that should “make it easier to wander from collection to collection and, more generally, to find your way around in the Semantic Web[175].” Although this definition does not mention the Semantic Web, its relation to Semantic Web technology is obvious. While DBpedia has become a major hub for the Linked Data community project and strives to deal with huge amounts of information, the tools that have been developed within the SIMILE project focus on real applications and user interaction.

Because information processing in this setting is more complex, the developed software is inefficient for larger amounts of information. While different approaches for managing and visualizing information are explored, no project is considering complex semantic inferencing. For example, the projects Longwell and Exhibit provide faceted browsing on RDF data and create mashups of different information sources. Timeline is a tool that can be used to visualize events over time. Fresnel is a project that works on a vocabulary that can be used to specify how RDF data should be presented to users in a non-generic way.

¹²Simile is an acronym for “Semantic Interoperability of Metadata and Information in unLike Environments”.

DBpedia draws structured information from Wikipedia that is domain-independent. Another project that is currently striving to aggregate a large amount of information from different institutions is Europeana [101]. Europeana’s project entitled Europeana Connect is trying to establish a “semantic layer” for the integrated content [11]. It strives to develop a federated repository of vocabularies, a semantic user interface and explores advanced semantic processing of content. Another project called the Semantic Search Lab is working on a semantic search engine that allows the user to submit complex queries on data that has been integrated from multiple cultural heritage institutions [100]. The research prototype uses semantic autocompletion for full-text searching. It is backed by a set of controlled vocabularies that are specialized vocabularies in the cultural heritage domain.

Without doubt, the Semantic Web attracts a large number of projects, both in the research and business communities. Most of these projects strive to create and integrate large amounts of data for communities that are acquainted with the Semantic Web. Searching, browsing and visualization have a high priority for most of the described projects. However, it seems that reasoning capabilities for Semantic Web data have not been considerably used or elaborated. This could be due to the involved complexity of the components that these projects must use.

4.5 Semantic Web Challenges

In their visionary work, Berners-Lee, Hendler and Lasilla [27] describe autonomous, intelligent agents that support humans in everyday activities. These intelligent agents need to draw information from different sources, perform autonomous reasoning and influence the environment by acting on behalf of human counterparts. The authors argue that the recommendations and standards proposed by the W3C provide the tools to do just that. A couple of years later, Shadbolt, Berners-Lee and Hall [217] recognized that this vision had not come true and, in fact, may be far from being realized. It seems that the Semantic Web vision, as it was laid out in 2001, is too ambitious, and recent reframings of possible uses of Semantic Web technologies are trying to attenuate high expectations. This section gives an overview of opportunities and challenges that are identified in the context of Semantic Web research.

The Semantic Web in general and the Linking Open Data community project in particular rely on the Web’s architecture because this architecture is easier for larger communities to publish structured information. Every institution that hosts a public web server and every individual that publishes via a web server can expose information on the World Wide Web. The number of web pages that have been indexed by search engines is estimated to be several billion.¹³ One can imagine the

¹³A method of estimating the size of the Web has been suggested by Kunder [80].

amount of structured information that would become available if only a smaller subset of these pages were enriched with structured data, for example, in RDF.

The problem of efficiency is related to, but not identical with, the challenge of considering huge amounts of information. Not only is the number of data elements huge but the reasoning problems to be solved are intrinsically hard. For example, in the context of Semantic Web research, RDFS is a knowledge representation language with the least expressivity and only provides limited means to form a concept hierarchy. The latter feature is viewed as a very foundational element for using shared vocabularies. Yet many still report that the construction of a semantic repository that compares to a relational database in terms of its performance and scalability is a very challenging task.[193].

The information humans use on a daily basis is either vague or uncertain. For example, the term “young” can be better expressed with a logic that allows for modeling degrees of truth. This means a one-year-old can certainly be considered young, but the degree of truth may only be 0.7 for a 30-year-old. It is often helpful to express uncertainties with knowledge representation formalisms. For example, if an archaeological information system dates an artifact in the Augustan era, the degree of certainty would be 0.6.

However, the different interpretations of probability that are currently being discussed have resulted in different implementations of the concept. As mentioned, the formal languages of the Semantic Web are rooted in first-order predicate logic, which defines predicates as functions that either map onto “true” or “false” truth values. Lukasiewicz and Straccia [170] provide a thorough overview of research that is trying to extend Semantic Web concepts to cover the concept of vagueness and uncertainty. They observe that relevant implementations of these logics have efficiency problems as well.

The Linking Open Data Project is a community project that aims to generate a global graph of information without being led by a global authority. Because the data providers are independent of each other, inconsistent and contradicting information will most likely be linked. Some providers could even intentionally publish erroneous content.¹⁴ The process of data fusion, an important aspect of information integration in the database community, is also significant in the context of the Semantic Web. The deductive reasoning that has been implemented in semantic stores is unable to deal with contradicting information.

Semantic Web technology is complex and interdisciplinary. The challenges that are associated with the introduction of formality for information systems have already been discussed. Sophisticated means of knowledge organization like the CIDOC CRM reach a complexity that requires dedicated training. Isaksen [149]

¹⁴For example, many lobbyists have manipulated Wikipedia entries [29]. It has been illustrated how this information becomes part of the Semantic Web.

refers to some problems of implementing the CIDOC CRM. Information integration projects should take this into consideration and allocate sufficient resources.

Marshall and Shipman [173] have elaborated their idea of harmful formality for the concepts of the Semantic Web. They observe that the term “Semantic Web” has been used to describe many different things: a universal digital library, a backbone for the work of computational agents and a federated knowledge base. The authors conclude that methods from information retrieval are probably better suited to establish a universal library than Semantic Web technology. Using the Semantic Web as a backbone for the work of agents seems to be problematic for both pragmatic and theoretic reasons. The approach that seems to be the most correct is the use of Semantic Web technologies to create a federated knowledge base. But, in this context, federated means that data is generated and linked with a certain type of application in mind.

Federated knowledge bases that rely on Semantic Web technology use RDF and its extensions as a foundation. However, this is problematic because traditional formal logic dominates Semantic Web technology, and this type of logic only allows for axiomatic reasoning. Complex tasks like resolving different representations of the same entity require inductive reasoning because it considers vagueness and uncertainty. Although these topics are currently being explored and developed, they still do not allow for proper information integration. Additional tools still need to be developed that can implicitly provide the needed reasoning capabilities. To that end, methods and techniques from research in artificial intelligence, data mining and information retrieval have been adopted.

Although Antoniou and van Harmelen [6] try to dispel reservations about the Semantic Web, they emphasize that partial solutions should be pursued. The Semantic Web is often interpreted as an all-in-one device suitable for every purpose with intelligent agents, which exhibit human-like reasoning. But it is unrealistic to instantly solve artificial intelligence problems, which have a long research tradition in artificial intelligence under a different notion.

While the Semantic Web provides a formal language which is expressive enough to deal with most of the popular data models, the inherited problems, which come from drawing concepts and techniques from other disciplines, remain. In these cases, a deep understanding of the involved computational complexity is helpful. In summary, system developers should be aware of the limitations in expressiveness of vague and uncertain information, and they should have the appropriate tools at hand to carry out feasibility studies for certain user scenarios.

4.6 Knowledge Representation Alternatives

It seems that semantic inference by formal deduction has a rather limited set of application areas. Furthermore, the problems associated with computational com-

plexity are observed in artificial intelligence – issues which were already anticipated in the early stages of the Semantic Web. Thus, the development of Semantic Web languages focuses on expressive power in favor of other goals like efficient reasoning capabilities. Additionally, the vision of computational agents as agents who act autonomously in a complex environment on behalf of their human counterparts seems to be unrealistic for both theoretic and pragmatic reasons.

However, Semantic Web technology could become the foundation of federated knowledge bases due to its popularity. Such knowledge bases must bring together related information from many sources because they depend on information integration to match schemas and entities. As argued, the current reasoning capabilities of Semantic Web tools are not sufficient for the task of entity resolution. Therefore, this section introduces a number of proposed alternatives for the sophisticated processing of Semantic Web information.

The various challenges of Semantic Web technology was discussed in the last section. In many situations only vague or uncertain information is available for processing and automated inference. For example, time spans are not referred to by their exact data but by qualitative values like “in the beginning of”. Algorithms that have been used for entity resolution determine similarities and make extensive use of machine learning for inductive inference. Thus, the extensions and alternatives that allow for representing and processing vague and uncertain information with Semantic Web technology have been explored.

Sheth, Ramakrishnan and Thomas [221] also observed how the current limitations of Semantic Web concepts emerge from the limitations of first order logic. Furthermore, most mechanisms for knowledge representation only allow for limited expressiveness to achieve acceptable computational characteristics. This means most mechanisms cannot represent imprecise, uncertain and approximate knowledge. The authors have systematized computational semantics as three categories according to their uses: implicit, formal and powerful.

The implicit semantics found in textual information or even as structured data without any documentation are not represented in any machine-processable syntax. It is subject to information extraction by algorithms that create more structured information probably in the form of *formal semantics*. These are represented in a well-formed syntactic form that can be interpreted according to model theoretic semantics and that follow the principle of compositionally. But once the information has been formalized in the aforementioned way, processing capabilities are limited by the limitations of description logic. Many applications such as question answering systems need more expressivity to work properly. This involves exploiting fuzzy and probabilistic representations of information. The authors recommend focusing research on these extensions of classical description logic and exploring their computational properties (*powerful / soft semantics*).

One method of information processing based on the theory of probability is statistical inference. It can use data that has been drawn from this population by (random) sampling (some archaeological artifacts in a database) to hypothesize on whole populations (all archaeological artifacts in a database system). Of course, inductive reasoning capabilities could also be applied to data that has been represented as RDF. Statistical inference can deal with data that is incomplete and subject to errors or other variations. The results of statistical inference can only be expressed as probability distributions, not as symbolic values. Again, it is not easy to express these probabilities as RDF.

Statistical inference uses the statistical models that are represented as sets of mathematical equations. For example, linear regression models use the least squares method to find systematic patterns in the data. Here, features of the unknown parameters are estimated from the data using linear functions. The conclusion of this process is called a statistical proposition. Several forms of statistical inference in combination with other methods and techniques will be used in the course of this dissertation project. In order to predict links between database records, links between a sample of the whole database can be used to hypothesize about links that have not yet been discovered.

Intelligent agent research has traditionally been biased towards symbolic knowledge representation. At the same time, other models like artificial neural networks seem to process information in analogy to the human brain. Gärdenfors [113] suggests that it would be more productive to consider the ways that humans handle concepts. Rich implicit semantics are typically hardwired as computer programs by software developers, but the Semantic Web makes the mistake of reducing explicit semantic content to first order logic and set theory.

Although humans categorize objects based on their similarity to other objects, similarities cannot be easily processed with current Semantic Web tools. Therefore, beneath the symbolic and connectionist approaches for information representation, Gärdenfors proposes a model that he terms *conceptual information representation*. This model allows information to be represented as geometric structures in Euclidean space. Methods and techniques like textual statistics and Latent Semantic Analysis, which have already been explored in the field of information retrieval, make extensive use of vector space models.

The shortcomings of current Semantic Web tools and concepts are due to its focus on description logics. Although OWL implements certain kinds of modality, it cannot deal with vagueness and uncertainty. Stoilos et al. [239] observe that the importance of uncertainty will increase in the future for Semantic Web tools and concepts. The authors present an extension of OWL that allows for representing uncertainty. The international conferences on the Semantic Web hold

recurrent workshops on uncertainty reasoning.¹⁵ Additionally, the more traditional approaches developed in the fields of computational linguistics, information retrieval and information extraction have been considered for Semantic Web research.

In summary, the main problem with Semantic Web reasoning is its reliance on description logics. It has a well-defined syntax and semantic, which allow for guided interpretations of the represented information. A Semantic Web system, which strives for information integration, can only rely on deductive reasoning. However, certain information integration tasks must perform inductive reasoning. Using humans to discover different references to equal entities is tedious, resource intensive and must be supported by automated methods. Although different projects try to implement these methods as Semantic Web tools, they are not yet a foundational part of Semantic Web technology. Thus, current Semantic Web reasoning tools play a subordinate role in ALAP and statistical reasoning is favored.

¹⁵More information on these workshops can be found at Staab and Sure-Vetter [235].

5 The CIDOC CRM

Semantic Web concepts mainly rely on description logic. A fundamental idea of description logic is to imitate the human tendency of using categories for describing things. Therefore, many propose using ontologies to form a foundation for description logic so that things can be easily and formally described. Different ontologies are introduced in different communities. These allow for describing the things which are prominent in these communities. In the field of classics and archaeological research, one particular ontology is popular: the CIDOC Conceptual Reference Model. Although it is introduced as a conceptual reference model, its structure bears all features of an ontology in the sense of knowledge representation.

A couple of the projects described in the previous sections explore the CIDOC CRM for information integration. For example, the Berlin Sculpture Network plans to publish a large amount of information according to the CIDOC CRM. The CRM is based on *events*, which are introduced to facilitate better information integration. Therefore, the CIDOC CRM is important for information integration research in the fields of classics and archaeology. This chapter introduces how CIDOC CRM is structured and highlights some of the relevant design concepts. However, the CRM leaves considerable room for interpretation when it comes to implementing concrete applications. Therefore, this section mentions best practices that are elaborated in current and finished projects.

It is argued that Semantic Web tools are still not well suited for discovering entity descriptions that refer to the same entity. Additionally, past projects have assessed the CIDOC CRM as being too complex, causing some of them to turn to more concise vocabularies. However, published experience is also available for projects that have explored the CIDOC CRM for information integration. Interesting ideas are introduced by the CRM, and should be considered for further semantic information integration research. Therefore, CRM concepts and their relation to Semantic Web technology are discussed in the following sections. The practical applicability is evaluated and deviations from the original concepts are evaluated for usefulness.

5.1 General Structure of the CRM

The CIDOC CRM provides an ontological structure to represent the cultural heritage domain. It is modeled in a rather abstract way: it has a wide applicability and can be considered a so-called top-level ontology. Although the CRM originates in the museum community, it is also applied to other related domains with similar or shared material objects. Therefore, the CIDOC CRM is also used in the domains of classics in general and archaeological research in particular. This section briefly introduces the general structure of the CRM and discusses its relation to

Semantic Web research. Although it is introduced as a reference model, a set of actual implementations has been published by different projects.

Reference models convey a basic idea of how objects in a certain system are arranged and interrelated. These fundamental paradigms are technology-agnostic but can – for example in software technology – be used to derive standards. The CIDOC Conceptual Reference Model is such an abstract model that has been crafted to support the exchange of cultural heritage objects. Over a period of more than ten years, the CIDOC Documentation Standards Group has been developing the CRM, which was accepted as the official standard ISO 21127:2006 [71]. Its main purpose is to establish interoperability by providing the cultural heritage community with means to document the meaning of shared information. Therefore, it comprises explicit definitions that have been arranged as a structured vocabulary. This should enable software developers to craft software that can deal with data that has been annotated with CRM definitions.

According to these definitions, a domain of discourse can be described for the field of cultural heritage. The domain comprises actors that participate in events affecting or referring to conceptual or physical objects. These events take place within certain time spans, and the objects can be associated with location information. Additional means have been provided to name or identify the elements of this structure and to introduce refinements of the basic vocabulary. Figure 13 illustrates the CRM from a birds-eye perspective.¹⁶ The classes and relations that are shown in the figure can be further refined by subclasses and subproperties. Technically speaking, the CIDOC CRM is a hierarchy of 90 classes defining concepts that are commonly used in museum documentation practice. Each class describes a set of objects that share common features, and 148 *properties* define semantic relations between these conceptual classes.

Using classes and properties to define sets of objects with shared features and their relations to each other is reminiscent of knowledge representation in description logic. By referring to these concepts of formal semantics, the CIDOC CRM is well prepared to play a role in the development of cultural heritage on the Semantic Web. But since the CRM is a reference model, it does not specify the peculiarities of particular implementations. Thus, there are many alternatives for providing standards, and various formats representing information about museum objects according to the CRM have already been described. For example, Oischinger, Schiemann and Görz [192] describe the Erlangen CRM, an implementation of the CIDOC CRM with Semantic Web concepts. Erlangen CRM uses OWL-DL as a formal language to express the classes and properties of the CRM. Some elements of the reference model cannot be implemented in a straightforward way due to the limitations and differences in OWL DL. Therefore, the reference model needs to

¹⁶The figure follows Dörr [86].

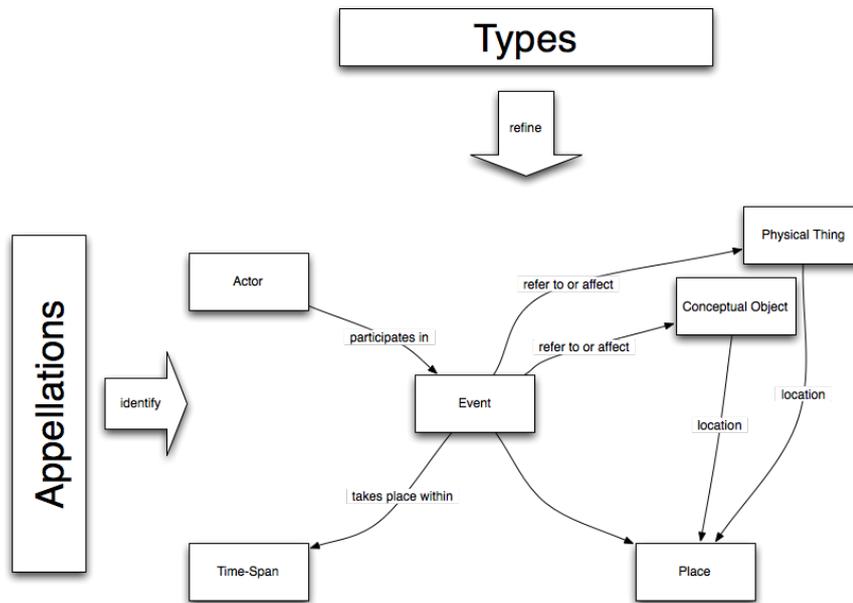


Figure 13: The General Structure of the CIDOC CRM.

be expressed through description logic, which approximates the intended meaning.

The CIDOC CRM is a state-of-the-art model for representing the structure of cultural heritage in information systems. The design concept of the CRM bears a strong affinity to formal methods in the field of artificial intelligence and – of course – the Semantic Web. By adopting these methods and techniques, projects using the CRM in this way need to deal with the challenges of formal information representation. Furthermore, the CRM is constructed as a top-level ontology for the domain of cultural heritage. Therefore, communities which strive to use the CRM must extend the vocabulary of the CRM to suit their needs.¹⁷ The CRM helps participants of information integration projects agree on concepts on a rather fundamental level, and it also helps them to share additional vocabularies.

5.2 Information Integration and the CRM

The previous sections emphasized the importance of information integration, and how it is one of the primary reasons driving the development of Semantic Web technology. Since the CIDOC CRM makes extensive use of Semantic Web concepts, its focus on information sharing and automated reasoning is evident. Some peculiarities of the CIDOC CRM are interesting for information integration beyond

¹⁷See, for example, the extension for archaeological excavation and analysis processes [147].

its affinity to the Semantic Web. Like every ontology which has been formalized with Semantic Web concepts, it can be extended by using the language elements of RDF(S) and OWL. In addition, the CRM introduces a “metaclass” to represent so-called “types”, which help create auxiliary controlled vocabularies. In particular, CRM introduces the notion of temporal entities, which allows the contents of an information system to be reified. These aspects, which are also related to information integration, will be discussed in the following.

As mentioned, CIDOC CRM is a top-level ontology. Thus, its expressivity will probably not be satisfactory for certain communities. For example, the CIDOC CRM defines a class for representing physical objects that have been intentionally created by humans. However, the CRM does not provide means to distinguish a sculpture from a building. Single communities can create and publish their own extensions to the CRM in these cases. Of course, RDFS allows for the creation of subclasses by using the property `rdfs:subClassOf` on the more general classes. But the CRM provides another mechanism for specialization by introducing a class that explicitly describes concepts from controlled vocabularies. Relations between broader and narrower terms can be modeled between instances of type classes. But communities should be aware that these extension mechanisms introduce concepts that are not part of the standard, and that need to be agreed upon and published.

One of the design principles of the CIDOC CRM is that it supports event-centric documentation. Figure 13 illustrates the important role of the notion of temporal entities or events in the CRM concepts. The class “Activity” usually describes actions that have been carried out intentionally: “This class comprises actions intentionally carried out by instances of E39 Actor that result in changes of state in the cultural, social, or physical systems documented [71].” Doerr and Kritsotaki reflect on the role of events in metadata formats in general and the CRM in particular [84]. They emphasize that the use of events in documentation of material cultural heritage leads to a more straightforward documentation practice. This is due to the fact that history can be interpreted as a sequence of events where people, material and immaterial objects “meet” at certain places to interact. Additionally, the authors state that structures not following the event-centric paradigm make meaningful information integration difficult.

Figure 14 picks up the user scenario elaborated above. Although it abstracts from its intrinsic complexity, it demonstrates how the concept of an event can align data of Perseus and iDAI.field. The event “Extension of Pergamum” can relate information from Strabo’s text to the documentation of archaeological field work. Entities are often related with respect to a certain context (the extension of Pergamum) that is made explicit by using events.

Bits of information extracted from each system can then be explicitly linked and canonical names can be assigned. But the example also emphasizes the Achilles

heel of this approach. Recording cultural heritage objects along with their history in a formal way would introduce a lot of overhead both for data generation and processing. Thus, there are situations where it is helpful to keep some information implicit or unstructured in the data and leave the interpretation to humans or special software components.

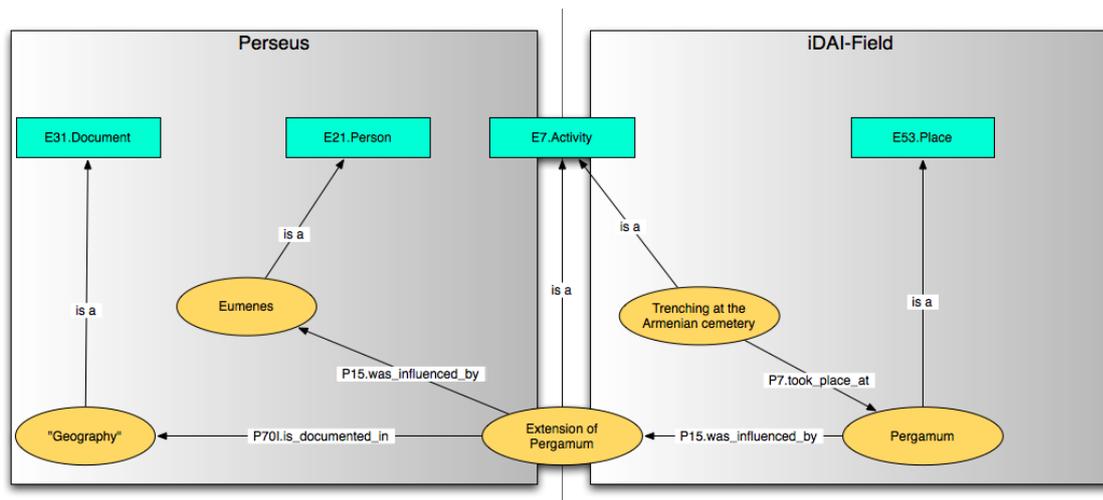


Figure 14: The (reduced) scenario as linked information objects.

It has been mentioned that data fusion is an important step in the process of information integration. Information that has been extracted from different information systems may not always be complementary. Often, more than one source will provide assertions on a certain object feature that may be complementary or conflicting. The data fusion component should decide whether a certain chunk of information is accepted or not. The event mechanism of the CRM may allow for the representation of conflicting and imprecise information if it makes the process of attribute assignments explicit. An attribute assignment can be associated with an actor and a place to model the provenance of information. Recording the provenance of information is important in Semantic Web research. Belhajjame et al. [18] introduce the PROV data model for representing provenance information (with Semantic Web concepts).

Can the CRM concepts be used to describe the process of finding links between descriptions of entities? And is event-centric modeling helpful for describing semi-automatic linking processes? Semi-automated linking is usually prepared by inductive reasoning with machine learning and additional specialized algorithms and methods. It has been argued that these reasoning methods result in vague and uncertain information that cannot be easily modeled using description logic. According to the CRM definition, “[A]ctions of making assertions about proper-

ties of an object or any relation between two items or concepts” can be modeled as an attribute assignment [71]. The CRM provides the vocabulary to describe pieces of documentation and could be extended to describe links between entity descriptions that refer to the same entity. However, it seems that this would go beyond the intended scope of the CRM because there are no means of expressing this relation in a straightforward manner.

The CIDOC CRM has been elaborated with the explicit aim of fostering information integration in digital cultural heritage communities. By describing an extensible, top-level ontology, it helps cultural heritage information systems to agree on a shared vocabulary for organizing information. The CRM has a typing mechanism, which helps to construct and publish shared, controlled vocabularies. Furthermore, events should facilitate explicit documentation and help to track the provenance of information. The CRM does not focus on representing the results of (semi-)automated entity resolution. It remains to be determined whether it can be extended in a straightforward manner. Therefore, the introduction of additional vocabulary elements for this purpose could be useful and is discussed later. However, the problems of representing vagueness and uncertainty remain because the CRM has a strong affinity to classical description logic.

5.3 Implementation Considerations and Applications

The CIDOC CRM is published as textual definitions which describe the general structure and the meaning of classes and properties. It is mentioned that it has a strong affinity to knowledge representation research and, therefore, to Semantic Web concepts. However, these definitions cannot be implemented with Semantic Web concepts in a straightforward manner. Knowledge representation languages like RDF(S) and the different dialects of OWL may require subtle deviations from the standard. The result is technical interoperability and should be considered in those projects which need to share information. In this section, certain projects are mentioned and the considerations for the implementation of the standard are discussed.

Many projects will need to extend the CRM for their own purposes. Binding, May and Tudhope [37] describe an extension of the CRM to model information about archaeological excavations. The authors also describe a terminology service that allows for querying controlled vocabularies. This service can be used to query for concepts within a vocabulary or to expand queries. The extension mechanism has also been used in the course of CLAROS [17]. A Wiki has been set up so that project partners can document how the CRM has been extended and implemented.¹⁸ The abstractness of CRM concepts could also foster interdisciplinary

¹⁸The Wiki is public and can be found at “<http://www.clarosnet.org/wiki/index.php>”.

communication processes. Lampe, Rieder and Dörr describe ways of using the CRM for transdisciplinary information integration. [165]

Nussbaumer and Haslhofer [189] describe how similar issues have been encountered in the BRICKS project. Since implementation issues have been abstracted by the CRM, technical heterogeneity could become problematic if different technical means are used for implementation. For example, Hohmann and Scholz [144] recognize that the CRM class “E59 Primitive Value” cannot be implemented in a straightforward way with Semantic Web technology. And because the CRM has been introduced as a top-level ontology, only rather abstract concepts have been defined. Therefore, there are multiple ways to map schemas onto the CRM, which can lead to additional overhead communication.

The problems that may arise with the introduction of formality in information systems have already been discussed. Problems of conveying additional formality to end users have also been described for BRICKS. Sophisticated user interfaces that provide means for efficient searching and browsing needed to be developed. Prior efforts of integrating information for Arachne and Perseus had to cope with additional structure that was imposed by the CRM [163]. For example, the CRM required information about events to be made explicit, even if it was only implicitly available in information systems.

A special interest group has formed that comprises members working with the CIDOC CRM in different projects (CRM SIG) [88]. The CRM SIG is a space where concerns of further development for the CRM are discussed. The Institute of Computer Science (ICS) of the Foundation for Research and Technology - Hellas (FORTH) also maintains a list of projects that have been exploring the CIDOC CRM [60]. These projects either use the CRM to assist in modeling decisions or they strive to implement the standard (exactly). The scope of projects range from information provision and integration to exchange format development.¹⁹

A reference model like the CIDOC CRM can describe top-level entities and their relations in certain domains. Thus, specialized models must be developed by modifying and extending the CRM for actual application. The different challenges and their implications for projects that need to implement the CRM have been mentioned above. The CRM concepts cannot be implemented with Semantic Web technologies in a straightforward manner. Syntactic and semantic heterogeneity is introduced during implementation because different projects agree on different implementations of the CRM. Therefore, modeling decisions and best practices should also be published, which would help to overcome these issues.

A central concern of the CRM is to help communities with sharing information. It is constructed as a reference model, which is structured like an ontology. Thus, application specific schemas can be mapped onto the CRM to facilitate better

¹⁹For example LIDO, an exchange format in a form that is compatible to CIDOC CRM [61].

information integration. However, different technical implementations of the model lead to technical heterogeneity, and this needs to be resolved. Additionally, by organizing entities according to the CRM, one does not directly support the task of entity resolution. Rather, if the introduction of a shared ontology is required for entity resolution, this must be satisfied beforehand. But entity resolution may make use of explicit structures that are imposed by Semantic Web technology.

6 Entity Resolution as Knowledge Discovery

The previous sections have emphasized how explicitly annotated information is usually not available for the purpose of finding entity descriptions that refer to the same entity in the world. Therefore, implicit meaning should be discovered by analyzing the information that is already available. But the amount of information that must be considered is too large for human processing. It would be helpful to automatize at least parts of the process. One aspect of knowledge discovery is analytical data mining. This is a field of computer science which explores means to automatically extract meaningful patterns from large amounts of data. A typical pattern would be the classification of entity descriptions into sets of matching and non-matching descriptions.

Chapman et al. [219] published best practices for defining a data mining process as the Cross Industry Standard Process for Data Mining (CRISP-DM). CRISP-DM is the leading methodology in the field of Data Mining and has been backed by several polls in the Data Mining community. The document describes a Data Mining process that consists of six steps: business understanding, data understanding, data preparation, modeling, evaluation and deployment.²⁰ The following paragraphs will describe the role of each step for entity resolution.

The Data Mining process starts with activities that help with understanding the project objectives and documenting requirements. Information integration projects should be aware of the purpose of information integration. It is important to understand how end-users will benefit from information that has been integrated from multiple information systems. If this is the case, only relevant information will be integrated, and the overall effort that needs to be put into matching and mapping will be reduced.

Another important step is to collect data from different information systems and to understand it. This step includes the activity of exploratory data mining, which allows the software developers to become familiar with the data. In the course of exploratory data analysis, summary statistics can be used to unveil the content and form data in an information system. One important outcome for information integration projects is the amount of possible overlap that indicates the number of links between data sets. But the discovery of unresolved data quality problems is important for enhancing the later data mining steps.

The next step, data preparation, models data into a shape that can be used by data mining tools for further analysis. For example, this involves selecting the relevant data elements and dealing with data quality problems. For the task of information integration and entity resolution, this step also involves computing

²⁰Traditionally, the process of Data Mining has been described as consisting of only three steps: pre-processing, data mining and validation of results.

the similarities between entity descriptions that have been selected and extracted from different information systems. In many situations the available data does not have appropriate granular structure for certain types of analysis. In these situations information retrieval models turn out to be helpful because they can analyze information that has been extracted from a large number of semi-structured text documents.

Another problem is that the number of comparisons is usually too high to be calculated in an efficient manner. Therefore, the entity descriptions that have been contributed by multiple information systems need to be preprocessed. By preprocessing entity descriptions, only the descriptions that have some preliminary similarity are considered in the comparison. This can be accomplished by applying clustering techniques to the data. Although clustering is introduced as part of data preparation, it can be viewed as a whole data mining workflow in itself. In this case, the whole data mining process needs to be repeated in a recursive manner for one activity of the entity resolution process.

Once the data has been understood and prepared, it can be fed to different data mining models. This is the central activity of the whole process: to calibrate and apply data mining models to the data to generate useful results. An iterative process of model calibration could be necessary because many machine learning algorithms behave very differently for various parameter settings. For example, the method of canopy clustering provides two parameters that determine the members of the same cluster. To produce useful results, these parameters should be carefully calibrated.

It is helpful to review the preliminary steps and evaluate the results that have been produced by the chosen and calibrated model. However, a thorough evaluation of the entity resolution results turns out to be quite complex. This is a classical chicken and egg problem: to evaluate the results of the entity resolution process, one needs a reference corpus of resolved entities. Since the compilation of such a corpus would be related to a certain domain, it would be difficult to generalize. It is necessary to have a random set evaluated by domain experts to make an assessment of the quality of the data integration process. These problems will be dealt with in more detail later.

The results of the modeling step are usually not comprehensible and therefore not instantly useful for end-users. Therefore, these results need to be worked up so that they can be effectively communicated. An obvious step would be to use the evaluated results of the entity resolution process to refine the machine learning models. Additionally, the information can be used to fuse different object descriptions and to bind identifiers to a canonical form. Furthermore, adequate user interfaces that support the process of evaluation and provide useful functionality to end-users need to be designed and implemented.

The CRISP-DM process has been adopted by a number of data mining projects in the industry. It should be adjusted to the needs of individual projects like ALAP. Additionally, the steps of the process cannot be worked through in a strictly serial manner. The information that is acquired at individual steps can be helpful for enhancing the preceding activities. In the case of information integration, single steps like data preparation may also make use of data mining models. Thus, the whole process must be repeated in a recursive manner for these activities.

This section illustrates the process of information integration and entity resolution in particular. Both processes are activities of knowledge discovery. The CRISP-DM process covers the relevant and useful steps rather well. Throughout this activity, a multitude of methods from various disciplines is used to prepare and analyze the data. The following sections introduce these methods in the context of their disciplines and point out their relevance for information integration and entity resolution in particular.

6.1 Information Retrieval

Information retrieval deals with the computer-aided search of complex information contents. To achieve this goal, information must be presented to users, stored in information systems, efficiently organized and accessed. Users formulate their information needs as queries, which can then be processed by an information retrieval system (e.g., a search engine). Some methods in the field of information retrieval are also helpful for the process of information integration. In addition to providing efficient access to information, certain retrieval models can be used to compare descriptions of entities. Information extraction is one specific task of information retrieval which strives to generate structured information from unstructured text; it is a preliminary step to prepare data for subsequent processing. This section introduces relevant models of information retrieval with respect to their role for information integration.

One very important problem, which information retrieval systems must address, is the prediction of documents that are relevant to users. One task of an information retrieval model is to define a ranking algorithm that assesses the relevance of a document. Research in information retrieval came up with three foundational models: boolean, vector and probabilistic. While the Boolean model represents documents and queries as sets, the vector model uses n-dimensional vectors. The latter is called an algebraic approach because linear algebra deals with vector spaces. Another branch of mathematics, probability theory, deals with the analysis of random phenomena. Findings from this area are being used in probabilistic models. Each of these models has been further enhanced.

The dominant opinion among researchers seems to be that the vector space model outperforms probabilistic models for general collections [14]. In the fol-

lowing sections two helpful algebraic models will be described: The classic vector model that uses TF-IDF as a weighting scheme and the more sophisticated latent semantic analysis (LSA). Both models seem to be very interesting for the purposes of ALAP.

Information extraction is another field of information retrieval, and it has been gaining in popularity over the past few years. In many situations, it is more useful to deal with structured information than unstructured text. The main concern of information extraction is how to generate structured information from unstructured or semi-structured forms of representation like text. In the field of information integration, IE is helpful in the course of data preparation. Many cultural heritage information systems cannot provide content with a level of structure that is adequate for certain matching tasks. Therefore, information extraction will be introduced as part of information retrieval.

Information retrieval is mainly concerned with the organization and access of textual documents. The models and techniques being explored in this field are helpful for the task of data preparation and entity resolution. Therefore, the following sections describe how these models are structured and how they relate to the project of information integration.

6.1.1 Information Extraction

Information extraction strives for the (semi-)automatic extraction of structured information from unstructured representations such as textual data. To that end, the models that are developed in the field of natural language processing are applied to natural language texts. One important information extraction activity is *part of speech tagging*, which is used to identify different classes of words. For example, it can identify the nouns that refer to entities in the world such as a person or a building. Once structured data has been extracted, it can be used for even more sophisticated types of processing. Additionally, the extraction of contextual information helps to properly interpret the data elements.

Information extraction is heavily used in text mining to extract relevant information from large amounts of unstructured or semi-structured textual data. McCallum [176] points out that information extraction can be seen as a preliminary activity for data mining. Figure 15 illustrates how information extraction fits into the process of knowledge discovery. If information systems can provide structured data, the information extraction step can be omitted. But since some information systems provide information as textual data, the entity extraction efforts should precede the actual data mining. Entity resolution is mainly based on data mining techniques and thus relies on high-quality structured data.

According to Konchady [159], the goal of information extraction is to establish structured information from unstructured text by assigning meaning or interpre-

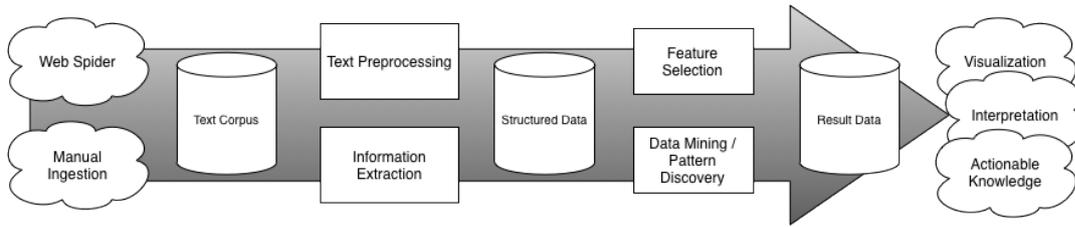


Figure 15: Information extraction and knowledge discovery workflow.

tations to text fragments. The extracted elements become part of the metadata of an entity and can be used for further processing. For example, a database that stores structured data can be populated with the extracted elements for querying. Heyer [139] makes use of the relational data model to store structured information, but semantic stores can be used too. Entity extraction is a sub-task of information extraction that focuses on the identification of names in unstructured text. Not only does entity extraction strive to detect names but also to resolve the names to types of entities like people, organizations, place names and so on. The optimal case would be to ascribe the name to a specific object or entity in the world.

Different techniques have been explored to implement entity extraction software. One of the most straightforward approaches is to maintain lists of entity names that are annotated with entity types. A simple search for tokens that occur in textual data can link names to certain entities. This method can be combined with extraction rules that are either hand-crafted or generated by statistical and machine learning models (e.g., Hidden Markov Models or Conditional Random Fields). A basic form of hand-crafting rules is to use a regular expression language that can be extended by providing access to dictionary data. Additional complexity can be covered by using a higher-order programming language to implement extraction rules.

The relevance of information extraction becomes obvious if one looks at the amount of data that has made available on-line. However, this information is unstructured and therefore cannot be processed according to its intended meaning [40]. Considering meaning that has been explicitly modeled as the metadata of a document is not possible in these cases. But information extraction can be interesting and useful for information integration projects as well. Some cultural heritage information systems provide information elements that have a certain amount of defined structure and meaning. But other information elements may consist of unstructured or semi-structured textual data that could contain useful information for entity resolution. Therefore, information extraction methods should be applied to these parts of information to achieve better results.

Of course, automatic information processing focuses on structured data because

it allows for “deeper” inspection and usually generates better results. Textual data may be represented according to a certain data model, and rudimentary schema elements may even be available. But major parts of the text will not have any explicit meaning associated with it. This allows for certain forms of processing that have been explored in the fields of information retrieval and text mining. The means that impose additional structure on this textual information make the information accessible to other methods of processing that can exploit the additional structural information. However, the disadvantages of methods that rely on a high level of formality have been discussed. A compromise must be reached between methods that deal with partially implicit information and methods that make inferences based on explicit meaning.

Text mining and information extraction in particular should be a central concern of information integration. Considerable amounts of information in the humanities and cultural heritage area are represented in unstructured or semi-structured textual form. In many cases, information extraction can also be useful if database fields have an internal structure that is not explicitly documented by the schema. Information extraction is, therefore, an important preliminary step because it may improve the results of knowledge discovery.

In the case of ALAP, the rules that are formalized as regular expressions turned out to be helpful for information extraction. More specifically, because some database fields do contain a rudimentary internal structure, they can be utilized to achieve additional granularity of information. Other content such as natural language text does not show implicit internal structure. In these cases, entity resolution should certainly be enhanced to make this information also available for extraction. However, extraction has not yet advanced to this level.

6.1.2 The TF-IDF Model

Vector space models have been used in the field of information retrieval for a long time. They belong to the group of algebraic models and represent text documents as vectors of terms that occur in these documents. Non-binary weights are assigned to tokens, which are then extracted from textual data. The weights express a degree of similarity between documents and queries. For example, in the case of information retrieval tasks, both documents and queries can be represented as vectors $d_j = (w_{1,j}, w_{2,j}, \dots, w_{t,j})$ where each dimension $w_{i,j}$ corresponds to a certain term. If a term occurs in the document, its vector component is different from zero.

One method of computing term weights of a vector is *term frequency / inverse document frequency* (tf-idf).²¹ The TF-IDF weight is often used in both infor-

²¹The foundation of TF-IDF has originally been elaborated by Spärck Jones [154]. Baeza-

mation retrieval and text mining. It is a statistical measure which evaluates the importance of a token for an individual document. In the vector space model, the TF-IDF weight is often used together with the cosine similarity. The combination of TF-IDF and the cosine similarity can be used to determine the similarity of two documents $\langle d_i, d_j \rangle$.

For information integration, the TF-IDF model is interesting because it can be applied to textual data. Entity descriptions in the field of cultural heritage are often comprised of full-texts that are only weakly structured. Although TF-IDF is known for matching user queries to documents, it can also be applied to calculate the similarity of documents according to the terms they contain. Since it does not rely on the explicit semantics of terms that occur in the documents, the model accepts a certain level of error. TF-IDF has been very successful in real-world applications like major web search engines.

The term frequency measures the importance of a term in a document. Terms are tokens that have been extracted from a document. $freq_{i,j}$ is the normalized number of occurrences of the term t_i in document d_j . $count_j$ is the number of occurrences of all terms in the document d_j . 1 is added to the number so that the logarithm becomes zero if the term does not occur in the document. The term frequency will be high if a term occurs in the document many times, or if only a few terms occur in the document. This can be a disadvantage with large documents.

$$tf_{i,j} = \log\left(\frac{freq_{i,j}}{count_j} + 1\right)$$

The inverse document frequency measures the overall importance of a term. N is the total number of documents in the corpus. n_i is the number of documents where the term t_i occurs. 1 is added to the result so that the logarithm becomes zero if the term does not occur in the corpus. The inverse document frequency will be high if the number of documents is large and if the term does not occur very often throughout the documents.

$$idf_i = \log\left(\frac{N}{n_i} + 1\right)$$

The tf-idf weight for documents i and j is the product of the term frequency and the document frequency. Therefore, it will be high if a term occurs in a query and a document, but not very often in all documents. This causes frequent words like “the” to have a very low weight.

Yates and Ribeiro-Neto [14] provide a comprehensive introduction. A number of variants exist for its application to calculate the *weighted cosine similarity*. For example, Leser and Naumann [168] describe one of them.

$$w_{i,j} = (\text{tf-idf})_{i,j} = \text{tf}_{i,j} \text{idf}_i$$

Finally, each document and the query can be represented as a vector of (tf-idf) weights. Each $w_{i,j}$ denotes the value for that specific term and the document or query.

$$d_i = \begin{pmatrix} w_{1,i} \\ w_{2,i} \\ \vdots \\ w_{t,i} \end{pmatrix}, d_j = \begin{pmatrix} w_{1,j} \\ w_{2,j} \\ \vdots \\ w_{t,j} \end{pmatrix}, q = \begin{pmatrix} w_{1,q} \\ w_{2,q} \\ \vdots \\ w_{t,q} \end{pmatrix}$$

The similarity can be expressed as the angles between the vectors. Equation 1 calculates the cosine similarity of two documents. The value of $\text{sim}(d_i, d_j)$ will vary between 0 and 1. For example, by determining a certain threshold like 0.9, one can define a set of documents that match for each query. Figure 16 illustrates the similarity of two vectors as the angle between them.

$$\text{sim}(d_i, d_j) = \frac{\vec{d}_i \vec{d}_j}{|\vec{d}_i| |\vec{d}_j|} = \frac{\sum_{i=1}^n w_{i,j} w_{i,q}}{\sqrt{\sum_{i=1}^n w_{i,j}^2} \sqrt{\sum_{i=1}^n w_{i,q}^2}} \quad (1)$$

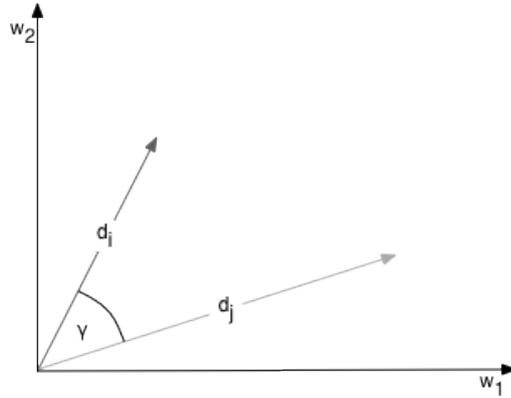


Figure 16: Determining the similarity of documents in a vector space model.

Because of its simplicity and the feasibility of assigning non-binary term weights, the vector model attracts considerable research. However, it has some shortcomings too: Long documents are not well represented because their term-frequency is very low. Keywords in documents must match exactly so that a weight can be calculated. This problem can be avoided by using string metrics to determine if

tokens match. Regarding the semantic structure of a document, the model assumes that the extracted terms are independent even if they aren't, but this does not seem to negatively affect the process anyway [14]. Another semantic problem arises from the fact that simple terms are represented as factors instead of complex document topics. Therefore, if a query uses a different vocabulary than the document to describe a topic, it is not successfully matched.

6.1.3 Latent Semantic Analysis

The vector space model in combination with TF-IDF weights turns out to be very useful for real-world information retrieval applications. Therefore, it is widely adopted and implemented by major information retrieval tools. However, the shortcomings of this model have been discussed, and ongoing research is currently working on the abovementioned problems. Latent Semantic Analysis (LSA) is one enhancement of TF-IDF that was created to eliminate some of its shortcomings. It strives to find the principal components in documents, which are referred to as concepts or topics. The technique promises better retrieval quality by matching the topics – instead of tokens – that are described in different documents. This section provides an overview of LSA and discusses its role for entity resolution.²²

Deerwester et al. published the seminal article on Latent Semantic Analysis, [83] and also hold the patent right for it. Latent Semantic Analysis can be applied to the results of statistical analysis according to TF-IDF. Thus, it can be interpreted as an extension of TF-IDF. LSA aims to discover latent semantics in document corpora by applying singular value decomposition to the term-document co-occurrence matrix. Latent meaning is derived when the dimensionality of the three resulting matrices is reduced, whereby each dimension represents a concept that is being dealt with in the documents.

The idea of Latent Semantic Analysis is based on decomposing the term-document matrix \vec{M} into three components via singular value decomposition (SVD). SVD is a factorization of an $m \times n$ matrix \vec{M} to three transformations \vec{U} (rotation), \vec{S} (scaling) and \vec{V}^* (rotation). After the factorization of \vec{M} , \vec{U} is a $m \times m$ matrix, \vec{S} is a $m \times n$ matrix and \vec{V}^* is an $n \times n$ matrix.

$$\vec{M} = \vec{U}\vec{S}\vec{V}^*$$

The column vectors of \vec{U} are called the left singular vectors, and the row vectors correspond to the term vectors. The values of the diagonal matrix \vec{S} are called singular values. They appear in descending order from the top left to the bottom right. The row vectors of \vec{V}^* are the left and right singular vectors, and the column vectors correspond to the document vectors. If the s largest singular values are

²²The description follows Stock [238].

document term	doc 1	doc 2	doc 3	doc 4	doc 5	doc 6	doc 7
text	12	18	9	52	3	1	2
information	8	19	11	48	3	2	0
extraction	12	21	8	52	4	2	3
trajan	2	1	2	3	18	32	11
column	3	1	2	1	19	33	8

Table 2: An example for a term-document co-occurrence matrix.

selected together with the corresponding left and right singular vectors, the result is a rank s approximation of \vec{M} .

$$\vec{M}_s = \vec{U}_s \vec{S}_s \vec{V}_s^*$$

As mentioned, a space of concepts is created by these calculations: the term vectors (rows of \vec{U}) and the document vectors (columns of \vec{V}^*) indicate the association of a term / a document with one of the concepts. The terms and documents can be compared and clustered by comparing these vectors.

Table 2 shows the frequency of terms in different documents. The distribution of term frequencies among different documents indicates that there are two big topics. Documents 1 through 4 belong to one topic, documents 5 through 7 belong to the other. It seems that the terms “text”, “information” and “extraction” belong to one topic, and “trajan” and “column” belong the other.

The term frequencies can be written as a matrix.

$$\vec{M} = \begin{pmatrix} \mathbf{12} & \mathbf{18} & \mathbf{9} & \mathbf{52} & 3 & 1 & 2 \\ \mathbf{8} & \mathbf{19} & \mathbf{11} & \mathbf{48} & 3 & 2 & 0 \\ \mathbf{12} & \mathbf{21} & \mathbf{8} & \mathbf{52} & 4 & 2 & 3 \\ 2 & 1 & 2 & 3 & \mathbf{18} & \mathbf{32} & \mathbf{11} \\ 3 & 1 & 2 & 1 & \mathbf{19} & \mathbf{33} & \mathbf{8} \end{pmatrix}$$

After decomposing the term-frequency matrix, the resulting three matrices allow for certain observations. The term-document matrix example in table 2 describes two topics: one is concerned with cultural heritage, and the other is concerned with this chapter. Therefore, it makes sense to choose the two largest singular values and to reduce the corresponding matrices accordingly. The parts of the matrices that form the reduced matrices have been emphasized.

$$\vec{S} = \begin{pmatrix} \mathbf{97.8273} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{54.1126} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4.36149 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2.40273 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2.19904 & 0 & 0 \end{pmatrix}$$

The row vectors of \vec{U} correspond to the terms that occur within the documents. The similarity of these vectors can be determined by calculating the cosine similarity. The emphasized vector elements suggest that the first three vectors are similar and correspond to topic 1. The remaining vectors are also similar and correspond to the other topic.

$$\vec{U} = \begin{pmatrix} -\mathbf{0.582299} & 0.0771626 & -0.31462 & 0.095706 & 0.739478 \\ -\mathbf{0.544605} & 0.0632085 & 0.814883 & -0.176143 & -0.0659432 \\ -\mathbf{0.593278} & 0.0512164 & -0.435546 & 0.0882979 & -0.669255 \\ -0.0861886 & -\mathbf{0.694576} & -0.151994 & -0.697225 & 0.0301781 \\ -0.0701864 & -\mathbf{0.710628} & 0.155489 & 0.682565 & -0.00330111 \end{pmatrix}$$

The column vectors of \vec{V}^* correspond to the documents that contain the terms. Here, the emphasized vector elements suggest that the first four documents belong to topic 1 and the remaining vectors belong to topic 2.

$$\vec{V}^* = \begin{pmatrix} -\mathbf{0.192653} & -\mathbf{0.341868} & -\mathbf{0.166521} & -\mathbf{0.895453} & \dots & -0.0455292 \\ -0.0272546 & 0.041769 & -0.0186818 & 0.127796 & \dots & -\mathbf{0.240561} \\ -0.532031 & 0.155141 & 0.608681 & -0.0446502 & \dots & -0.541993 \\ 0.604372 & 0.089723 & -0.166128 & -0.123105 & \dots & -0.729444 \\ 0.16625 & -0.895745 & 0.286327 & 0.260835 & \dots & -0.101525 \\ 0.502633 & 0.215298 & 0.50852 & -0.259324 & \dots & 0.220977 \\ 0.183139 & 0.0234371 & 0.482965 & -0.171464 & \dots & 0.23468 \end{pmatrix}$$

Latent Semantic Analysis is a promising approach for capturing latent semantics in large document corpora. But while the technique is well-suited for synonyms, it has problems identifying the relation of polysemy. The model assumes that similar tokens have similar meanings, and it does not include contextual information to distinguish the homograph tokens. This results in a good recall but bad precision for information retrieval and comparison operations. However, LSA is still useful in the field of information integration. By concatenating the elements of the entity descriptions which have been identified as referring to the same entity, the tokens that constitute the descriptions can be associated with similar topics. This could also be helpful for information integration in an international context. In this case, descriptions in different languages would be associated with similar topics. But this approach presupposes that a considerable amount of documents have already been aligned.

Latent Semantic Analysis has been discussed with respect to information integration in an international environment. A number of software libraries are currently available to support the singular value decomposition of large matrices.

The implementation of LSA in information integration is quite promising, as is evident in open source software [256]. But since many entity descriptions with a comprehensive vocabulary need to be aligned, the required matrices can become quite large. It would be interesting to assess the amount of computational resources needed for the initial decomposition. However, the exact effect for the comparison process – with only a few object descriptions aligned – has not yet been evaluated. For ALAP, a simple version of LSA has been implemented and tested on a small corpus.

6.2 Data Mining

The amount of information available in digital form is becoming overwhelming, and researchers are trying to tackle this problem. The methods and techniques from information retrieval that deal with sifting through immense amounts of data have already been introduced above. Data mining is another field which is mainly concerned with analyzing huge amounts of information for meaningful patterns. Data mining draws insights from many other areas. These comprise the aforementioned information retrieval, but also statistics, visualization, machine learning, etc. Thus, it is difficult to clearly define data mining. A few of its functions are introduced in this section.

Data mining tasks can generally be categorized into two groups: descriptive or predictive methods. While descriptive methods characterize a certain set of data, predictive methods use inference to make predictions. Han and Kamber [130] distinguish characterization / discrimination, classification / prediction, cluster analysis, outlier analysis and evolution analysis. Both descriptive and predictive methods will be used in the following experiment. Each will be examined with regard to how well it can establish links between entities.

For example, data can automatically be associated with classes or concepts, or with techniques of characterization and discrimination. Association analysis searches for attribute-value pairs that frequently occur together. Oftentimes, the data objects that bear different features than the majority of objects are discarded as noisy or erroneous data. In the context of data preparation, an outlier analysis can help to detect dirty data. Evolution analysis is another technique that can be used to analyze the behavior of data over time.

Classification is a method used to find a model that discriminates among the objects in different classes. The model is usually based on the evaluation of a set of labeled objects, the training data. Additionally, predictive models can be used to figure out missing numerical data values. In contrast to classification, data can be analyzed by clustering techniques even if the class labels are unknown. These labels will then be created by the clustering algorithm. Objects within a cluster share a high similarity, and objects of different clusters are very dissimilar.

Furthermore, clustering techniques can be used for taxonomy formation. These taxonomies could enhance entity resolution.

This section argues that entity resolution is a kind of knowledge discovery which can be modeled as a workflow. Data mining is described as the central analysis step of the knowledge discovery process. The descriptive methods which make use of statistical summaries and outlier analysis are used for exploratory data mining (EDM). EDM describes a set of techniques which facilitate a better understanding of data. In particular, predictive methods such as classification can help determine whether a set of object descriptions refer to the same entity or not. However, most predictive methods depend on a set of pre-labeled data, which is not always available. Finally, cluster analysis makes the entity resolution process efficient by arranging object descriptions with a high similarity into sets (clusters).

Most data mining techniques attempt to find meaningful patterns in the data, which can then be exploited to generate some (economic) advantage. Machine learning algorithms, for example, are the foundation of recommender systems –an established practice for e-commerce applications. A comprehensive suite of open source tools to implement data mining is maintained by Witten and Frank [261]. Data mining and machine learning libraries such as Mahout [242] provide for scalable solutions.²³ It seems that many techniques being explored under the notion of data mining are useful for entity resolution. The next section introduces relevant methods from this field. It describes and discusses their contributions to the entity resolution process.

6.2.1 Exploratory Data Mining

Heterogeneity and diversity are described as major obstacles for information integration. To achieve effective and efficient information integration and entity resolution, a good understanding of the actual data is helpful. It is the precondition for designing the information integration workflow. Suitable methods of knowledge discovery need to be selected and parameterized so that they work well with the data. Exploratory data mining (EDM) comprises methods that allow software developers to gain prior experience of the data. To that end, summary statistics and other models must be applied to the data. This section provides an introduction to EDM in the context of information integration and entity resolution.

According to Dasu and Johnson [77] EDM is “the preliminary process of discovering structure in a data set using statistical summaries, visualization and other means.”²⁴ EDM is also a way to perform a preliminary analysis of the provided data by different information systems. Choosing which modeling strategy to use

²³A comprehensive list of open source and machine learning tools can be found at “<http://mloss.org/about/>”.

²⁴The methodology that is described in this section follows Dasu and Johnson [77].

for information integration strongly depends on prior knowledge of the data. A prior analysis of information can reveal features that are relevant for a later analysis task or are even irrelevant due to dependencies.

EDM is also concerned with data quality; it highlights values that significantly deviate. By filtering and adjusting erroneous data, the results of the following analysis will be enhanced. Dasu and Johnson emphasize that up to 80% of project resources will go into data cleaning. Additionally, EDM has had to face the same problems as information integration. Many situations call for a joint analysis of data from different information systems. But EDM can also be used to enhance entity resolution..

An important task of EDM is to compile statistical summaries. A quick overview of certain data features can be compiled by making use of summary statistics, such as finding typical values like the mean, median and mode for the attributes of an object description. This also includes finding the measures of diversion like the variance and the position of quantiles. If single values turn out to be outliers, this may be an indication of wrong or erroneous data. Additionally, relationships between attributes like covariance, correlation and mutual information are informative. Strong relations between attribute values can be used for imputing information or finding information that is irrelevant for entity resolution.

Sometimes the distribution f of attributes values can be estimated on the basis of prior knowledge about the data. If the discovered distribution bears the features of a known probability distribution, the parameters of that function can be estimated from the data. Prior knowledge about the distribution is helpful because it allows for powerful statistical inference. But in many situations absolutely nothing is known about an underlying mathematical model like a probability distribution. If this is the case, the inferential power is limited and the focus shifts to determining the anchor points of the density f based on the data:²⁵

$$\{q_i\}_{i=0}^{i=K}, q_0 = -\infty, q_i = \infty, \alpha \approx \frac{K}{n} \quad (2)$$

q_i are the α quantiles that are used to create a histogram. K is the probability that any value of an attribute will be observed. And n the number of quantiles that will segment the overall probability in even parts α . The area between two adjacent quantiles can be determined by using the integral:

$$\int_{q_{i-1}}^{q_i} f(u)du = \alpha_i \forall_i \quad (3)$$

The fact that computers will process absolutely nonsensical data and use it to produce useless output is referred to as the principle of “garbage in – garbage

²⁵Equations 2 and 3 have been taken from [77].

out”. As discussed, problems of data quality can become a major obstacle for information integration and entity resolution in particular. Since it is not feasible for a human expert to inspect every database record, statistical summaries can provide useful heuristics for the identification of errors. Data quality problems comprise incomplete and suspicious data as well as its suitability for certain analysis methods.

Incomplete and erroneous data can result in biased analyses if not dealt with properly. Data is suspicious if it contains attribute values that behave in a different manner than the majority of the data. These outliers could be caused by, for example, any errors at the time of information entry generation. Another important activity is to determine whether the data is in a format and quality that can be passed onto certain analysis tools. For example, some data mining methods only accept categorical data while others rely on cardinal data.

Besides guaranteeing a certain degree of data quality, exploratory data mining can help with other preliminary analyses. Frequency counts for nominal values, and the probability distribution for cardinal values can aid in understanding the main focus of different information systems. These statistics can also be used to estimate whether information from different information systems overlaps. For example, a histogram can be created for cardinal attribute values to compare the distribution in different information systems. If the histograms bear similarities, the probability that information overlaps is higher as compared to dissimilar histograms.

Other forms of analysis can help to prepare and enhance the entity resolution process. Exploratory data mining techniques produce better results if the data that is subject to analysis has already been partitioned. The described partitioning techniques can also be used to make the entity resolution process more effective and efficient. The partitioning of whole data sets allows for better entity resolution because only the object descriptions that belong to a similar category (e.g., sculpture, vase, building, etc.) are jointly analyzed. Additionally, attribute values that are extremely frequent do not contribute beneficially to the similarity analysis. Some entity resolution methods automatically take this into consideration, but the analysis should focus on attributes that have high entropy. Omitting these values could result in more efficiency without strongly affecting the effectiveness.

Many data mining projects consider duplicate values and implicit correspondences as a data quality problem. Data quality issues complicate joint analysis and the processing of data from different information systems because duplicate data objects lead to biased analysis results. Additional automatic inference, which is discussed in the field of description logics, is misguided by dirty data. In this section, the entity resolution process is described as a data mining workflow in its own right. Thus, exploratory data mining also helps to identify the incomplete and

erroneous data which may prohibit proper entity resolution. Additionally, partitioning techniques and the identification of influential attributes can be treated as belonging to EDM for ALAP.

Entity resolution uses data analysis methods and techniques which are explored in the field of data mining and machine learning. According to the principle of “garbage in – garbage out”, the process relies on data with a certain degree of quality. Data preparation and cleaning does involve manual inspection by experts and can become rather laborious. But the techniques that are discussed under the notion of EDM support these activities. Some of the described methods provide a quick overview of the data contributed by different information systems and estimate their overlap. Other methods focus on detecting erroneous data and dealing with missing or incomplete data. Several of these methods are implemented as software components for ALAP.

6.2.2 Prediction of Coreferences as Classification

The preliminary steps of the data analysis process comprise extracting structured information from unstructured or semi-structured forms of representation. To that end, a certain degree of data quality must be guaranteed. Another important activity is to determine the similarity of object descriptions. This is usually done by considering the similarity of several features which have been identified as relevant. Then, a decision must be made on whether two (or more) object descriptions refer to the same entity. This decision can be modeled as a classification problem. The tuples of object descriptions are labeled as co-referring or not, based on the determined degrees of similarity.

By having the data analyzed by a classifier, the class membership of data elements can be predicted. The basis for this prediction process is a model that represents the peculiarities of each class. The classification models are most often learned by comparing them with example data, which is a set of data elements that have already been associated with a particular class (labeled data). In the field of machine learning, these methods are summarized as learning techniques since supervised training data is used to generate the model. Different paradigms have been proposed to represent these classification models, like simple classification rules, more complex decision trees or mathematical functions.

A group of machine learning models that has become rather popular is decision tree learning. Like many other models for classification, it relies on supervised learning, where examples are presented to the learning component for model creation. For decision tree learning, the model is represented as a set of rules that is arranged into a tree-like structure. Decision trees have a major advantage in so far as they use an explicit model that can be read and understood by human beings. This enables domain experts to evaluate a trained model and to manu-

ally introduce corrections and enhancements. This behavior and the simplicity of the used concepts make up the attractiveness of the decision tree approach to machine learning. However, other approaches like Support Vector Machines outmatch decision tree learning in some situations.

Mitchell [182] lists a number of features of decision trees that are helpful for the analysis process. Decision trees seem to have some beneficial features for the process of entity resolution: The determined similarity of two object descriptions will be expressed as a set of key-value pairs (e.g., height similarity: 87%) that represent the similarities of several features. To form a representative sample, these sets are labeled as describing a coreference relation or not, which results in a significant amount of missing values and leads to missing similarities for certain features. In these situations, and in situations where data is wrong or erroneous, decision trees remain robust and can produce useful results. Another advantage of decision tree learning is its computational features. The number of entity descriptions is quite large in situations where cultural heritage information systems need to share and integrate data.

Figure 17 illustrates how decision models can be represented as decision trees.²⁶ According to this model, a number of decision rules are organized into a tree-like graph structure. Each node of the graph represents a test of a certain attribute that controls the successive decision process on the basis of an attribute value. Thus, each instance is classified by a chain of tests that begin with the root node and recursively follow the edges according to the attribute value. The leaf nodes of the tree represent the final classification decision.

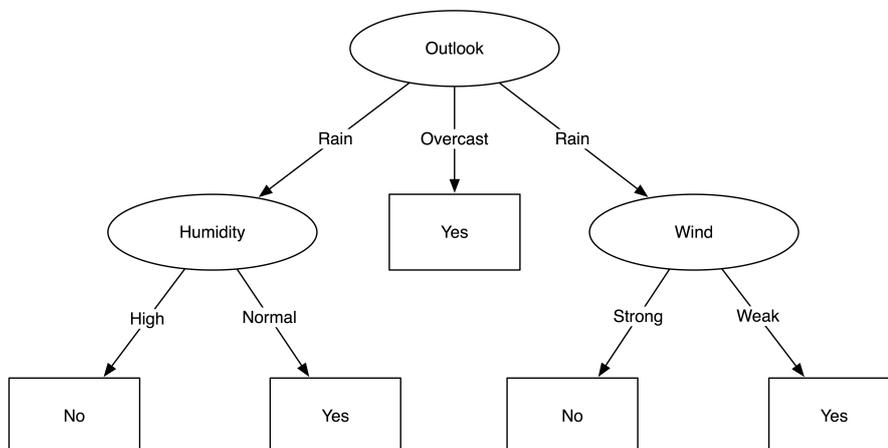


Figure 17: Classification of an example day by a decision tree. The learned concept is whether tennis should be played on days with specific features.

²⁶This example has been introduced by Quinlan [204].

Most approaches to decision tree learning use greedy heuristics, which always choose a local optimum at each step of the problem-solving process. Although this approach may reveal the global optimum, it cannot be guaranteed for greedy heuristics [65]. One popular and successful approach to decision tree construction is Quinlan’s Iterative Dichotomizer 3 (ID3)[204]. Since many other algorithms are variations of ideas that are covered by ID3, it will be dealt with here in more detail. Algorithm 1 illustrates the process of creating a decision tree with ID3 as a simplified pseudocode. After a root node has been constructed, a decision tree is created recursively by selecting the attribute that best classifies the instances at a given iteration and constructs a sub-tree for each attribute value.

Algorithm 1 Abbreviated pseudocode for the ID3 algorithm.

```

if all data elements belong to one class then
    construct a leaf with a class label
else
    select the feature with the highest information gain
    for all values of the selected feature do
        recursively construct all partial trees with the particular subsets as data.
    end for
    construct a node for the selected feature
    append all constructed partial trees to the node
end if

```

As mentioned above, the algorithm selects the attribute that best classifies the instances at every recursion. At each step, the instances are partitioned into subsets that form the input for the next recursive decision step. But which factors influence and guide this decision? The English Franciscan friar William of Ockham wrote “entities must not be multiplied beyond necessity” [106]. A modern interpretation of this principle is that the simplest explanation is usually the correct one. If different hypotheses are to be compared, one should choose the hypothesis that introduces the fewest assumptions while still sufficient to solve the problem. According to this principle, the algorithm prefers small trees over large ones.

The core of the decision process for model learning in ID3 is a statistical test (information gain), which reveals the attribute that best classifies the instances. A measure that is commonly used in the field of information theory forms the basis of this test. Shannon [218] introduced the entropy function in the field of information theory. In general, it measures the uncertainty that is associated with a random variable. The entropy can be computed as shown in equation 4. S is a collection of examples, and i represents the class that an example belongs to. p_i is the proportion of examples that belong to class i . It can also be interpreted as the probability that an example of class i is observed.

$$Entropy(S) \equiv \sum_{i=1}^c -p_i \log_2 p_i \quad (4)$$

Since the decision tree algorithm recursively tries to sort the instances into one class or the other, the entropy should be reduced at each decision step (node in the tree). The overall entropy in all subsets that have been created by the partitioning along one attribute should be lower than the entropy of the input set of the decision process. And the overall reduction of entropy at each decision step can be expressed as the information gain. The information gain can be formalized as equation 5. $Gain(S, A)$ is the information gain of an attribute A for a set of instances S . $Values(A)$ is the set of values for attribute A . S_v is the subset of instances S with the value v for the attribute A .

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \quad (5)$$

Several extensions have been proposed for the basic decision tree learning algorithm. These extensions are concerned with the problems of overfitting, alternative decision methods, computational complexity, missing values and continuous-valued attributes. Overfitting means that decision trees are trained to classify the example instances very well. But once they are confronted with real test-data, they perform worse because the classification process is too biased. In these situations it is helpful to prune the decision tree after the learning process so that it can be better generalized. Since measures like the information gain are biased towards attributes with many values in comparison to attributes with few values, alternative measures have been explored to deal with the problems of bias. Most of the above-mentioned challenges have been tackled by Quinlan [205] and his introduction of ID3's successor, C4.5. Both are very important for the problem of entity resolution.

First, the input for the classifier component, as mentioned, is the similarity of pairs of object descriptions. The similarity is represented as a set of degrees of similarities / matching attributes for different features of the object description. In many cases, the degree of similarity is represented as a continuous value that cannot be dealt with by the algorithm illustrated above. Therefore, means have been explored that allow for classification with non-discrete attribute values. A naive approach would be to decide for arbitrary thresholds. But C4.5 automatically decides for thresholds that maximize information gain.

Second, for a significant number of object features, the similarity will be missing due to missing parts of entity descriptions. In these situations it would be helpful to impute reasonable values for these attributes so that the decision process is not impeded. A simple approach could be to assign the most common attribute value

at a certain recursion step. Fractional instances have already been introduced in C4.5; they represent the probability that an instance has a specific attribute value. These split instances are then sorted down the tree, and the probabilities at the leaf-nodes determines the actual classification decision. This approach can be applied to both learning and actual model application.

The classification of pairs of instances into matching and non-matching object descriptions is an important step of entity resolution. Like every learning algorithm for classification, decision tree learning has advantages and disadvantages. In particular, algorithms that belong to the family of ID3 deal well with missing values and continuous-valued attributes. The computational complexity of the discussed algorithms allow for efficient learning and classification in comparison to other machine learning approaches. Additionally, open source implementations are available for C4.5. However, overfitting is a problem of decision tree learning, but methods such as pruning are available to mitigate these issues. Decision tree learning is based on a greedy heuristic, which does not guarantee the revealing of a global optimum.

Most important for every machine learning classifier is the performance of a model that, to a certain amount, depends on the quality of the learning data. In order to generate good and useful decision-tree models, the training data must be representative of the whole data set. Since the training data will probably have to be labeled by hand, it is doubtful that such a sample can be compiled in adequate time and with reasonable effort. Therefore, the model should be trained and evaluated in an iterative manner. This method is termed *bootstrapping* in machine learning. The model becomes more and more similar to a targeted “ideal” state with every iteration.

The type of knowledge learned by a machine learning algorithm is called the *target function*. In the case of decision tree learning, a target function is learned; it accepts an instance as input and outputs a class label corresponding to that instance. Thus, the target function for the task has a discrete output, and each instance (a set of degrees of similarity) either describes a matching or non-matching pair. Therefore, it is difficult to present matches to users in a weighted way. It would be helpful to present sets of entity descriptions that correspond to a high probability more prominent for users. This approach would support the process of bootstrapping in an iterative manner.

Crnkovic-Dodig [70] describes a survey on how the three different classifier approaches performed in different situations: naive bayesian classifiers, support vector machines and modular multilayer perceptron neural networks. He identifies several criteria which help to decide for or against a certain paradigm of classification:

- efficiency of the algorithm, its computational characteristics

- for training
- for actual classification
- complexity of the problems that can be handled
- sensitiveness to noisy data
- susceptibility to overfitting
- manual work for configuration

Decision tree learning is a useful paradigm to explore entity resolution for the information integration of cultural heritage information systems. Thus, decision tree learning is used for the implementation of ALAP. At a later stage, other approaches which can handle more complex situations should be included if the necessary computational resources are available.

6.2.3 Cluster Analysis of Vast Data Sets

Determining the degree of similarity of entity descriptions forms an important part of the entity resolution process. Classification has been recognized as a helpful approach because it allows for determining whether two entity descriptions refer to the same entity or not. But the number of entity descriptions that must be compared can be quite large. Since every entity description of one information system must be compared to every description of another, this method is prohibitive in most situations because it takes too long. A popular approach to this problem is to partition the search space into groups of entity descriptions which have a high degree of similarity according to a “cheap” similarity measure. A group of methods that can be used for this purpose is called *cluster analysis*.

Cluster analysis means that entities are automatically grouped with respect to the similarity of certain defined features. While classification relies on classes that have been defined a priori and learned by example data, this is not the case for clustering. Thus, clustering belongs to the group of unsupervised machine learning methods, where no training data is used for learning. Clustering algorithms usually use distance measures to determine the cluster membership of entities. Different clustering algorithms have been developed that use, for example, partitioning methods or density-based methods [130]. In the partitioning method, a number of clusters are generated by partitioning the data. Each partition here contains similar objects, and objects are dissimilar among partitions. While partitioning methods can only find clusters that are spherical-shaped, density based methods can find clusters of arbitrary shapes.

Cluster analysis has been used in many disciplines and is an area under active research. In biology, for example, hierarchical clustering has been used to derive taxonomies for plants and animals. The decision for a particular clustering algorithm depends on both the data and the desired outcome of the analysis step. Since clustering is a resource intensive task, effective and efficient methods are an active area of research. This is particularly important for the entity resolution of cultural heritage objects. Millions of object descriptions that exhibit a large set of features need to be clustered with respect to the similarity of relevant features.

McCallum, Nigam and Ungar [177] present canopy clustering, a method for cluster analysis that works on very large datasets with many features. Canopy clustering eliminates less influential data dependencies to make successive, more expensive operations efficient. The algorithm can cluster object descriptions into overlapping Canopies by using a rather cheap distance metric. For the determination of object descriptions, those in separate Canopies can be treated as having infinite distance. Algorithm 2 illustrates the method of canopy clustering, which is abbreviated as pseudocode.²⁷ Figure 18 illustrates how canopies have been assigned by the algorithm.

Algorithm 2 Abbreviated pseudocode for a canopy clusterer.

```
while there are unmarked data points do
  pick a point which is not strongly marked
  make that point a new canopy center
  mark all points within some threshold of it as in it's canopy
  strongly mark all points within some stronger threshold
end while
```

Canopy clustering uses a cheap distance measure to determine the initial Canopies for successive processing. Furthermore, it can be parallelized via a technique called MapReduce, which was introduced by Dean and Ghemawat [82]. MapReduce is a “programming model and associated implementation” that allows for easy and massive parallelization by abstraction. The framework relies on two functions that need to be implemented by the user, “map” and “reduce”, which take care of parallelization. The “map” function processes key/value pairs and outputs a set of intermediate key/value pairs. Then, the “reduce” function merges intermediate values according to its intermediate key.

Kimball, Michels-Slettvet and Bisciglia [158] reflect on how to apply cluster analysis to massive amounts of data. They propose to partition the very large data set into random chunks. These chunks of data are then distributed to different

²⁷The algorithm description follows a video lecture about MapReduce at <http://code.google.com/edu/submissions/mapreduce-minilecture/listing.html>.

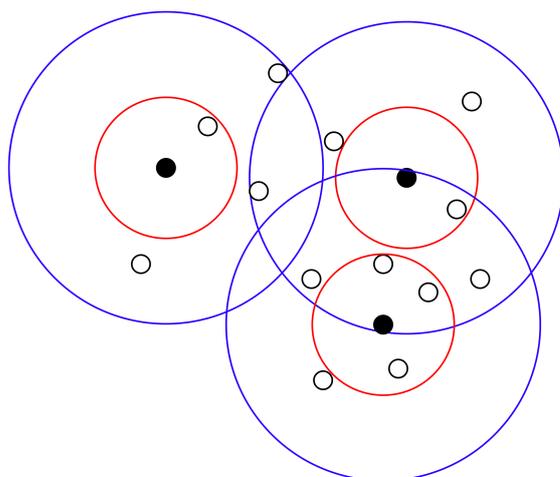


Figure 18: An example for Canopy clustering.

machines that perform the map operation. The map operation takes the chunk of data as input in the form of key/value pairs and basically executes algorithm 2 on them. The output is the canopy centers as intermediate key/value pairs, and these form the input of the reducer. A single reducer clusters the canopy centers again to find new canopy centers that can be applied to all data objects. As a final step, the overall data set is then partitioned into canopies with these final canopy centers. An implementation of this approach has become a part of “Apache Mahout” [242].²⁸

Clustering is an important step for entity resolution because it helps to significantly reduce the amount of comparisons which need to be carried out. Thereby, the amount of data which must be considered for classification is reduced to a feasible amount. Similar to classification, clustering is a resource-intensive endeavor and there is a need for efficient methods that can handle vast data sets. Thus, Canopy clustering in combination with massive parallelization is a promising approach. After the clustering step, only object descriptions within the same canopy need to be considered for the comparison process. Object descriptions which do not share the same canopy can be treated as having an infinite distance between them.

Canopy clustering in combination with existing open source implementations has been used in ALAP because it has superior computational features. Canopy clustering is more adaptable and flexible than the other proposed methods – particularly for the data sets considered in this experiment – and makes the overall process more efficient. MapReduce is seen as a way to parallelize search space

²⁸The Wiki of Apache Mahout provides more information about the actual implementation [7]

reduction. However, it does not parallelize the process of entity resolution itself. Kim and Lee [226] present a model that strives to analyze the whole entity resolution process. Such a model should be implemented in ALAP in future development phases.

7 A Theory of Similarity and Identity

Entity resolution links the pieces of documented information which co-refer to things in the world. These pieces of information are represented as (semi-)structured (explicit) or unstructured (implicit) semantics. The aspects of data mining and machine learning that are useful for automatizing parts of this progress have been explored and discussed in chapter 6. But in the course of determining the similarity of entity descriptions, it is not always easy to draw a conclusion on whether a coreference relation exists or not. For example, many philosophers investigate whether entities which tend to change over time remain the same even if certain features have changed. These debates have also influenced the way that sameness and identity are treated in knowledge representation and information integration. Therefore, this section reflects on the philosophical background of the notions *comparison*, *similarity* and *identity*. It introduces both philosophical arguments and mathematical foundations to derive further requirements for ALAP.

7.1 Philosophical Implications

Betrachte z.B. einmal die Vorgänge, die wir “Spiele” nennen. Ich meine Brettspiele, Kartenspiele, Ballspiel, Kampfspiele, usw. Was ist allen diesen gemeinsam? – Sag nicht: “Es muß ihnen etwas gemeinsam sein, sonst hießen sie nicht ‘Spiele’ ” – sondern schau, ob ihnen allen etwas gemeinsam ist. – Denn wenn du sie anschaust, wirst du zwar nicht etwas sehen, was allen gemeinsam wäre, aber du wirst Ähnlichkeiten, Verwandtschaften, sehen, und zwar eine ganze Reihe. Wie gesagt: denk nicht, sondern schau! – Schau z.B. die Brettspiele an, mit ihren mannigfachen Verwandtschaften. Nun geh zu den Kartenspielen über: hier findest du viele Entsprechungen mit jener ersten Klasse, aber viele gemeinsame Züge verschwinden, andere treten auf. Wenn wir nun zu den Ballspielen übergehen, so bleibt manches Gemeinsame erhalten, aber vieles geht verloren. – Sind sie alle ‘*unterhaltend*’. Vergleiche Schach mit dem Mühlfahren. Oder gibt es überall ein Gewinnen und Verlieren, oder eine Konkurrenz der Spielenden? Denk an die Patiences. In den Ballspielen gibt es Gewinnen und Verlieren; aber wenn ein Kind den Ball an die Wand wirft und wieder auffängt, so ist dieser Zug verschwunden. Schau, welche Rolle Geschick und Glück spielen. Und wie verschieden ist Geschick im Schachspiel und Geschick im Tennisspiel. Denk nun an die Reigenspiele: Hier ist das Element der Unterhaltung, aber wie viele der anderen Charakterzüge sind verschwunden! Und so können wir durch die vielen, vielen anderen Gruppen von Spielen gehen. Ähnlichkeiten auftauchen und verschwinden sehen.

Und das Ergebnis dieser Betrachtung lautet nun: Wir sehen ein kompliziertes Netz von Ähnlichkeiten, die einander übergreifen und kreuzen.

Ähnlichkeiten im Großen und Kleinen [262].

The definition of the concepts *similarity* and *identity* has consequences for the formalization of the entity resolution process. In logical systems, two things, A and B, are considered to be identical if no differences can be determined. This approach is problematic in the majority of situations where features of entities change over time. For example, a researcher may be interested in the development of a certain geographical location over time. For logical systems, the notion *identity* has a rather strict and precise definition. And because geographical locations tend to change constantly, different identities need to be attributed over time. Therefore, it would be helpful to link descriptions of places without making strong assertions about their identity. This section analyzes the concepts of similarity and identity from a philosophical perspective and discusses their impact on information integration and Semantic Web technology.

The human ability to determine similarities between entities in the world is very important. We tend to group things that have similar features into classes that we can refer to and talk about. Similarity is defined as the consistency of characteristics in at least two objects. If all features agree, including spatio-temporal ones, one would intuitively speak of identity. Thus, the similarity of objects can be determined by the process of comparison. The degree of similarity can be interpreted as the ratio of features that agree in relation to features that do not agree. It seems to be a paradox to say that two entities that agree on all features are identical, because they are not two but one. But in the course of the integration project, the entities are not compared but rather the representations or descriptions of entities. If these representations correspond to a high degree, this will be interpreted as an indication for coreference.

Similarity is of fundamental importance in almost every scientific field. For example, the field of geometry includes the notions of congruence and similarity. Two shapes are congruent if they can be transformed into the other through translation, rotation or reflection. The notion of similarity is introduced by adding the activity of scaling to the set of possible transformations. Similar geometrical objects bear the same shape. The field of statistics has explored methods to measure the similarities of entities under the notion of similarity analysis. Similarity metrics are frequently applied to nominal and ordinal variables, while distance metrics are applied to metric variables (interval, ratio). The Jaccard-Distance and the Tanimoto distance are two important examples of similarity metrics for nominal or ordinal values. The Euclidean distance is an example for metric variables. These similarity and distance metrics will be dealt with in more detail later.

The information that is represented in information systems could be interpreted as documented measurements and assessments. The abovementioned statistical methods could be used to determine the similarity of entity description

by analyzing the correspondence of the documented features. If there is a strong correspondence, it is more probable that two representations describe the same object in the world than otherwise. But attributing the relation of coreference to pieces of description is a strong and rigid assertion. This is significant because an entity not only exists in space but also in time. Modifications of entities can be problematic for the notion of identity.

This is illustrated by the Ship of Theseus, as reported by Plutarch [230]. The story about the many variations of the Ship of Theseus raises the question of whether an object remains the same if every part of it has been replaced. Rosenberg [210] tells a variation where the ship has 1000 planks that are being replaced one by one. The old planks are being used to build a new ship. Is Theseus' Ship the one that is being made of new planks or the one being rebuilt from the old planks? Rosenberg fears that each ship is Theseus' Ship, and he uses Calvus' paradox²⁹ and the factor of time to motivate his intuition.. Many criticize Rosenberg's first argument. They claim that it cannot overcome the problem of vagueness, which seems to require arbitrary definitions; and as far as his second argument is concerned, Ted Sider's [224] states that objects can indeed have identity over time (four-dimensionalism).

The introductory quotation illustrates what Wittgenstein calls *language games*. These are utterances in a practical context that also comprise manifestations of language in mathematics and formal logic. The quotation emphasizes that there cannot be a clear definition of the word "game". Rather, words obtain meaning by being used in everyday language, which is always tied to practical (nonverbal) contexts. Semantic Web technologies, in particular RDF(S) and the dialects of OWL, are based on traditional description logics. Therefore the effects that Wittgenstein illustrated also apply to Semantic Web languages. Semantic Web concepts rely on clear and sharp definitions of concepts and identities.

In an essay about the shortcomings of Semantic Web technology, Leavesley [166] points out that the Semantic Web is based on the identity of concepts (and consequently identifiers). He notices that the Semantic Web community is not aware of these problems and thus repeats the same mistakes as the AI community. In compliance with what has already mentioned as the limits of Semantic Web technologies, he states that "the world is fuzzy, sloppy and uncertain" and that "the same and similar are worlds apart." One of his commenters adds that "the vast majority of software in use today is based on similar conceptual approximations, yet somehow manages to be useful." Wittgenstein himself came to the conclusion that the propositions he made in the *Tractatus Logico-Philosophicus* were vague and therefore preliminary. He sees his propositions as a vehicle to illus-

²⁹This paradox considers how many hairs one must pull out of someone's head before he is bald.

trate his standpoint to others, who need to be emancipated from such propositions if they are to have a “correct” world view: “Meine Sätze erläutern dadurch, dass sie der, welcher mich versteht, am Ende als unsinnig erkennt, wenn er durch sie – auf ihnen – über sie hinausgestiegen ist. (Er muss sozusagen die Leiter wegwerfen, nachdem er auf ihr hinaufgestiegen ist.)” and “Er muss diese Sätze überwinden, dann sieht er die Welt richtig.”

Glaser et al. [118] further discuss how these problems affect the creation of a service that is able to manage coreference relations. The authors emphasize that the process of creating metadata (for the Semantic Web) itself is problematic. Two questions need to be addressed: How are things defined and represented by humans? And, what is the meaning of “identity”? These questions are directly related to the pragmatic aspects of language that have been discussed so far. The problem of identity can be resolved by considering the effects that have been mentioned by the notion of “language games”. Referring to Wittgenstein’s concept, two things can be considered identical if this is suggested by the use of identifiers in a pragmatic context. But this also means that the rigid notion of identity that has been developed in the context of formal logics must be abandoned.

Gärdenfors [113] criticism of traditional Semantic Web approaches has been outlined above. He begins by quoting Shirky’s [223] thoughts about the coreference problem:

No one who has ever dealt with merging databases would use the word ‘simply’. If making a thesaurus of field names were all there was to it, there would be no need for the Semantic Web; this process would work today. Contrariwise, to adopt a Lewis Carroll-ism, the use of hand-waving around the actual problem – human names are not globally unique – masks the triviality of linking Name and Person Name. Is your ”Person Name = John Smith” the same person as my “Name = John Q. Smith”? Who knows? Not the Semantic Web. The processor could “think” about this til[sic!] the silicon smokes without arriving at an answer.

According to Gärdenfors’ approach, geometric structures mediate between reality and symbolic representations. This is due to the fact that conceptual spaces are partially grounded in reality. Similarities can be modeled as distances in space along multiple dimensions, and they provide a tool to determine “identity”. While concepts are represented as regions in multidimensional spaces, names are points that fall into certain regions. If two names are mapped onto the same point, they can be considered to be identical. However, this vision of grounding Semantic Web languages in reality is far from being implemented, and its computational characters have not yet been explored. Thus, many of the necessary operations are based on similarity calculations by linear algebra and clustering algorithms.

Related to the problem of symbol grounding is the question of how the HTTP URIs that are used to refer to things in the Semantic Web acquire their meaning. Hayes [134] points out that there is currently no mechanism “for assigning referents to names, even for objects on the Web.” HTTP URIs can be resolved by an HTTP server to a representation for example as HTML. At the same time, URIs are used to name things that are represented in some Semantic Web languages. According to Hayes, this has led to confusion about what these URIs actually refer to. He points out that the access mechanism of the Web should not be confused with the mechanism of assigning a referent to a name. Many URIs, such as the city of Paris, will point to things that are not accessible on the Web. Therefore, reference cannot be established by ostentation and has to be determined by a description that will never be unambiguous. Hayes proposes trying to use this inherent ambiguity instead of trying to get rid of it. He argues that contextual features will help to select the right referent for a name.

Halpin [129] supports this position and observes at least three different approaches of defining the meaning of HTTP URIs: descriptivist, causal and social. Hayes [133] proposes a model theoretic semantic for RDF(S). According to this paradigm, URIs attain their referents by description or model interpretation. Halpin points out that the view of model theory is connected to the descriptivist theory of names, which has been mainly developed by Russell [212] and Frege [111]. This position has been questioned by Kripke [162], who argues that reference by description will fail in certain situations. Kripke is a supporter of the causal theory of reference, where names are assigned by an act of initial baptism. This name is then passed on to a greater audience that uses the name to refer to the same thing or person that was initially intended. Halpin finds that this view is advocated by Tim Berners-Lee, who claims that URIs are assigned referents by their owners.

The notion of “language games” can also be used to define the meaning of HTTP URIs, which has been elaborated above. According to this position, the identity of names can be determined by their use in language. The speaking of a language becomes part of a form a life, and Halpin asks what the form of life of the Web would be: the use of search engines. Wilks [174] points out that the extremely successful information retrieval is strongly influenced by Wittgenstein’s ideas. Margaret Masterman was one of Wittgenstein’s students and founded the Cambridge Language Research Unit. Under the auspices of this organization, the basic principles of information retrieval, including TF-IDF, have been developed. This is a strong argument for the Semantic Web community to adopt methods and techniques from information retrieval.

However, most implementations of Semantic Web technology rely on model theoretic semantics. There is no mechanism to determine the meaning of a URI in relation to its context of use. It seems that humans use a very efficient and

effective system of reference that is dependent on a certain context of use. In certain contexts, two entities can be viewed as causally related. For example, Theseus would consider the new ship as belonging to him, because it holds a causal relation to the old one. In this context, Theseus and his affiliates would refer to the new ship anytime they use the name “Ship of Theseus”. It would be prohibitive to record this causal relationship in a formal way because of the involved complexity. Means must be developed that are able to find an abstract representation of complex relationships in these types of situations. Spaermann [234] reflects on similarity and reminiscence, which emphasize the role of time:

Denn Selbigkeit, Identität “stellen” wir nur “fest” aufgrund der Ähnlichkeit der Weisen, in denen sich etwas durch die Zeit hindurch präsentiert, etwas, von dem wir im übrigen annehmen dürfen, daß seine Stellen im Raum innerhalb des Zeitraums seiner Existenz eine kontinuierliche Linie bilden.

From what has been said above, it seems that a specialized vocabulary for managing resolved entities would be helpful to Semantic Web information integration. To that end, Glaser, Jaffri and Millard [117] propose a coreference framework that makes use of so-called “bundles”. These bundles are sets of references to entities that depend on a certain context; different bundles are used for different contexts. If bundles are used, then the entities that are referred to by different URIs can be modeled as being equivalent in a specific context. These entities can also be modeled as distinct in other contexts. Additionally, the authors propose a Coreference Resolution Service (CRS). This service works in a distributed manner by introducing a way to move from URIs to associated coreference data. The authors also introduce a RDF vocabulary, which represents and shares coreference information as bundles.

Bouquet et al. [44] describe an Entity Name System (ENS), which was developed in the course of the OKKAM project. This project strongly focuses on supporting the systematic re-use of identifiers for entities on the Semantic Web. By providing the ENS as a service that is available on the Web, individual organizations are encouraged to lookup and re-use identifiers. The designers of the project expect that the problem of merging different representations of the same thing will be reduced to looking up identical identifiers in different information systems. The authors are aware of the arguments that have been introduced by Glaser, Jaffri and Millard, but argue: “While we share this general view, their point about URI potentially changing ‘meaning’ depending on the context in which they are used is philosophically disputable: the fact that several entities might be named in the same way (‘Spain’ the football team, ‘Spain’ the geographic location) must not lead to the conclusion that they can be considered the same under certain circumstances.”

Efforts towards information integration, such as those being pursued in the Semantic Web community, should consider the philosophical implications of similarity, identity and coreference. There is a long-standing debate about these topics, particularly in the field of language philosophy, which should not be ignored. The approaches illustrated above tend to circumvent the OWL property *owl:sameAs* because it is inadequate in many situations. It seems that *owl:sameAs* is only applicable if more than one URI is created for exactly the same / identical thing. However, the debate fosters different interpretations and implementations of “equivalence” as an alternative to “identity”. This results in semantic heterogeneity, which could undermine efforts to establish interoperability.

The issues and approaches discussed in this section must be considered for the alignment of Arachne and Perseus. A way of modeling and sharing coreference information in a semantically adequate way should be found. However, the effort and infrastructure that are required for such an endeavor are beyond the scope of ALAP. Therefore, a rather naive approach is pursued for the time being, which should be expanded upon by future follow-up projects.

7.2 Methodological Considerations

As emphasized in the previous section, our ability to compare is important for making sense of our environment and thinking about it. In order to recognize an entity, we must be able to distinguish it from another, which is achieved by determining the similarities of comparable features. Entity resolution is faced with the problem of instances that are referred to by different information systems in multiple ways. By comparing identifiers and assertions about entities, one can infer whether the same entity in the world is being referred to in different pieces of documentation. Therefore, the methods of comparison which allow for determining the similarity of information bits are very important to the process of entity resolution. The quintessential methods and challenges associated with the process of comparison are dealt with in this chapter.

The process of comparison presupposes that entities have at least some commonalities without having identical characteristics for some or all features [264]. It does not make sense to throw information together in an arbitrary manner, and then to try to resolve the entity descriptions. Therefore, it is helpful to assess in advance whether the two information systems contribute objects that are comparable. The result of comparison is a relation of either similarity or dissimilarity, as Husserl [146] has formulated for the field of arithmetic. For information integration, the activity of comparison aims to determine whether two descriptions refer to the same entity. Unlike the field of arithmetic, these descriptions are not simply similar or dissimilar but have degrees of similarity which must be determined.

The notion of measurement is traditionally defined as the activity of determin-

ing the magnitude of a quantity, such as length or mass. These measurements are determined relative to the specific units of measurement. To compare measurements, it is problematic to use different units of measurement to measure identical phenomena (length: meters, miles). These units of measurements are organized according to different systems that have historical meaning (Imperial System, Metric System). For the comparison of material cultural heritage entities, the two basic physical quantities meter and kilogram are frequently used.

In addition to these quantitative, qualitative measurements, Stevens [237] has introduced the “theory of scale types”. Steven’s theory has been widely adopted for statistical research despite the ongoing debate about whether it adequately represents reality. According to Sarle, meaningless statements about reality can be avoided if one considers the properties of the different scale types[214].

The lowermost level of measurement has been defined for nominal values. Things that bear a similar value for a certain attribute are assigned a similar symbol (e.g., material: marble, limestone etc.). This symbol denotes their membership in the set of things that share a certain attribute value. An important operation, which is applicable to nominal values, is to determine the similarity or dissimilarity of a characteristic value. Furthermore, there are ordinal values that are organized according to a specific order of precedence (e.g., epoch: roman, republican, late republican etc.). However, no information is available on the distance between two values. Because ordinal values can be represented as ordered mathematical sets, an additional operation can be applied to determine whether one value is greater or smaller than the other. Nominal and ordinal values can be summarized as categorial values.

If the order of values is known and the distance between them is defined, they can be measured on the interval scale (for example dating: 753 BC, 333 BC etc.). The interval scale determines the exact distance between two values of an attribute. But since the point of origin for these values has been arbitrarily defined, it does not make sense to determine the ratios between two values of an attribute. This is possible for attributes that can be measured on the ratio scale (e.g., height: 45 cm, 8.4 m). Interval and ratio values can be summarized as cardinal values, and they are sometimes also referred to as metric scales.

It is important to pay attention to the levels of measurement when trying to determine the similarity of two objects. Cardinal values can be compared by computing the distance or ratio of two values for the same attribute (numerical comparison). However, many of the values that are recorded for material cultural heritage need to be measured on the categorial scales. The mathematical operations that are defined for categorial can only determine the exact identity and have difficulty calculating distances. Although some formal conditions have been formulated for nominal values, pure statistical analysis seems to be agnostic on the

exact meaning and relations of categorial values. If different information sources are not using a shared, controlled vocabulary, then “string-metrics” can often help to find the equivalent values that are spelled differently.

The mathematical structure of categorial values is the (ordered) set. Classes and properties in description logics and model theory are also interpreted as sets. Thus, some knowledge representation systems have been crafted to define the additional semantics for categorial values. If multiple cultural heritage information systems are using shared, structured and controlled vocabularies, then the equivalence of categorial values could be determined by simple inference. However, the problems of this approach have already been discussed.

Entities of the world bear observable features. The characteristics of these features can be described as different levels of measurement. These levels of measurement guide the activity of comparison insofar as they define which mathematical operations are applicable. If shared vocabularies are not in place and variations in spelling cannot be easily resolved by string metrics, cardinal values still provide helpful information for the comparison process. These have the additional advantage of being language independent. They can be easily translated from one measurement system into the other.

Entity resolution depends on determining degrees of similarity for the different attributes that more than one entity have in common. These similarities must be combined to determine the similarities between the objects’ descriptions. Both nominal and cardinal values are considered for ALAP. Since the majority of values have been encoded as nominal values, string metrics and structured vocabularies turn out to be helpful. However, these vocabularies are currently not available and should be considered in the future.

7.3 (String) Distance Metrics

For categorial values and other textual information, the logical and mathematical operations as well as the alternatives of compiling summary statistics are rather limited. A common approach is to test categorial values for equality or inequality and to determine the relative position of a value in rankings. Even if different cultural heritage information systems use structured and controlled vocabularies for their documentation and description practice, they will probably not be compatible. The situation is even more complex if shorter or longer textual descriptions must be considered to determine the similarity of entity descriptions. While it is easy to measure the distance between interval values and ratio values, more complex distance metrics are proposed for values that are represented as textual strings: *string distance metrics*.

In mathematics, a metric has been defined as a function that assigns a real number R to all tuples of a set X [152]:

$$d : X \times X \rightarrow R$$

For all x, y and z in the set X , the following conditions need to be satisfied.

$$d(x, y) \geq 0 \text{ (non-negativity)}$$

$$d(x, y) = d(y, x) \text{ (symmetry)}$$

$$d(x, z) \leq d(x, y) + d(y, z) \text{ (subadditivity)}$$

Popular metrics are, for example, the discrete metric (the distance is either 0 or 1), the Euclidean metric (the distance between two points measured by a ruler) and the Manhattan metric (the sum of the absolute differences of the coordinates).

String (distance) metrics are defined as functions that assign a real number to all tuples of a set of strings. The application of string distance metrics is based on the anticipation that there may be spelling variations between strings that refer to the same entity. According to this approach, the probability that two strings refer to the same entity is high if the determined string distance is low. Different algorithms have been developed to compute distances, which then determine the similarity or dissimilarity of strings. String-based approaches consider a value as a string, while token-based approaches split a string value into tokens that form the subject of comparison. Cohen, Ravikumar and Fienberg [63] compared different string distance metrics. Some of these metrics work well with names, while others perform better if the order of string tokens is varied.

Levenstein [169] has introduced a string metric that has become popular for information integration and entity resolution. It is a metric over the space of symbol sequences. Figure 19 illustrates how the minimal number of steps can be determined for transforming the word “Archaeometry” into the word “Ethnoarchaeology”.³⁰ (1) The first column that corresponds to transforming the first string into an empty string is initialized. (2) Then, the first row that corresponds to transforming an empty string into the second string is initialized. The algorithm starts with the cell that corresponds to the first character of both strings and walks through the matrix column by column. If the characters that correspond to a cell do not match, the algorithm adds 1 to the value of the left (insertion), upper-left (substitution) and upper (deletion) cell and decides for the minimum. The result is written on the actual cell and the algorithm continues with the next cell. The minimum number of steps that is the Levenshtein distance appears in the bottom right cell.

³⁰This visualization was created with a tool developed by Charras and Lecroq [52].

	-1	0	1	2	3	4	5	6	7	8	9	10	11
		A	r	c	h	a	e	o	m	e	t	r	y
-1		0	1	2	3	4	5	6	7	8	9	10	11
0	E	1	1	2	3	4	5	6	7	8	9	10	11
1	t	2	2	2	3	4	5	6	7	8	9	10	11
2	h	3	3	3	3	4	5	6	7	8	9	10	11
3	n	4	4	4	4	4	5	6	7	8	9	10	11
4	o	5	5	5	5	5	5	5	6	7	8	9	10
5	a	6	6	6	6	6	6	6	6	7	8	9	10
6	r	7	7	7	7	7	7	7	7	7	8	9	10
7	c	8	8	8	8	8	8	8	8	8	8	9	10
8	h	9	9	9	9	9	9	9	9	9	9	9	10
9	a	10	10	10	10	10	10	10	10	10	10	10	10
10	e	11	11	11	11	11	11	11	11	11	11	11	11
11	o	12	12	12	12	12	12	12	12	12	12	12	12
12	l	13	13	13	13	13	13	13	13	13	13	13	13
13	o	14	14	14	14	14	14	14	14	14	14	14	14
14	g	15	15	15	15	15	15	15	15	15	15	15	15
15	y	16	16	16	16	16	16	16	16	16	16	16	16

Figure 19: A matrix for calculating the Levenshtein distance.

If a string consists of multiple words, the order of the words matters. The Levenshtein distance between “Lewenstein” and “Levenštejn” is 3. And the distance from “Samuel L. Jackson” to “Samuel Leroy Jackson” is 4, but it is 14 to “Jackson, Samuel L.”, even if both strings refer to the same person. Thus, this distance metric seems to be prone to omissions and insertions, and even more to the rearrangements of words. Other distance metrics have been proposed, such as those in the field of bioinformatics, which are similar to the Levenshtein metric but can deal with these challenges. Smith and Waterman [233] have proposed a metric that can assign lower weights to the prefixes and suffixes not being shared among strings. Winkler has proposed [258] a string distance metric, based on the work of Jaro [153], that specializes in matching the names of persons .

Another approach has been pursued under the notion of token-based distance metrics. Jaccard [151] detailed a statistic for comparing the similarity and diversity of sample sets. According to the Jaccard coefficient, the formula

$$J(A, B) = \frac{A \cap B}{A \cup B}$$

can compare two sets of tokens’ A and B .

To apply the formula, two strings need to be split into sets of tokens A and B . Then, the fraction of tokens that are shared among the strings in relation to all

tokens counts as a similarity metric. Of course, there are different ways to obtain tokens from whole strings. One naive approach is to take each word of a string as a token.

TF/IDF is another important method that has been explored in the field of information retrieval (see Chapter 6.1.2). According to this principle, each string is represented as a vector of tokens. By calculating the cosine between vectors, one can determine the similarity between vectors and strings, respectively. TF/IDF differs from the Jaccard coefficient by assigning weights to components of the vector elements, so that the rare tokens are assigned higher weights than the frequent tokens. However, if tokens have frequent spelling variations, this approach treats them as different. This leads to a distance that tends to be too high. Therefore, two methods have been considered to tackle this problem: n-grams and hybrid approaches.

One way to deal with spelling deviations in words is to use a different approach for the tokenization process. One could use an n-gram tokenization instead of a standard word-based tokenizer. N-grams form character sequences with a predefined length that are created from the original string. To that end, a window of a certain size is slid over the string from character to character. At each step, the character sequence that fits into the window is extracted as an n-gram. To avoid assigning lower weights to word boundaries, auxiliary characters can be inserted at the beginning and the end of the word. The following example illustrates how the string $s = \textit{Lewenstein}$ can be tokenized to 3-grams:

$$q\text{-gram}(s) = \{\#\#L, \#Le, Lew, ewe, wen, ens, nst, ste, tei, ein, in\#, n\#\#\}$$

The set of n-grams can now be analyzed with a token-based string metric like the Jaccard-coefficient or TF/IDF.

An alternative to using n-grams is to combine token-based approaches with string-based distance metrics. According to this paradigm, words can be treated as tokens that are compared with a distance metric. If the distance is below a certain threshold, the tokens are treated as identical. Cohen, Ravikumar and Fienberg [63] have elaborated a hybrid model that combines a secondary string-based metric (e.g., Jaro-Winkler) with a token-based approach (e.g., TF-IDF) and compared it with several other string distance metrics. The approach is similar to TF-IDF but also considers tokens that are close matches:

$$\text{SoftTFIDF}(S, T) = \sum_{w \in \text{CLOSE}(\Theta, S, T)} V(w, S)V(w, T)D(w, T)$$

$\text{CLOSE}(\Theta, S, T)$ is a set of words $w \in S$. $V(w, S)$ and $V(w, T)$ correspond to the calculation of the cosine similarity on the basis of TF-IDF. But in this

case, close matches will also be considered and weighted by $D(w, T)$, which is the similarity of the closest match for a word w to a word $v \in T$. Due to the considering and weighting of additional tokens, the calculated score can be larger than 1 in certain cases.

In addition, Ananthakrishna, Chaudhuri and Ganti [5] have presented an extension to the Jaccard coefficient that considers similar tokens. And Naumann and Weis [252] have introduced additional weights that correspond to the similarity of tokens for detecting duplicates in XML documents.

Monge and Elkan [183] compare the three algorithms that match pairs of tokens from two strings. Navarro [185] gives a thorough introduction and historical overview of different string matching algorithms. In the meantime advanced methods that comprise machine learning have been applied to model string metrics. Bilenko [36] gives an extensive overview of learnable similarity functions for record linkage in which strings metrics can be adapted to particular domains.

Different (string) distance metrics have been developed over time to determine the similarity of nominal values and attributes which bear shorter textual descriptions. The entity descriptions that share many similar values are considered as a match with higher probability than the entity descriptions only sharing few or no similar values at all. The underlying intuition of this approach is the observation that attribute values are prone to spelling deviations due to erroneous data acquisition or different data entry habits. Several approaches are introduced in this section, which are either string-based, token-based or hybrid. String-based algorithms account for spelling deviations, but they are prone to differences in the order of words. Token-based algorithms can deal well with different word orders, but they treat tokens with tiny spelling deviations as different tokens. Hybrid string distance combines the advantages of both by considering tokens with high similarity to be the same.

Some of the functions discussed above, such as Levenshtein, determine a distance between two strings which is not normalized (e.g., six edit operations are needed to transform one string into the other). Other functions like TF-IDF determine a normalized similarity value (always between zero and one). A distance function can be converted into a normalized similarity function and vice versa:

$$\text{sim}(x, y) = \frac{1}{1 + \text{dist}(x, y)}, \quad \text{dist}(x, y) = \begin{cases} \frac{1}{\text{sim}(x, y)} - 1 & \text{if } \text{sim}(x, y) > 0 \\ \infty & \text{if } \text{sim}(x, y) = 0 \end{cases}$$

For aligning Arachne and Perseus, information which bears both spelling deviations and differences in word order must be considered. Some entity description aspects stick to controlled vocabularies. These are not effectively processed by only applying string metrics because Arachne and Perseus use different national

languages. But other parts of the object descriptions comprise short textual characterizations with frequent usage of peoples', places' and institutions' names, which can be matched by string metrics. Thus, a hybrid metric can be applied to the values of entity descriptions that meet the above-mentioned demands. In addition to other means, this approach generates valuable input to determine whether several entity descriptions refer to the same entity or not.

7.4 Automatizing the Decision Process

As emphasized, the resolution of entities relies on the activities of comparison and decision making. The previous section has shed some light on how to determine the similarity of object descriptions. The means required to represent the relation of identity in an adequate and helpful manner have also been discussed. But, how exactly can the results of different comparison activities be combined so that a comprehensive picture can be acquired? How can a decision process be formalized to determine whether two entity descriptions refer to the same entity in the world? These activities must be defined so that human effort can be supported by the automated processing of datasets. Thus, the mathematical foundations of comparing and drawing inferences from this information are dealt with in the following paragraphs.

Automatizing the process of determining the similarity of entity descriptions presupposes that these descriptions are available in a machine-readable manner. Additionally, multiple information systems should be analyzed with respect to the comparability of their contents. For these information systems, comparable attributes need to be aligned by transformation and the mapping of schema elements. Data types that are associated with certain schema elements may indicate how the comparison process needs to be carried out. In most cases, it is helpful to know the actual meaning of the data elements to determine their similarity. For example, the data type integer does not indicate if a certain value needs to be measured on the interval or ratio scale. The data type string does not indicate the difference between nominal, ordinal or something completely different.

The result of each comparison activity should be represented, for example, as a quantification of the determined degree of similarity. And there should be some (semi-)automated process that makes a decision on the identity or dissimilarity of the described entities. Different statistical summaries like the arithmetic mean could be applied to these quantified results to calculate an overall similarity. Since degrees of similarity can be interpreted as fuzzy values for the similarity of each attribute, they could also be combined using t-norms.³¹ In many situations, par-

³¹T-norms introduces fuzzy values to generalize the logical conjunction . A popular t-norm is $\top_{\min}(a, b) = \min(a, b)$, which always results in the smaller of the two values a and b .

ticular attributes are a better indicator for the identity of reference than others. Means that are based on a simple combination of these t-norms would neglect this phenomenon and the results would be suboptimal. Additionally, only results that have been quantified could be combined, while a decision activity could also result in “high similarity”, “moderate similarity” and “low similarity”.

The rules that guide the decision process can either be handcrafted or learned from examples. If the peculiarities of the data sets to be integrated are well-known, it can be useful to hand-craft decision rules. Many claim that different attributes contribute different amounts of useful information to the decision process. For example, if two information systems need to share data that focuses on sculptures, the information that a record represents a sculpture is obviously not helpful for the process. By carefully crafting rules that consider this, the results can be enhanced. However, in many situations, the amount of information that the attribute contributes in advance is unknown. In these situations, it can be advantageous to learn the decision rules based on examples.

Many learning algorithms that have been explored in the field of data mining are based on the statistical analysis of large amounts of data. Currently, there are two dominant paradigms for constructing statistical models: the frequentist paradigm and the Bayesian paradigm. They differ in their fundamental approach with regard to how they interpret probabilities and draw inferences from data. The way that hypotheses are formulated according to the Bayesian approach can be particularly helpful for making decisions that are based on statistical inference in a record linkage setting.

Frequentist statistics relies on experiments that have been performed a large number of times. This is called the law of large numbers. Based on the central limit theorem, frequentists assume that the mean of a sufficient large number of independent random variables will be approximately normally distributed. In this paradigm a hypothesis is either true or false. Bayesian statistics assumes that there is a priori knowledge about the probability distribution. Based on sampling, the a priori knowledge can change, evolve and become a posteriori knowledge about the distribution. This procedure can be described as learning by experience. According to this paradigm a hypothesis can be true with a certain probability. In many cases, the prior probability is based on a subjective estimation which was acquired by considering previous studies, intuition or expert knowledge.

The Bayesian paradigm makes use of Bayes’ theorem. The following example demonstrates how Bayes’ theorem can be applied to a small experiment.³² Figure 20 shows a probability tree for deciding for a coin and observing a head or a tail for two coins A and B. Coin A has a head and a tail, coin B has only heads.

³²This is an elaborated version of an example described by Herzog, Scheuren and Winkler [138].

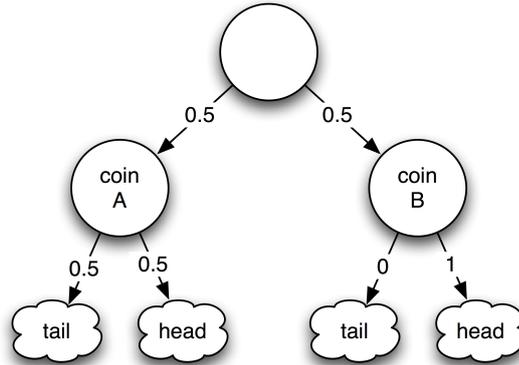


Figure 20: A probability tree showing the example experiment.

One could ask how often one observes six heads after having chosen a coin:

$$1^6 \times 0.5 + 0.5^6 \times 0.5 = 0.51 \text{ (51\%)}$$

Now, one may – after having observed six heads – be interested in the probability of choosing the coin with only heads. This is the proportion of the probability where six heads are observed on the coin with only heads (0.5×1^6), and the overall probability of observing six heads in a row (0.51):

$$\frac{0.5 \times 1}{0.51} = 0.98 \text{ (98\%)}$$

This can be formalized as Bayes' Theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- $P(A)$ is the prior probability of A . It is “prior” because it does not take into account any knowledge about B . The probability of A (selecting the coin with only heads) is .5 because there are only two coins.
- $P(B|A)$ is the conditional probability of B given A . This is also called the likelihood. If the coin with only two heads is selected, the probability of drawing a head is 1.
- $P(B)$ is the prior probability of B . It acts as a normalizing constant. The probability of B (observing six heads) is the sum of the probability of selecting a coin and observing a head. In this case this is $1 \times 0.5 + 0.5^6 \times 0.5 = 0.51$.
- $P(A|B)$ is the conditional probability of A given B . This is the value we are looking for.

$$\begin{aligned}
P(II|\text{SHO}) &= \frac{P(\text{SHO}|II)P(II)}{P(\text{SHO})} \\
&= \frac{P(\text{SHO}|II)P(II)}{P(\text{SHO}|II)P(II) + P(\text{SHO}|I)P(I)} \\
&= \frac{1 \times 0.5}{1 \times 0.5 + 0.5^6 \times 0.5} = \frac{64}{65} = 0.98
\end{aligned} \tag{6}$$

If one has observed six heads, the probability of selecting the coin with heads on both sides changes from $\frac{1}{2}$ to $\frac{64}{65}$. The latter is called the a posteriori knowledge and is the quantified result of a learning process.

Bayes' Theorem has been introduced here because it provides a foundational model for many machine learning applications. For example, the entity extraction that needs to determine the most probable entity type for an observed token can be enhanced by using the Bayesian approach. Additionally, record linkage software has implemented or used Bayesian learning where each comparison activity should result in a decision for a match or a non-match. This can be modeled as a classification problem with two classes M (match) and \overline{M} (*non-match*).

$$C = (M|\overline{M})$$

The probability for a match or a non-match according to the determined similarity can be formulated as Bayes' theorem:

$$P(M|S) = \frac{P(S|M)P(M)}{P(S)}, P(\overline{M}|S) = \frac{P(S|\overline{M})P(\overline{M})}{P(S)}$$

The matcher is guided by the relation of the probability for a match compared to a non-match with respect to a certain similarity:

$$P(M|S) > P(\overline{M}|S)$$

And because $P(S)$ only has a normalizing function, it can be omitted by calculating the quotient R :

$$R = \frac{P(M|S)}{P(\overline{M}|S)} = \frac{P(S|M)P(M)}{P(S|\overline{M})P(\overline{M})}$$

If R is greater than 1, the classifier will choose a match. If the ration is smaller than 1, the classifier will decide for a non-match. The overall similarity is composed of single similarities that have been determined in isolation. After defining a certain matching threshold for each similarity, it can be expressed as $P(S_i|M)$:

$$P(S_i|M) = \frac{|\text{matching records with attribute-specific similarity } S_i|}{|\text{matching records}|}$$

In the case of a non-match, the similarity distribution can be formalized too:

$$P(S_i|\bar{M}) = \frac{|\text{non-matching records with attribute-specific similarity } S_i|}{|\text{non-matching records}|}$$

If the matching features, which have already been determined, are conditionally independent, they can be further decomposed:

$$P(S|M) = P(S_1|M)P(S_2|M) \dots P(S_n|M)P(M)$$

The same applies to non-matches:

$$P(S|\bar{M}) = P(S_1|\bar{M})P(S_2|\bar{M}) \dots P(S_n|\bar{M})P(\bar{M})$$

Then, the ratio Q can be determined by considering these single probabilities:

$$R = \frac{P(M|S)}{P(\bar{M}|S)} = \frac{P(S_1|M)P(S_2|M) \dots P(S_n|M)P(M)}{P(S_1|\bar{M})P(S_2|\bar{M}) \dots P(S_n|\bar{M})P(\bar{M})}$$

The process of finding object descriptions that co-refer to an entity in the world can be modeled as a classification problem. However, the problem of estimating the prior probabilities $P(S_n|M)$, $P(S_n|\bar{M})$, $P(M)$ and $P(\bar{M})$ remains. These must be determined in advance by experts or by introducing a bootstrapping step, which enhances the classifier by ongoing training. The probabilistic modeling of a decision process is based on the observation of events, which can either be observed or not observed. It would be useful to have an algorithm which could decide by determining the *degree* of similarity for each feature. The techniques which consider degrees of similarity are introduced and discussed in more detail later.

The basic principles of making a decision on whether entity descriptions refer to the same entity or not have been introduced in this section. Bayesian learning, which makes extensive use of conditional probabilities, can be used to model this process. It is related to the first formalization of a record linkage model by Fellegi and Sunter [104], which is described in more detail later. However, it turns out that the underlying model of Bayesian learning is too simplistic for more complex learning challenges. Therefore, different models are considered for ALAP. In particular, decision tree learning is also considered in the following because it has already been successfully used for entity resolution applications.

8 Towards an Entity Resolution Framework

The previous sections have elaborated on the basic building blocks that are beneficial for the construction of a cultural heritage entity resolution framework. This section focuses on how to connect the individual components and concepts in order to construct a framework for ALAP. To that end, the topics that have already been identified as relevant are elaborated and described in relation to each other. Entity resolution is a complex and domain-specific endeavor; it requires individualized frameworks that suit the particular needs of projects. Therefore different alternatives are presented, and their potentials and shortcomings are discussed to foster an informed decision.

To acquire first-hand experience of semi-automatic information integration, the concepts that have been elaborated so far have been implemented as several software components. In a kind of experimental information integration project, these software components have been applied to the information that was extracted from Arachne and Perseus. A better understanding of how an adequate entity resolution should look could be achieved by applying these methods to cultural heritage information. Information on the performance of different alignment methods has been collected and the expected success of the experiment has been assessed.

For the discussion of entity resolution frameworks, it is helpful to keep in mind the global infrastructure in which they will be embedded. For example, Dörr and Iorizzo [89] sketch the idea of a global knowledge network for the humanities and for cultural heritage in particular. They emphasize the importance and challenges of entity resolution in such an environment. Eide [94, 95] discusses a number of peculiarities for the implementation of an entity resolution system. He also describes a type of system with an endpoint, which allows for retrieving, creating and editing coreference information.

The majority of approaches to information integration and entity resolution attempt to align the descriptions of homogeneous entities with limited complexity. But the expected complexity of the information that is managed by Arachne and Perseus is much higher. The number of considered entity types is high compared to other approaches. Additionally, the number of relations between entities is higher, and the types of these relations are very diverse. These requirements cannot be comprehensively considered in the initial alignment experiment. Thus, one challenge is to pick the methods that are efficient as well as effective and that can be implemented with reasonable effort.

Because of the involved difficulties, ALAP focuses on specific entity types for entity resolution. The entity resolution methods that deal with few entity types are well-understood and established in other fields. Subsequently, the advanced methods for working with multiple entity types and complex relationships are explored. In particular, matchers must be evaluated with respect to the peculiarities

of entity descriptions for the archaeological domain. The next sections strive to outline the architecture for the entity resolution experiment. The discussion begins with a real-world example, and is followed by an elaboration of the architecture's requirements.

8.1 A Real-World Example

After carrying out a prescreening, the pairs of entity representations are presented to a matcher component of an entity resolution system. The matcher determines the similarities by considering the similarities of features that have been extracted. The quality of the assessment depends on adequate similarity functions, which must consider the types of extracted features. For example, there are entity resolution approaches which treat cardinal values with string distance functions. If one entity description contains the value "2.01 meters" and the other "79,13'", the result of the comparison process would not be satisfactory. Therefore, this section reflects on how to treat different features of an entity description by using an example from Arachne and Perseus.

Figure 21 shows two database records, one originating from Perseus and the other from Arachne.³³ Interestingly, both seem to describe the same entity, a statue of the emperor Augustus which was found at a place named Ariccia near Rome. It is now on display at the Museum of Fine Arts in Boston. The figure shows a description of both entities (in this case archaeological objects) according to the vocabulary of the CIDOC CRM. This way of representing information should help with comparing the features that make up the entity description.

One could imagine several approaches to help machines with resolving the two entity descriptions with the identifiers. "Perseus : Boston 99.334" and "Arachne : 2913" refer to the same entity. Several of the (string) distance functions and similarity metrics that have already been introduced can be applied to nominal and cardinal values or to values that hold textual information. In many cases, more than one distance or similarity metric can be used. And this decision could have a major influence on the quality of the final matching decision. But always deciding for very precise metrics may lead to problems with data elements that do not meet certain quality standards. Thus, applying these metrics would lead to additional effort in data preparation and may even be prohibitive if the resources for analysis tasks are scarce. The following paragraphs reflect on the applicability of different metrics with regard to the example that was introduced above.

Only some of the information in the entity descriptions fall into the statistical category of cardinal values. For example, the entity feature that is modeled as "has dimension" represents the dimensions of the material entities. The entity

³³This is an elaborated version of the example that has been first presented in [12].

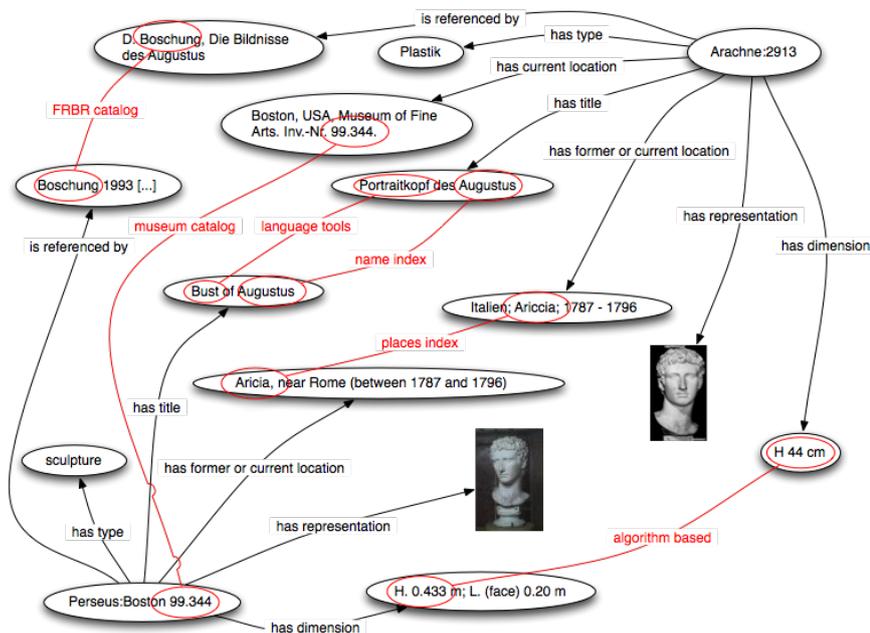


Figure 21: Approaches to coreference resolution.

description of Perseus represents the dimensions as a string “H. 0.433 m; L. (face) 0.20 m”. This string value obviously shows an internal structure by using a semi-colon to delimit two measurement elements. Additionally, letters like “H” and “L” indicate the type of measurement, and extra information is given in parentheses. The entity description of Arachne seems to represent the height of a material object in a more granular fashion by simply stating “H 44 cm”. To calculate a ratio of the heights for Arachne and Perseus, the corresponding values first need to be extracted from these strings and normalized as the same unit. The success of this endeavor strongly depends on how well rules can be formulated or learned for the information extraction process.

Due to the CRM’s predefined class structure, it is unable to model the types of archaeological objects, so any additional information about the type needs to be represented as the CRM property “type”. Since this is a textual representation, string metrics may be the most appropriate method for comparison. In this case Perseus named the type “sculpture”, and Arachne decided for “Plastik”. This example shows the complexity of the semantic relations between provenance vocabularies. Beyond the problem of having different national languages for the baptism of types, these terms are not even semantically equivalent: the German term “Plastik” describes a particular type of sculpture that is made out of stone. Thus, the German term is not only represented in a different national language

but denotes a type that is more specific. Since the number of entity types is rather limited in many cultural heritage information systems, a simple, manual approach to this problem would be to split both data sets into partitions. These partitions would consist of comparable object types to allow entity resolution to be performed in a straightforward manner.

A more ambitious and complex approach would be to maintain some kind of knowledge organization system in the background. This knowledge organization system would provide the names of entities in multiple national languages and represent the semantic relations between entities. Then, the word “Plastik” could be compared with all the more general and specific terms in English. But, this approach would require the information systems that strive for information integration to stick to that vocabulary. Spelling deviations in the same national language would make it difficult to find a word in this structured and controlled vocabulary. Although string metrics could be used in these situations, it is obvious that the complexity of this approach is high and no information is available on its scalability. In fact, this could be seen as an additional entity resolution problem.

Each entity description is linked to a short textual summary about the entity. While Perseus describes the example entity as “Bust of Augustus”, Arachne uses the string “Portraitkopf des Augustus”. In this case a token-based (hybrid) string distance metric would match the string “Augustus”, but it would neglect the other two terms. It may be helpful to introduce a preliminary step of “normalization” via machine translation here. But in the example, the phrases that are used for the short textual entity description cannot be interpreted as parts of a longer text. Since many (statistical) methods of machine translation rely on bits of natural language that are associated with more context, other techniques like Latent Semantic Analysis and its extensions seem to be more promising. Even if only low similarities are determined between matches, the relations between the similarities of non-matches are still more important.

In this case a more ambitious approach would also be possible. It must be kept in mind that the short textual entity descriptions cannot be interpreted as full sentences in natural language because of their implicit internal structure. In these short textual phrases, the names of the ancient people, places and institutions that are used could be extracted via an information extraction component. A structured and controlled vocabulary could then be used to resolve names with spelling variations, different names that have been used for the same person (e.g., “Augustus” and “Octavius Caesar”) or identical names that have been used for different persons or places (e.g., Alexandria”). However, these controlled and structured vocabularies of names must be comprehensive to cover the majority of occurrences in information systems.

Many cultural heritage information systems record the provenance of the in-

formation that has been used in an entity description. To determine whether two entity descriptions are referred to by the same bibliographic reference, the probability of a coreference relation must be higher according to the context attraction principle. The bibliographic information that refers to the entity has been represented via the CRM property “is referenced by”. Both entity descriptions comprise the information that this specific bust of Augustus was referenced by a monograph that was published by Boschung in 1993. While Perseus’ description only mentions the name of the authors, the year of publishing and provides a list of occurrences in the work, Arachne’s description includes all of these elements as well as the full title.

Since there is no unique number like an ISBN, other means for comparison need to be found. A simple string metric would result in a low similarity of bibliographic references despite the fact that both descriptions refer to the same monograph. Therefore, it would be wise to normalize both bibliographic references to the information that they have in common, the author and the year. However, a residual uncertainty remains because an author or an author collective could have published more than one work in a certain year. Additionally, different editions of the same work are most often published in different years. Again, the usage of background knowledge could lead to better results. A bibliographic information system could determine whether the combination of an author with different years of publishing refers to the same work.

Any information about the former and current locations of the described entities is represented by “has (former or) current location”. While Perseus describes the entity as being “Found at Aricia, near Rome (between 1787 and 1796)”, Arachne provides the following information: “Italien, Ariccia, 1787 – 1796”. Although the description of Perseus is more verbose than that of Arachne, a hybrid distance metric would probably result in a high similarity, which would be adequate for this pair of entity descriptions because of their high number of similar tokens.

Both information systems mention Ariccia with minor spelling deviations and provide some contextual information about it (i.e., it is situated near Rome or is part of Italy). The place names could be isolated by a preliminary step of information extraction (e.g., a fuzzy index lookup in a comprehensive list of names). The extracted name information could then be passed to a so-called gazetteer with the request to resolve this information into a canonical identifier. Projects like Geonames [255] run information systems with geo-referenced materials online. Geonames provides a web service that publishes its data according to the principles of the Linked Data initiative. This makes it easy not only to embed the service in some kind of coreference resolution infrastructure but also to use the identifiers that are already provided.

Visual information is also represented by using the CRM property “has representation”. This mechanism is often used to associate a number of visual representations (digitized photographs of the material object) with the entity. It would be useful to provide the means to analyze image contents in order to determine the similarities between different images. A subfield of computer vision has been established to actively explore image similarity functions: Content Based Image Retrieval (CBIR). The CBIR community explores robust image annotation and similarity functions. Datta et al. [78] has compiled a comprehensive overview of these activities.

In summary, several areas have been identified that contribute methods and techniques to assess whether the two digital representations of an (archaeological) entity refer to the same thing. First, straightforward calculations of distances and ratios can be applied to the entity features that have been represented as cardinal values. Second, (hybrid) string distance metrics can be applied for the comparison of entity features that have been represented as nominal values or short textual descriptions. Longer textual descriptions that form a narrative should be treated with information extraction methods first. Third, nominal values that are part of structured, controlled vocabularies can be compared by evaluating the semantic relations between different values. In the future, computer vision can provide additional information that could help with entity resolution.

The discussion of the example pair of entity descriptions emphasizes that systems striving for information alignment have to face a number of diverse problems. Some matching strategies focus on the object descriptions themselves to determine their similarity. Other techniques rely on background information, which must be constantly maintained and published in a machine-actionable way. In a number of cases, computing the similarities of strings will fail due to the usage of multiple natural languages. Furthermore, external resources may not be available.

An important strategy is to focus on those parts of the entity descriptions that are best suited for comparison. Then, statistical methods such as Latent Semantic Analysis can be used to “learn” the similarities of values that cannot be processed adequately by other means. Since scientists use entity resolution as part of their everyday work, a good start may be to build a system that makes it easy for them to resolve entities online in the course of their work. The resulting data could then be used to train systems how to make informed recommendations on further coreferences that may still be in need of resolution.

The analysis of the example above has been used to decide on adequate similarity functions that can deal with relevant aspects of the entity descriptions. These similarity metrics are either to be implemented or re-used. The latter is possible if there are third-party software libraries available that can be used for the entity resolution framework. For the implementation of the experimental

matching workflow, simple approaches are preferred over more fitting but highly complex approaches. Moreover, the entity resolution architecture can be successfully complemented with additional components that may enhance the matching quality. These components should address whether using string metrics is in fact the appropriate approach for the discovery of synonymy and homonymy without contextual information.

8.2 A Theory of Entity Resolution

A number of components have been introduced which are relevant for the task of entity resolution. These components must be combined to form an entity resolution framework, which has a matcher at its core. Machine learning techniques have already been introduced in chapter 6, which are apt to implement such a component. The alignment of entity descriptions that are organized along a certain schema and data model has been studied for quite some time now under the notion *record linkage*. This section introduces early formalizations of techniques for entity resolution³⁴ and discusses their influence on ALAP.

Dunn [92] and Newcombe et al. [187] developed the first ideas about record linkage. Fellegi and Sunter [104] were the first to suggest that a mathematical formalization should be the theoretical foundation of record linkage. Since their approach has strongly influenced further developments in the area of entity resolution, it will be introduced in this section. Fellegi and Sunter's approach will be the starting point for elaborating the approach that has been chosen for ALAP. Elmagamid, Ipeirotis and Verykios [98] classify the theory of Fellegi and Sunter as probabilistic entity resolution.

Similar to the Bayesian approach, the ratio of two probabilities needs to be determined. Fellegi and Sunter's model calculates the ratio of the probability that an observed pair of entities belongs to the set of matches *to* the probability that the pair belongs to the set of non-matches. These probabilities are conditional probabilities in which a specific constellation of matches or non-matches of attributes is observed. The observation is formulated as being dependent on whether the pair belongs to the set of matches or non-matches. Equation 7 shows the formalized representation of the model.

$$R = \frac{P(\gamma \in \Gamma | r \in M)}{P(\gamma \in \Gamma | r \in U)} \quad (7)$$

This equation can be applied to a pair of entity descriptions (describing archaeological objects). The first example illustrates a pair where an agreement

³⁴Herzog, Scheuren and Winkler [138] provide a good introduction of this theoretical model. The presentation in this section follows this introduction.

can be observed for location and height. The agreement or disagreement can be interpreted as a discrete metric where the distance is either infinite (or “one” in the normalized case) or zero. The distance can be determined, for example, with the means that have been elaborated so far. The probabilities that have been introduced above can be used to compute the need ratio:

$$R = \frac{P(\text{agree on location, agree on height}|r \in M)}{P(\text{agree on location, agree on height}|r \in U)}$$

The next example illustrates a disagreement on the height of an object. In this case, the ratio will be lower if the height of an object is a good indicator for a match. It will not influence the ratio if all objects have approximately the same height.

$$R = \frac{P(\text{agree on location, disagree on height}|r \in M)}{P(\text{agree on location, disagree on height}|r \in U)}$$

After calculating the ratios for pairs of entity descriptions, a decision needs to be made on whether the pair describes the same entity or not. Fellegi and Sunter have introduced a decision rule that makes use of thresholds for the ratio R . If R is greater or equal to the upper threshold, the pair r refers to the same entity. If R is below the upper threshold but greater than the lower threshold, the pair r *potentially* refers to the same entity. And if R is below or equal to the lower threshold, the pair r refers to different entities.

To simplify the calculation, one must assume the conditional independence of all attributes that are relevant for determining the agreements. The example can then be rewritten with conditionally independent *marginal probabilities*. These can be divided into m- and u- probabilities for matches and non-matches.

$$P(\text{agree on location, agree on height}|r \in M) = P(\text{agree on location}|r \in M) \cdot P(\text{agree on height}|r \in M) \quad (8)$$

The m-probability corresponds to the probability of a feature agreement, provided that a pair of records refers to the same entity. And the u-probability corresponds to the probability of a feature agreement, provided that a pair of records does not refer to the same entity.

$$m_i = P[\text{agreement in field } i|r \in M], u_i = P[\text{agreement in field } i|r \in U]$$

Equation 9 demonstrates how the marginal probabilities are calculated. It distinguishes two cases, one for agreements and one for non-agreements. If m_i is large and u_i is small, then w_i will become large in case of an agreement and small

in case of a non-agreement. If m_i increases and u_i decreases, then the likelihood for a match in case of an agreement in field i would be reduced. According to this mechanism, agreements are increased by large values for m_i and small values for u_i . At the same time, disagreements are weakened by large values for m_i and small values for u_i .

[I do not yet understand, what to do with negative values for disagreements if m_i is greater than u_i .]

$$w_i = \begin{cases} \log_2\left(\frac{m_i}{u_i}\right), & \text{if agreement in field } i \\ \log_2\left(\frac{1-m_i}{1-u_i}\right), & \text{if otherwise.} \end{cases} \quad (9)$$

Equation 10 demonstrates the ratio that is assigned to each matching pair. The vector $\gamma = (\gamma_1, \dots, \gamma_n)$ holds the configuration of agreements and disagreements within the cross-product space $\Gamma = \Gamma_1, \dots, \Gamma_n$, which denotes all possible configurations of agreements and disagreements.

$$R = \frac{P[(\gamma_1, \dots, \gamma_n) \in \Gamma_1 \times \dots \times \Gamma_n | r \in M]}{P[(\gamma_1, \dots, \gamma_n) \in \Gamma_1 \times \dots \times \Gamma_n | r \in U]} \quad (10)$$

Because the *matching weight* needs to be computed, the logarithm needs to be applied to both sides of the equation. The *matching weight* is defined as $\log_2(R)$.

$$\log_2(R) = \log_2 \left\{ \frac{P[(\gamma_1, \dots, \gamma_n) \in \Gamma_1 \times \dots \times \Gamma_n | r \in M]}{P[(\gamma_1, \dots, \gamma_n) \in \Gamma_1 \times \dots \times \Gamma_n | r \in U]} \right\}$$

Equation 8 has introduced the concept of conditional independence. If this concept is being applied, the equation can be rewritten:

$$\log_2(R) = \log_2 \left\{ \prod_{i=1}^n \frac{P[\gamma_i \in \Gamma_i | r \in M]}{P[\gamma_i \in \Gamma_i | r \in U]} \right\}$$

Since the logarithm of a product of numbers is the sum of the logarithms of these numbers, the equation can be rewritten as:

$$\log_2(R) = \sum_{i=1}^n \log_2 \left\{ \frac{P[\gamma_i \in \Gamma_i | r \in M]}{P[\gamma_i \in \Gamma_i | r \in U]} \right\}$$

The last term is just the sum of the weights. Large positive matching weights suggest that the pair of records is a match. Large negative weights suggest a non-match:

$$\log_2(R) = \sum_{i=1}^n w_i$$

The advantage of this approach is that individual agreement weights are properly calculated if conditional independence holds. But the m - and u -probabilities must be estimated. This can be difficult if one only has poor knowledge about the data. In addition, the upper and lower thresholds for the decision rule must be determined a priori by evaluating the error bounds on false matches and false non-matches. Thus, data from prior studies must be used, or parameters must be estimated by using the data from current files. Another possibility is to use the Expectation-Maximization-Algorithm to estimate the parameters by statistical inference. In any case, experiences or data from prior studies has to be available, or representative samples from current files have to be produced by clerical work, before one can apply the described approach.

Winkler [259] notices that the approach by Fellegi and Sunter has a lot in common with the Bayesian approach of inferencing. The discussion above shows how the method is quite similar to the Bayesian classifiers, which were introduced in section 7.4. Additionally, Bilenko et al. [33] observe that commonalities exist between the entity resolution model by Fellegi and Sunter and TF-IDF. The terms “frequencies” and “inverse document frequencies” are then interpreted in analogy to the marginal weights.

A number of modern approaches to the problem of entity resolution allude to the foundations that have been elaborated by Dunn and formalized by Fellegi and Sunter. The formal model exhibits a set of useful features. These features include both weighting the contribution of agreements on particular features and the decision of whether a pair of entity descriptions refers to the same entity. At the same time, the model bears similar problems, which refined approaches must learn how to deal with. In order to craft adequate models for entity resolution, prior knowledge about the data is mandatory. A set of representative reference pairs that have already been labeled can tremendously enhance the record linkage quality.

8.3 Entity Resolution Frameworks

Entity resolution frameworks consist of multiple components which implement specific steps of the entity resolution process. Information integration projects must choose between different models for each component. For example, different approaches are available to extract relevant information from information systems. Additionally, the text mining community is exploring multiple ways to combine entity resolution components. Techniques that rely on statistical learning for entity resolution require different combinations of components, as opposed to rule-based approaches. This section reflects on a number of relevant architectural decisions and their applicability for ALAP.

The task of entity resolution has been described as both a preliminary step to

knowledge discovery and as a subject of knowledge discovery itself. Thus, entity resolution is not an end in itself, but it does make certain use cases possible (e.g., the user scenario introduced in chapter 2.3). It has been argued that several methods and techniques that are considered parts of the knowledge discovery process are vital for entity resolution. For example, clustering has been explored as a way to make entity resolution more efficient. And learned or hand-crafted classification rules combine the results of several matchers or string similarity functions. Some approaches also use clustering for the matching component, which assigns the entity descriptions that refer to the same entity to the same cluster.³⁵ In the end, these approaches depend on high quality data and granularity.

Köpcke, Thor and Rahm [161] have evaluated several entity resolution approaches for “real-word” data. Most importantly, they have identified the two major approaches that the projects have pursued to model the entity resolution process. These are contrasted in figure 22. Non-learning matching approaches rely on hand-crafted models to make match decisions (left), while learning-based match approaches use sample data to train the model (right). Exploratory data mining seems to be useful in both cases to get familiar with the information that is subject to integration. In particular, hand-crafted decision models presuppose preliminary knowledge about the way that entities are described in their respective information systems. Additionally, model generation that is based on learning requires a set of examples that are either hand-picked or discovered by another (automatized) strategy.³⁶

The entity resolution workflow starts by acquiring relevant information from participating information systems. Thus, almost all architectural designs suggest starting with the extraction of relevant data from the information systems that strive to share and integrate information. It has been mentioned that one should allocate adequate resources for this step because many information systems cannot provide data that is well-suited for knowledge discovery. The information extraction activity has been described as part of a data cleaning process.

Many approaches for entity resolution depend on a pair-wise treatment of entity descriptions. Thus, all entity descriptions in one information system must be compared to all entity descriptions in another information system by determining the similarity of their documented features. Thus, the Cartesian product of both sets of entity descriptions forms the basis of the comparison process. But for most data sets the number of required comparisons would be prohibitively high. If both information systems contribute 1 000 entity descriptions, the number of comparisons will be as high as 1 000 000. And if the comparison of all features of

³⁵For example, Bhattacharya and Getoor [31] use relational clustering for entity resolution instead of a model-based classifier.

³⁶For example, Bhattacharya and Getoor [30] have proposed an algorithm that is based on Gibbs sampling and that requires no labeled data.

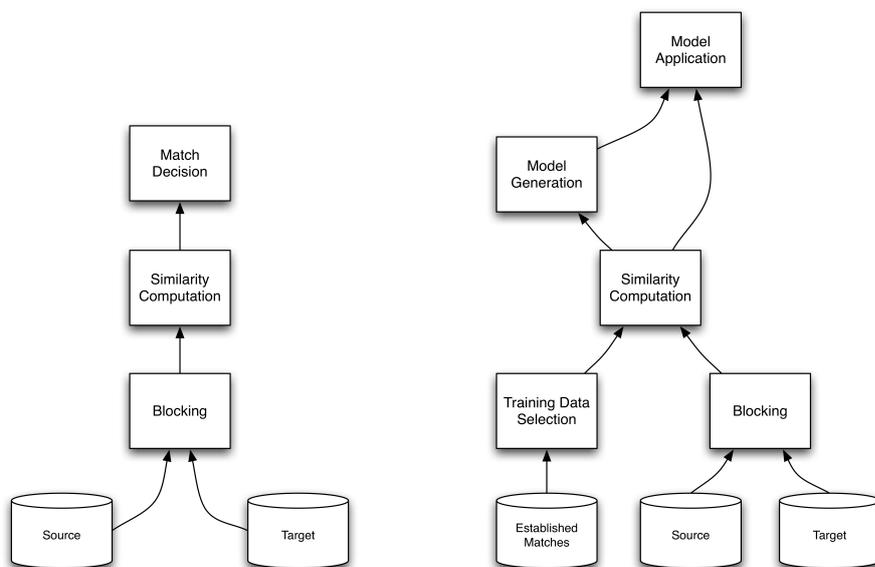


Figure 22: Architecture of an entity matching framework.

two objects (attribute values, contextual information, etc.) lasts 0.01 seconds, the process will run almost 12 days. The field of entity resolution is thus exploring methods that are suited to either reduce the number of needed comparisons and / or speed up the comparison process.

The architectures rely on models that have been crafted either (semi-)automatically or manually. The input data for these decision models is an n-tuple of discrete or continuous distance or similarity values that have been determined by different (string) similarity metrics. The decision model puts the pair of entity descriptions whose similarity is represented as the n-tuple either into the category of matches or into the category of non-matches. Some approaches may make use of more than two classes by introducing an additional class for possible matches. Other approaches do not use classification on the basis of similarities; instead they consider unsupervised methods and techniques like clustering. Here, several entity descriptions that refer to the same entities are put into the same clusters.

It is useful to present automatically generated match decisions to expert users. A user interface should present matching pairs together with additional contextual information, because this makes it possible for the domain expert to verify the decision and either acknowledge or refuse the match. Matches that have been acknowledged by expert users become additional training data that can be used to generate better models if training-based model generation is part of the architecture. This approach starts with a model that is bootstrapped with a few pairs of entity descriptions, which leads to iteratively enhanced model performance. An

additional advantage of having users verify the match decisions is that the information about a match can be annotated with provenance information that users can decide to either trust or not.

It has been mentioned that the quality of entity resolution depends on proper customization to a high degree. Many of the steps that have been described so far rely on the manual creation of rules or parameterization of machine learning components. For example, Baxter, Christen and Churches [16] observe how unsupervised machine learning methods such as canopy clustering tend to rely on adequate parameterization. The performance can significantly drop if the model parameters of single components are not optimally tuned. Although the underlying methods and techniques can be generalized, this is not the case for the parameterization of a specific information integration project. Therefore, it would be interesting to explore methods that semi-automatically determine model parameters to achieve iterative optimization. Therefore, entity resolution architectures should be built in a way that is highly customizable to guarantee a wide applicability and quality.

All of the abovementioned approaches bear individual advantages and disadvantages. Hand-crafted models tend to be explicit, easy to understand and can be adapted by manual work. However, this approach requires a good understanding of the underlying information, and additional modeling effort is often needed. Architectures that arrange for model generation based on machine learning require adequate training data. Again, this requires manual parameterization and the introduction of automatic parameterization methods. Machine learning approaches are disadvantageous insofar as the supervised and unsupervised methods require considerable computational resources. Additionally, methods that use examples to generate model require an additional learning step, and this can be resource intensive for some model types. However, Köpcke, Thor and Rahm [161] have observed that certain machine learning approaches outperform manual modeling approaches. This is particularly relevant for situations where more than one attribute must be considered for the entity resolution process.

Two major architectural approaches for entity resolution are discussed in this section. These approaches either make use of hand-crafted models or of data mining techniques that rely on supervised and unsupervised machine learning. Since the amount of data (the number of entity descriptions) that needs to be considered by a matcher is usually too high to be efficiently processed, all architectures make use of a blocking method. Blocking is the activity of prescreening entity descriptions with respect to the probability that they refer to the same entity. Usually, adequate prior knowledge about the data that is provided by the information systems or actual training data is not available. Therefore an approach of iterative model generation must be pursued. In addition, all approaches rely on customization and parameterization. Even the approaches that rely on machine

learning depend on a number of parameters which need to be optimized in order to generate useful results.

It is useful to experiment with different entity resolution architectures for information alignment. A high number of heterogeneous entity descriptions must be aligned in the course of ALAP. Therefore, an entity resolution architecture which allows for training-based model generation should be preferred. Decision tree learning is a reasonable choice to start with because it is based on an explicit model representation and has beneficial computational characteristics. Explicit model representation means that the model can be understood by humans without an extensive process of analysis and interpretation. Additionally, the beneficial computational characteristics of decision tree models allow it to be applied to a large amount of entity descriptions for experimentation. In the future, more advanced techniques such as Support Vector Machines should also be considered, even if their success heavily relies on adequate parameterization. Combinations of multiple machine learning approaches are also possible and are discussed later.

8.3.1 Blocking and Resolving Entities

Blocking entities and detecting matches can be considered to be the core activities of the entity resolution workflow. Therefore, most research projects focus on enhancing these components to enhance the quality of the output. This section focuses on how to introduce different blocking approaches and discusses their applicability for the information integration experiment. Foundational components and straightforward approaches have already been addressed in chapters 6 and ???. In contrast to these foundational components and straightforward approaches, the following paragraphs introduce alternatives and advanced approaches to enhance entity resolution workflow. Unfortunately, these alternative approaches are also significantly more complex than the more basic approaches.

It has been mentioned that most architectures determine the pair-wise similarities of entity descriptions that stem from different information systems. This requires computationally intensive activities to do the actual matching, which makes some kind of prescreening for entity descriptions inevitable. Thus, information integration projects usually introduce a blocking component that applies a strategy which is qualified to group together entity descriptions. This entails that the entity descriptions from different information systems that refer to the same entity with a high probability be considered together. For the blockage process, distance functions will be used that are computationally cheap, because they do not have the same precision as more sophisticated distance functions. Blocking can dramatically reduce the number of pairs that need to be processed by the matcher component.

If only the sculpture records of Arachne (40 077 sculpture records) and Perseus

(2003 sculpture records) were to be compared, $40\,077 \times 2\,003 = 80\,274\,231$ comparisons would need to be performed. The maximum expected number of matches is 2,003 in this case, and the number of possible non-matches is $80\,274\,231 - 2\,003 = 80\,272\,228$. By creating a so-called blocking key, an algorithm can separate database records into multiple buckets and compare them. For example, if only sculptures of a specific collection were considered, this value would decrease dramatically. In a comparison of the pieces in the collection of the Museum of Fine Arts, Boston, this results in $1\,599 \times 197 = 315\,003$ comparisons. Furthermore, it would be possible to mark all pairs of records as non-matches that disagree on the blocking-key.

But if an unsuitable blocking key is selected, the rate of false non-matching pairs could be rather high. This is due to the records that disagree on the blocking key, which will be considered non-matches and not be compared. For example, if two different persons assessed the size of a sculpture differently, there would be two records with different measurement information for the same sculpture. If the measurement ranges are used as a key for blocking, these records could end up in different buckets and will never be compared. To avoid this situation, multiple passes can be performed, each one with a different key for blocking. Thus, if the measurement is the first blocking criterion, the second could be, for example, the depository. Although, the first key separated the linking records into different buckets, the second may mark them for comparison. Therefore, it is important to contrive keys independently. The keys are oftentimes a combination of more than one field, and these fields should not be used in more than one strategy if the aim is to create independent keys.

To maximize the efficiency of the blocking approach, certain features of blocking keys should be taken into account. The entity description aspect that serves as a blocking key should contain a large number of possible values to ensure reasonable bucket sizes. Additionally, the number of records that fall within one bucket should be evenly distributed among all buckets. As a third condition, the chosen data element for the blocking field should have a low probability for errors. For ALAP, the depository of an archaeological object seems to be a reasonable choice as a blocking key. 561 sculpture records are associated with the place “Athens, National Archaeological Museum”, and 68 sculpture records are associated with the place “Berlin, Antikenmuseum” (approximately 8 / 1). This key seems to be slightly better than “material”, for example. Here 1471 sculpture records consist of “marble” and 148 sculpture records consist of “bronze” (approximately 10 / 1).

Jaro [153] has applied a standard blocking method, which is illustrated in the example above: the entity descriptions that share a certain blocking key are grouped together. A number of additional, suitable approaches to prescreen entity descriptions have been proposed to make the overall entity resolution process

more efficient. These approaches can be broadly categorized into techniques that apply windowing and clustering, each of which may be supported by an index structure. The large amount of available publications emphasizes the importance of partitioning methods for the efficiency of the entity resolution process.

Hernández and Stolfo [136, 137] describe another approach: the sorted neighborhood method. This procedure uses a virtual window of fixed size that is slid over the pairs of entity descriptions that have been sorted beforehand according to a defined key. This window defines a neighborhood of pairs that are tested according to whether they refer to the same entity or not. Yan et al [263] propose an extension to the sorted neighborhood approach. They use less cost-intensive procedures to bring co-referring entity descriptions closer to each other.

McCallum, Nigam and Ungar [177] propose using the technique of canopy clustering on high dimensional data sets, as was discussed above. Hadjieleftheriou et al. [126] propose variants of string similarity functions like TF-IDF that allow for designing efficient indexes to speed up the comparison process. Christen [58] describes Febrl (Freely Available Record Linkage System), which uses bigram indexing, among others. Ananthakrishna, Chaudhuri and Ganti [5] have developed a duplicate identification filter to form sets of potentially duplicate entity descriptions.

Machine learning techniques are not only useful for matching but also for blocking. Bilenko [32] observes that not only is the matching process domain dependent but the blocking strategy is as well. Thus, he proposes an adaptive framework in which the blocking functions can be trained for the peculiarities of different domains. Michelson and Knoblock [180] observe that most blocking schemes have been created by deciding for a set of attributes and a blocking method ad hoc. They present a machine learning approach that can automatically learn suitable blocking schemes.

Blocking methods that rely on matching or sorting according to a certain feature are problematic for the integration experiment. Most of the features are either not granular enough or contain tokens that are not in a consistent order. Therefore, a comparable blocking key cannot be easily generated. For example, since the order of hierarchical elements in the names of collections that archaeological objects are curated in is not standardized, a token-based extraction had to be performed beforehand (e.g., name of the museum and name of the city). Baxter, Christen and Churches [16] compare different blocking methods that are based on indexing. They observe that modern methods such as bigram indexing and canopy clustering significantly perform above average in most scenarios because they do not rely on the order of unprocessed tokens. As canopy clustering is used in the Apache Mahout project, it is available as an open source software library.

A number of the methods mentioned here have been developed in different

information integration projects. Compared to other techniques, canopy clustering is lean, effective and efficient. The fact that an open source implementation of canopy clustering already exists is a good argument for starting the information integration experiment with this technique. Data mining methods such as canopy clustering rely on correct parameterization. The experiment starts with reasonable parameter values, but an automatic optimization of these parameters could be implemented in the future.

8.3.2 Matching Approaches

Blocking is an important aspect of any entity resolution framework, as it dramatically increases efficiency. Different blocking strategies were discussed in the previous section. They can be combined in order to achieve optimal prescreening results. A number of approaches to matching itself have already been elaborated in several other information integration projects, which are each dedicated to a particular domain. Many modern datasets tend to provide a considerable amount of links between interrelated objects. Therefore, the suggested models perform collective entity resolution instead of pair-wise approaches. Additionally, these models facilitate the combination of multiple decision models in different ways.³⁷

This is particularly relevant for cultural heritage entities with rich and structured contexts. This requirement has been elaborated as part of the introductory user scenario in section 2.3. Considering the above-mentioned techniques for the process of data mining could yield better and more significant results. In particular, entity resolution benefits from this approach because the object descriptions that refer to the same entity may have a similar link structure. This section provides an overview of the modeling approaches that are used by different entity resolution architectures.

Most approaches to entity resolution deal with the descriptions of rather homogeneous entities, such as bibliographic entries that comprise information about authors, institutions and titles. These approaches focus on certain entity types and their relation to other entities that are treated as elements of the entity description itself. However, in the cultural heritage domain, a large number of very heterogeneous entities must also be considered. Dong, Halevy and Madhavan [85] present an entity resolution approach for the domain of Personal Information Management (PIM), which only considers the entities of several classes with scarce descriptions. This approach strives to resolve descriptions of multiple entity types and uses a dependency graph to propagate resolution decisions. In this graph, each node represents similarity information for a pair of entity descriptions, and the edges

³⁷Köpcke and Rahm [160] have evaluated a number of entity resolution frameworks that make use of the techniques mentioned in this section.

represent the dependency of multiple coreference decisions. This enables the algorithm to exploit associations between references, to propagate information about resolution decisions and to merge references for enriching attribute information.

For the implementation of an entity resolution framework, it must be determined whether a machine learning approach should be applied or not. Entity resolution frameworks like MOMA [246] provide means to manually configure a workflow that involves matchers and combiners. This does not necessarily involve the application of machine learning methods. Chen, Kalashnikov and Mehrotra present an example of a framework that uses multiple diverse learning methods [56]. The different paradigms for learning fall into the categories *supervised* (e.g., the Active Atlas system [241]), *semi-supervised* (e.g., the approach suggested by Bilenko [36]) and *unsupervised* (e.g., the approaches evaluated by Hassanzadeh et al. [131]).

Machine learning approaches rely on sample data for training. One way to generate a set of training examples is to manually or semi-automatically identify pairs of entity descriptions that refer to the same entity. Although certain heuristics and clustering algorithms can help to prescreen the entity descriptions, this is most often described as a laborious and resource intensive task. Sarawagi and Bhamidipaty [213] propose an interactive workflow that helps to select the training pairs that maximize the information gain for the machine learning model. Bilenko and Mooney [35] further discuss the challenges and approaches for training data generation. Bhattacharya and Getoor [30] describe an unsupervised sampling algorithm that also considers the relations between entities. Christen [57] presents an additional approach that allows for automatic training data generation.

To classify entity resolution approaches, one could also distinguish the deterministic models from the probabilistic models. Deterministic approaches work with fixed rules and usually categorize pairs of entity descriptions as matches or non-matches. The pairs in need of additional treatment or review by domain experts can also be categorized by being tagged. Thor and Rahm [246] use a combination of matchers and hand-crafted workflows to resolve the entities that do not explicitly use machine learning.

Probabilistic approaches use methods that refer to probability theory to determine whether a set of entity descriptions match. In contrast to deterministic approaches, the result of the matching process is not a clear categorization but is expressed as probabilities. Probabilistic models can be further categorized as being discriminative or generative. Discriminative models like Support Vector Machines or Conditional Random Fields [164] mainly rely on conditional probability distributions to model the dependence of variables. Generative models like the Hidden Markov Model, Naive Bayes or the Latent Dirichlet Allocation [42] make use of joint probability distributions. Figure 23 shows probabilistic relational ER models as part of a hierarchy of different approaches to entity resolution.

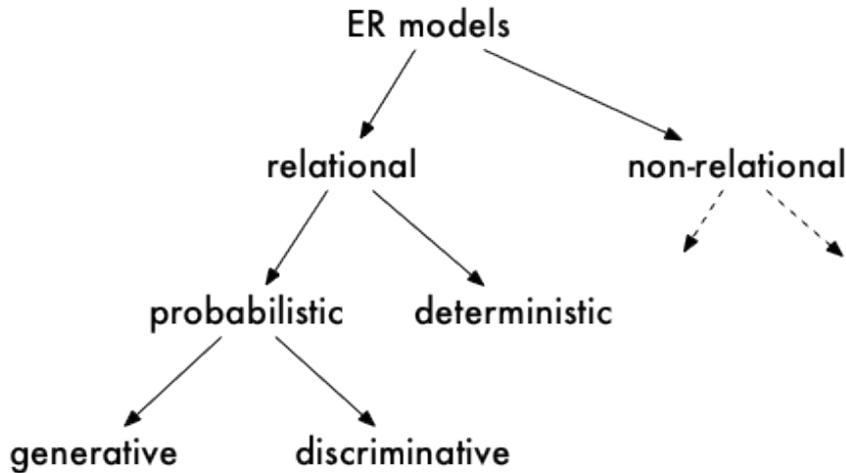


Figure 23: Probabilistic approaches in the taxonomy of relational ER models.

Singla and Domingos [229] propose an approach that is based on Conditional Random Fields. Because matching decisions are based not only on pairwise comparisons, this approach also belongs to the class of collective models that will be introduced in more detail below. A collective decision is made for a set of candidate pairs by propagating information through a graph model. The shared attribute values that are used in the propagation process enable more informed decisions for further candidates. Graph nodes are used for pairwise comparisons (record nodes) and attribute similarities (evidence nodes). Edges in the graph represent the value dependencies of attributes. By merging evidence nodes and introducing auxiliary information nodes, the model propagates information about the matches. To drive the probabilistic inference process of CRFs, certain model parameters can be learned if the appropriate training data is provided.

Wellner et al. [254] pursue a similar approach. Colutta and McCallum [72] present an approach that is also based on CRFs, but which tackles some of the shortcomings of prior approaches. More specifically, their model supports the resolution of multiple entity types. Bhattacharya and Getoor [30] have extended the Latent Dirichlet Allocation Model for entity resolution. They aim to develop a probabilistic model for collective entity resolution. However, their model is limited to homogeneous entities, and extensions would need to be developed to deal with multiple entity types.

While traditional entity resolution approaches rely on a pair-wise treatment of entities, newer approaches adhere to collective resolution strategies. Pair-wise strategies determine the similarity of pairs of entity descriptions and often depend on a blocking method to gain efficiency. Some approaches consider descriptions of

more than one entity at the same time, and others explicitly model the relations between entities for the resolution process. These relations can be represented, for example, as hierarchies or graphs. Ananthakrishna, Chaudhuri and Ganti [5] present a framework that exploits hierarchical structures for entity resolution. Puhmann, Weis and Naumann [203] suggest a framework for the deduplication of hierarchical XML data.

Bhattacharya and Getoor [31] have explored a relational clustering approach which jointly resolves entity descriptions. Similar to traditional approaches, the authors observe that two references with similar attributes are more likely to refer to the same entity if they are linked by similar descriptions of entities. The proposed method takes an input graph. In this graph the nodes denote references, and the links between these nodes denote possible coreference relations. This reference graph is then transformed into an entity graph, where each node corresponds to an entity. The links in the entity graph correspond to the real-world relationships among entities.

This approach has been split into two sub-problems: the identification problem and the disambiguation problem. In this context, the identification problem is to discover references that possibly refer to the same entities. The disambiguation problem, on the other hand, is concerned with references that are similar but do not refer to the same entity. Here, the identity of each reference depends on the identity of other references and vice versa. This results in a ‘chicken-and-egg’ problem because the algorithm must be provided a node to begin with. The authors attempt to start with less common references, where one can be more confident about the referred entity. An algorithm that is based on relational clustering then incrementally constructs the whole entity graph while referring to the coreference information that has already been discovered.

Chen, Kalashnikov and Mehrotra [55] interpret entity resolution as a challenge in data cleaning and as part of the preprocessing step of data mining. They propose a method that not only considers object attributes but also additional semantic information that has been explicitly modeled. In particular, this semantic information is represented as inter-object relationships that need to be included in the analysis. The approach is based on the Context Attraction Principle (CAP):

The CAP hypothesis:

- if two representations refer to the same entity, there is a high likelihood that they are strongly connected to each other through multiple relationships, implicit in the database;
- if two representations refer to different entities, the connection between them via relationships is weak, compared with that of the representations that refer to the same entity [55].

Getoor and Diehl [115] have described collective entity resolution as part of *link mining*. They define *link mining* as “data mining techniques that explicitly consider [...] links when building predictive or descriptive models of the linked data.” Link mining describes several methods that focus either on the object itself and consider its links to other objects, focus on the links that connect different objects or focus on complex graph-related tasks. Graph-related tasks deal with groups of objects that are related by a certain link structure. Here, the whole structure is the object of analysis, for example, the input and output of cluster analysis. A link-related task would be link prediction, which tries to predict if two objects are linked on the basis of their observed attributes and existing links to other objects.

Many of the approaches to entity resolution that have been discussed so far use a single decision model. To enhance the performance of the decision component, approaches have been explored that apply more than one model for the identification of coreferences (ensemble learning). For example, Rokach [209] has provided an overview of ensemble learning methods for enhancing the results of models that rely on machine learning. Zhao [266] has observed that several ensemble methods have been used for entity resolution, among them are bagging, boosting, stacking and cascading. While bagging and boosting combine a set of homogenous classification models to make a decision, stacking and cascading combine a set of heterogeneous classifiers, either horizontally or vertically.

However, the application of multiple models to a decision problem leads to additional usage of computational resources. Thus, decision models with favorable computational characteristics like decision tree learning should be preferred for ensemble learning. Breiman [46] describes an approach referred to as *random forests*, which has become rather popular for combining multiple decision tree learners. Random forests are a type of bagging where many of the decision trees are trained with different randomized samples. Each tree casts a vote for the classification of an item, and all of the votes are then combined to produce a final decision.

A strain of research that is related to entity resolution explores the methods and techniques that predict links in the entity graph, not the reference graph. Getoor and Diehl [115] define link prediction as “the problem of predicting the existence of a link between two entities, based on attributes and other observed links.” For example, these techniques should determine the relation of friendship in a social network on the basis of observed features and other relations, including existing friendship relations. However, Rattigan and Jensen [207] observe an extreme class skewness of connected pairs in comparison with disconnected pairs. Thus, link prediction is a difficult endeavor because the prior probability for connected pairs is usually very small.

The links that are discovered by link mining techniques can be interpreted as semantic relationships. Thus, by predicting links between entities, existing semantic networks can be enriched. While entity resolution describes the methods and techniques that mine for co-referring object descriptions, link prediction focuses on the relations between the entities themselves. For example, the CIDOC CRM has defined a set of relations that can be modeled as links in a graph representing related pairs in the world. The CRM properties that can be applied to physical things (`crm:E18.Physical_Thing`) are, for example, the former or current location of the physical thing (`crm:P53.has_former_or_current_location`) or the material that it is made of (`P45.consists_of`). In this case, link prediction could help to impute the missing relations for objects that only have a scarce amount of information associated with them.

The versatility of these approaches indicates that the entity resolution landscape is rather diverse. The number of proposed methods and techniques for entity resolution is too high to be exhaustively explored in the course of ALAP. Therefore, the experiences of other entity resolution projects are considered with respect to their applicability to the data of Arachne and Perseus. Ultimately, a compromise must be found between the rather complex and powerful techniques and the straight-forward but weaker models. Complex and powerful techniques have the advantage of being more appropriate for the data that is subject to integration. However, these methods can also be difficult to implement because no open source implementations are available and no implementation experience has been published so far.

A growing number of modern information systems deal with data that is organized according to richly interlinked structures, which can be expressed as networks or graphs. In addition to attribute information, considering the relationships between entities adds valuable information, which should be considered for the entity resolution process. Thus, both probabilistic and deterministic models are explored to perform collective entity resolution, because these models can concurrently resolve entity references, in contrast to traditional pair-wise approaches. These newer models are assessed as very powerful for information integration in the field of cultural heritage; however, the modeling effort for the process is quite high. Most of these models concentrate on certain entity types, and their efficiency has not yet been estimated for vast datasets with heterogeneous entity types.

Because research in the field of entity resolution is highly interdisciplinary, it is becoming increasingly difficult to keep track of the different approaches. Therefore, this section summarizes some of the distinguishing features of entity resolution. The information integration experiment starts with well-known and established techniques, such as decision trees and random forests. Then, the elaborated framework is further enhanced by including more advanced and complex techniques and

by experimenting with different combinations.

8.4 Evaluating the Success of Entity Resolution

The previous sections elaborated the various alternatives for composing entity resolution architectures. For any information integration project to be successful, it is important to measure and compare the performance of different approaches. And these projects should be able to iteratively test their own efforts, which allows for the revision and optimization of the choice of components and their composition. This section introduces different means, which, for the most part, stem from the field of information retrieval. These means are used to assess the success of information integration and entity resolution in particular.

Information retrieval involves establishing the conceptual and technical means to satisfy the information needs of users. Therefore, information retrieval's main concern is to identify a set of documents within the large document corpus that is relevant in specific contexts. A number of projects in this area have put considerable effort into measuring the success of information retrieval processes. In this context, the notions of precision and recall in the field of information retrieval are discussed in detail in the following. The number of relevant and retrieved documents in relation to the total number of documents is an accepted quality measure of IR systems.

Precision measures the ratio of retrieved relevant documents in relation to all retrieved documents.

$$\text{precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|} \quad (11)$$

Recall indicates the ratio of retrieved relevant documents in relation to all relevant documents.

$$\text{recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|} \quad (12)$$

In some cases it is favorable to condense precision and recall to one single measure, the f-measure. The f-measure is a weighted average of precision and recall that produces values between zero and one. Higher values indicate that precision and recall have higher values:

$$F = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (13)$$

The f-measure is useful in situations where one needs to find a compromise between precision and recall. An algorithm could either be very strict with the selection of relevant documents or it could be very tolerant. In the first case,

precision would be very high but recall rather low. In the second case, recall would be high but precision very low. By combining both values with the harmonic mean, low values for either precision or recall would have a greater effect than by simply combining them with the arithmetic mean.

Herzog, Scheuren and Winkler [138] provide a formal notation for measuring the success of entity resolution as a probabilistic interpretation. They also provide metrics from the field of information retrieval to measure the success. In a hypothetical integration scenario, it is assumed that two data sources A and B provide data records. If record $a \in A$ and record $b \in B$ represent the same entity in the world, the pair (a, b) can be considered as a matching pair. M is a set of matching pairs with a non-trivial identification.

$$M = \{(a, b) : a \in A, b \in B, (a, b) \text{ is a true matching pair}\} \quad (14)$$

U is a set of records that does not refer to the same entity in the world. Non-matching pairs can often be established easier than matching pairs.

$$U = \{(a, b) : a \in A, b \in B, (a, b) \text{ is not a matching pair}\} \quad (15)$$

Both M and U denote a partition of the cross-product space $A \times B$, which is the combination of all records in A with all records in B . Matching pairs can be identified by making use of these different techniques. \tilde{M} refers to a set of pairs that has been identified as a match by some algorithm.

$$\tilde{M} = \{(a, b) : a \in A, b \in B, (a, b) \text{ is designated as a matching pair}\} \quad (16)$$

\tilde{U} denotes a set of pairs that has been identified as not representing the same entity in the world.

$$\tilde{U} = \{(a, b) : a \in A, b \in B, (a, b) \text{ is not designated as a matching pair}\} \quad (17)$$

In practice, $\tilde{M} \neq M$ and $\tilde{U} \neq U$ and a couple of metrics have been proposed to quantify and measure these deviations.

The false match rate is the probability of falsely identifying non-matches as matches. In contrast, the false non-matching rate is the probability of falsely identifying matches as non-matches.

$$f_{\tilde{M}} = P[(a, b) \in \tilde{M} | (a, b) \in U], \quad f_{\tilde{U}} = P[(a, b) \in \tilde{U} | (a, b) \in M] \quad (18)$$

The notion of precision and recall can be applied to the problem of entity resolution as well. The following equation expresses precision and recall as conditional

	matching	non-matching
designated	true positive	false positive
not designated	false negative	true negative

Table 3: A contingency table to evaluate the success of a classifier for entity resolution.

probabilities:

$$\text{precision} = P[(a, b) \in M | (a, b) \in \tilde{M}], \text{ recall} = P[(a, b) \in \tilde{M} | (a, b) \in M] \quad (19)$$

If these conditional probabilities are expressed as fractions, precision and recall appear in their original interpretation.

$$\text{precision} = \frac{|(a, b) \in M \cap (a, b) \in \tilde{M}|}{|(a, b) \in \tilde{M}|}, \text{ recall} = \frac{|(a, b) \in M \cap (a, b) \in \tilde{M}|}{|(a, b) \in M|} \quad (20)$$

As emphasized, the core task of an entity resolution framework is to decide whether the two entity descriptions that refer to the same entity. System engineers need to sound whether this decision can be implemented as a problem of classification. There are different measures of success that can be applied to classification algorithms in communities that are concerned with machine learning. In this area, contingency tables have been proposed to judge the decision of a classification algorithm for each classified entity description. Table 8.4 shows a simple contingency table for a classifier that classifies instances into two categories (matches and non-matches). In this example, four cases need to be distinguished: true and false positives as well as true and false negatives. According to this scenario, precision can be calculated by $\frac{t_p}{t_p+f_p}$ and recall by $\frac{t_p}{t_p+f_n}$.

The information retrieval community has also elaborated extensions to the above-mentioned measures. For example, precision-recall diagrams plot the precision in relation to the increasing rates of recall after sorting a set of documents (designated matches) according to their determined relevancy. If an adequate similarity measure has been chosen, the diagram will show a significant drop at a certain recall rate. This form of visualization can be helpful for finding correct similarity thresholds. Hassanzadeh and Miller [132] propose using probabilistic queries for datasets in cases where the entity resolution strategy is unclear. They make extensive use of a kind of diagram that combines precision, recall and f-measure in relation to different similarity thresholds. Christen and Goiser [59] give an overview of issues associated with measuring the quality of entity resolution results. In particular, they emphasize that the measures should consider

precision and recall to avoid deceptive results. Measures relying on single numerical values must be applied and interpreted properly.

All the approaches for measuring the success of entity resolution that have been discussed so far require perfect knowledge about the information that is subject to integration. Therefore, the information retrieval community has been testing information retrieval systems on reference collections that have been manually labeled or created synthetically. Such datasets should also be available for entity resolution systems that focus on the data related to the humanities and cultural heritage content. But even for the field of entity resolution in general, only a few very domain specific and specialized datasets have been created and published. Naumann and Herschel [184] provide an overview of available datasets and emphasize how difficult it is to find relevant datasets. And they also mention that even if common datasets are available, it is still difficult to compare and benchmark different approaches to entity resolution due to “lack of algorithm documentation”, “different testing environments” and “obscure methodology.”

A thorough evaluation of the entity resolution results turns out to be very hard if not impossible. A useful infrastructure, one which allows for the evaluation of entity resolution performance, is not even in place in the popular domains. It is even less likely that the domain of cultural heritage or ancient history would be able to create such reference datasets. However, the entity resolution approaches that allow for the manual review of matching decisions as part of their workflow can certainly accept lower precision in favor of higher recall. But a manual review of the entity description pairs that have been suggested by a matcher is resource intensive if the expected amount of matches is rather high.

Naumann and Herschel [184] argue that there is not only a tradeoff between precision and recall but also between the two measures (that represent effectiveness) and efficiency. For example, smaller partition sizes increase efficiency in favor of recall. If entity descriptions are sorted into a raising number of partitions, the probability that entity descriptions referring to the same entity are being put into different partitions also raises. Figure 24 illustrates the coherence of precision and recall with efficiency.³⁸

For the time being, it is difficult to establish a reference collection for the humanities in general and for ancient history in particular. The amount of data that needs to be considered is immense. The complete verification of all pairs has quadratic complexity. Thus, an approach for assessing the quality of information alignment should be twofold. A reference collection could be created synthetically by using and contaminating data that has been extracted from Arachne. This data could then be used to tune the entity resolution framework until satisfactory results are reached. Subsequently, the framework could be applied to data that has

³⁸The figure extends [184].

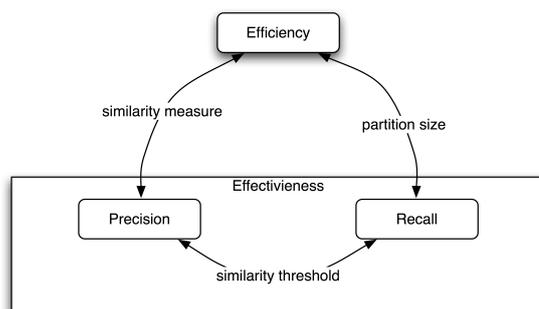


Figure 24: The relation of precision and recall to efficiency.

been provided by both Arachne and Perseus. It may also be possible to create a reference collection on a reduced data set. However, if the number of samples that are reviewed by domain experts is reduced, the expected number of coreference relations is quadratically reduced as well. This raises the likelihood that the sample will only contain very few duplicate entity descriptions or possibly none at all.

8.5 Complementary Approaches

The state-of-the-art techniques of entity resolution have already been introduced and discussed. Different alignment projects have applied these techniques to information that has been extracted from multiple data sources. The primary aim of these techniques is to find entity descriptions that refer to the same entity, with precision and recall being as high as possible. This section touches on further approaches that either rely on auxiliary information or make use of new methods.

It is almost always useful for entity resolution frameworks to have access to auxiliary background information. This additional information can be considered as auxiliary because it is not part of the information that is subject to alignment. For example, string metrics can be enhanced by allowing the entity resolution frameworks access to lookup tables, which resolve tokens into common abbreviations. The cost of transforming a token into an abbreviation is thus significantly reduced. With the help of the lookup table, a string replacement then counts as a regular character insertion, deletion or replacement. In addition to the various controlled vocabularies, which have been explicitly encoded and editorially maintained, other forms of external information could be considered.

In fact, a couple of APIs are now available to the public. They allow for the acquisition of different forms of information, such as structured bibliographic records or geographic entities. A promising field of research that is gaining momentum attempts to capture the semantics from online user behavior. Feedback mechanisms are used to enhance the quality of search results, and *clickstreams* have been

used to relate the content that has not yet been explicitly related. Complementary to using additional background information, techniques from related fields can be exploited. For example, link mining could be used to generate a graph-like link structure that relates different entities to each other. This structure could be analyzed to generate additional information for entity resolution. Furthermore, some cultural heritage information systems need to deal with spatial and visual information. For these cases, research in the field of multimedia information retrieval could contribute valuable methods.

The ideal information integration system of cultural heritage information exploits as many methods and sources of background information as possible. But simply accumulating as much information and methodology will probably not lead to the desired results. Information integration projects should carefully assess the applicability of techniques to the type of information that should be integrated. Additionally, suitable background information, which enhances the results of entity resolution, may or may not be available. In certain cases it may even be justifiable to invest resources in creating and maintaining specialized background information in the form of structured and controlled vocabularies.

8.5.1 Controlled Vocabularies

Libraries have traditionally invested considerable amounts of human resources into establishing and curating authority files. But for digital collections with massive amounts of digital born data, these standards of curation cannot be manually achieved. This problem is not only related to the issue of vastness but also to the challenge of massive heterogeneity. While traditional libraries work mainly with bibliographic records, digital cultural heritage information systems allow for the retrieval of diverse entity descriptions. This influences the number and volume of controlled vocabularies which need to be established and maintained.

However, Sieglerschmidt [225] emphasizes the importance of making use of structured vocabularies for cultural heritage information alignment projects. The author focuses on how knowledge organization and alignment can be supported by such vocabularies in an international environment. Consequently, a growing number of initiatives concentrate on the different ways to encode and publish structured vocabularies. Smaller communities will probably also establish and publish vocabularies to enhance information management and access within the scope of their projects.

As mentioned, string metrics do not genuinely identify semantic relations, like synonymy and homonymy. In these situations, background information that exhibits a certain structure can be used to alleviate the discussed shortcomings. Controlled vocabularies that model semantic relations between distinct terms also allow for certain types of inference. For example, possible semantic relations be-

tween different terms in a thesaurus could be inferred and used for information alignment. However, the same term can have multiple occurrences in a controlled vocabulary, and it is not always possible to disambiguate its meaning by exploiting additional contextual information. Thus, more than one semantic relation can be discovered, which needs to then be considered and weighted.

It seems that the computational complexity of this endeavor is relevant for the efficiency of the whole entity resolution process. However, to assign different likelihoods to discovered semantic relations and to calculate a similarity value by using the likelihood, seems to be an interesting and helpful approach. A similarity function that makes use of controlled vocabularies with rich semantic structure could be used to combine additional contextual information with a token. This would help to find the corresponding concept in the vocabulary and result in improved disambiguation and string comparison. However, in many cases semantic relations and contextual information are not always explicitly represented, and the algorithms may lack clever heuristics.

There are a number of useful, published vocabularies for the cultural heritage and museum community. A couple of institutions have made a set of vocabularies for download available under the notion “Museumsvokabular” [127]. The project WissKI (German abbreviation for Scientific Communication Infrastructure) maintains a list of controlled vocabularies for the names of events, places, actors and subjects [260]. The German National Library and the Library of Congress maintain large authority files for the names of persons [198] [190]. Comprehensive vocabularies that focus on information about places and their relations have been developed as gazetteers. For example, the Geonames project provides online service to query vocabularies of place names [255]. The Getty Foundation has published a thesaurus of geographic names online [148]. Moreover, controlled vocabularies that comprise multiple entity types are being developed or have already been published [197, 245].

Heterogeneity could also be an issue if auxiliary external information needs to be exploited. Different means of internal representation, technical heterogeneity and particular scope could result in further information integration issues. For a few of the online vocabularies, the established means may allow for certain querying and / or processing. Due to the observed heterogeneity of vocabularies, one development stands out in the area of Semantic Web research, the aforementioned Simple Knowledge Organization System (SKOS). While the Web Ontology Language (OWL) has been crafted for expressing complex conceptual structures, SKOS intends to provide a more straightforward approach to publishing multilingual structured vocabularies. SKOS builds upon the foundation of RDF, which makes the seamless processing of vocabulary data and database content possible.

Consequently, initiatives such as “Museumsvokabular” publish their vocabu-

larities as XML, HTML and SKOS. By relying on an established standard, at least certain issues with heterogeneity can be avoided due to the formalized syntax and semantics. Additionally, Binding and Tudhope [38] describe a service that adds behavior to these published thesauri comprising search, browsing and semantic expansion across structured vocabularies. The first steps to integrate this service and to establish semantic interoperability have already been taken [37].

Although a number of initiatives publish structured controlled vocabularies online, their accessibility for algorithmic processing differs. Some published vocabularies are accompanied by an API, which allows for certain operations like searching, browsing and retrieval. Other vocabularies are available in a machine-readable format and can be downloaded. The needed operations can then be implemented by the developers of an alignment system. However, some vocabularies are only available for human searching and browsing. Although these could be a useful resource in certain settings, they are not useful for (semi-)automatic information alignment.

The usage of auxiliary information from external sources heavily depends on its relevance and availability. In some situations it may be justifiable for smaller communities to craft their own specialized vocabularies for alignment projects. In turn, this information could also become relevant for other related communities and foster information sharing among neighboring groups. However, the use of these resources could result in additional issues of heterogeneity and introduce the issue of computational complexity. ALAP does not initially rely on external information sources to form a first impression of its viability. But auxiliary information is integrated later on to enhance the entity resolution quality.

8.5.2 Other Resources

The previous section discussed how auxiliary information from external sources can be exploited and used for entity resolution. Of course, it is helpful to have external information that is already standardized and in an available explicit structure, so that complex matching operations are possible. But a number of information sources provide information which bears only implicit or vague structure. Additionally, the data mining community has elaborated techniques that help to formulate automatically structured information by monitoring the behavior of users. These approaches can be exploited for entity resolution if explicit and structured vocabularies are not (yet) available.

Some information systems have published their structured information along with suitable algorithms, which define certain operations like querying and browsing. Thus, entity resolution systems can use the external APIs that have been provided by different institutions to exploit explicitly structured information. Information systems like Freebase [119] provide massive amounts of common sense

information that is structured and that can be used by external clients in a database-like fashion. But these auxiliary resources can also be useful even if they do not (mainly) rely on structured vocabularies. For example, Microsoft provides an API[181] that can be used for the purpose of inter-language translation. And Amazon [3] provides a public Web Service that allows for querying its bibliographic information. Websites like “ProgrammableWeb” [201] give a thorough overview of APIs that have been made available for public use.

However, the use of external resources may result in additional challenges for many reasons. For example, the misconceptions about the effectiveness and efficiency of networks need to be considered, which may result in the decrease of efficiency. In a worst case scenario an external service that is not working at all may bring the entity resolution process to a halt. Additionally, control cannot be exercised over these external services to the same extent that the internal components can be managed. Even small changes to the calling convention or the information itself can have unpredictable effects on the entity resolution process. The APIs may be subject to frequent change or the service may even be discontinued.³⁹

Institutions that provide information and services for the Web use monitoring techniques to track user behavior. This information is then used to learn about the information needs of users in a way that allows them to further enhance user experience. This knowledge provides information to users that is of higher relevance, based on their prior behavior. Monitoring user behavior can also be exploited to generate meaningful information from the way that users walk through information systems. For example, Baeza-Yates [13] reflects on mining user behavior on the Web to find relations between previously unconnected bits of information. An interesting approach for the domain of archaeological information systems could be to implicitly observe user behavior by logging the aforementioned clickstreams. If a user navigates from one entity description to another, the probability that they are related in some way (e.g., by coreference) could be increased.

It is a common practice for certain websites to offer elements of interaction and collaboration with increasing functionality and complexity. This enables closed or public communities to collaborate on archiving goals that would be difficult or impossible to achieve with a small number of people. For example, an archaeological information system could provide a user interface that displays similar entity descriptions that have been extracted from different information systems. Then, domain experts could cast votes on whether the different entity descriptions refer to the same entity. These votes would influence whether a coreference link should be weakened or strengthened. Thus, entity resolution could benefit from means

³⁹For example, Google has declared that their Translate APIs will be discontinued and replaced by a paid version.

that encourage the larger communities to work together towards a common goal.

So far, techniques have been discussed that focus on information that is encoded as textual or structured data. It has also been emphasized that this data should be available in a standardized and machine-actionable way. But many information systems manage considerable amounts of data like images, audio files and maps in combination with spatial data. For example, most cultural heritage databases like Arachne and Perseus consider images of archaeological entities as first class information. Additionally, images provide information about the association of objects with one or more places or more complex geographic entities. Thus, indexing methods and similarity functions for spatial information and information that is represented as signals seems to be very helpful. Faloutsos [102] describes means that have been elaborated in the field of signal processing for retrieving multimedia information by content. In addition to the established methods that have been explored by the database community, including the retrieval of spatial information, Faloutsos describes various methods to extract relevant features from signals in order to compare database content.

Different approaches of generating and exploiting auxiliary information for entity resolution have been discussed in this section. External resources that provide information that is either explicitly annotated or modeled in a more implicit way should be harnessed if available. Useful information can also be created by the conscious or unconscious collaboration of end users who use cultural heritage information systems. Additionally, valuable information for entity resolution can be extracted from signals that represent images, audio or other multimedia content. However, additional effort needs to go into the discovery, development and parameterization of software that supports the mentioned techniques.

It is worthwhile for information alignment projects to perform careful exploratory data analysis. This is a significant preliminary step for identifying valuable sources of information for entity resolution. In a number of cases, the additional effort of implementing complex software to monitor user behavior or for feature extraction from signals may be justifiable. For the information alignment experiment, monitoring user behavior could reveal entity descriptions that are frequently accessed together. Additional features could be extracted from visual information to determine the similarity of images. However, for the time being, this information is excluded because of the associated complexity of the required models.

9 An Alignment Experiment

The previous sections have surveyed and elaborated a methodology for approaching the information alignment problem of cultural heritage information systems. This section aligns the entity descriptions that are managed by two different cultural heritage information systems, Arachne and Perseus, thus putting the selected methods and techniques into practice. The aim is not to elaborate a perfectly working alignment system but to walk through the necessary implementation steps in order to identify challenges and opportunities for further research. The discovered problems are discussed, and the recommended approach for the implementation of an alignment system is documented.

The description of ALAP and the implementation process focuses on two main aspects. The first part focuses on the sources that are involved in ALAP. The features of each data source, which were partly determined by exploratory analysis, will be described from a birds-eye-perspective. This serves to identify the overlapping aspects on a very basic level, and it gives first insights into the structure and content of both sources. Subsequently, a couple of data elements that intuitively seem to be suited for ALAP will be selected for further analysis. Finally, methods of exploratory data mining will be applied to both Arachne and Perseus to obtain more insight into how the managed material is distributed.

The second part introduces a couple of heuristics that help with bootstrapping the alignment process and with training machine learning models. Several methods will be introduced, which range from simple SQL statements to more complex indexing approaches and search heuristics. The pairs of entity descriptions that are discovered in this bootstrapping step will be further analyzed. For example, with regard to the entity descriptions that belong to the bootstrapping sample, it is helpful to extract as much normalized information as possible. Then, one or more machine learning models will be trained by using the co-referring entity descriptions that were generated in the bootstrapping process.

Software has already been developed for some of these tasks as part of ALAP. Additionally, third party software is used, either as library elements or as service called over the network. These components have been put together to form the architecture of the information alignment experiment. The following sections will motivate architectural decisions and describe the way that the components have been implemented. The identified shortcomings will be addressed and considerations for further research will be discussed.

The following chapters do not describe an approach that is pursued in strict sequential order. The experiment is carried out in an iterative manner beginning with intuitions, which have been tested and validated by means of exploratory data analysis and preliminary matching. Instead of documenting every iteration, the documentation highlights the significant progress that is made at different

stages of the process. Although it has already been emphasized that information integration for Arachne and Perseus is conducted as an “experiment”, it is difficult to measure success or failure. A number of other information alignment projects have recognized the difficulty in measuring the matching quality without having access to almost perfect reference collections.

Because of the aforementioned situation, the search for a reasonable interpretation of the results is challenging. Although the chosen approach does not work with precise accuracy measures, the results of it can still be derived and beneficially interpreted. Matches that are designated by the matcher are further studied to reveal the reasons for correct and false matching decisions. The analysis of individual matching decisions also shows information about the advantages and disadvantages of the chosen matching methodology. Additionally, the studied examples reveal how the peculiarities of the data (e.g., data quality problems) can influence a decision process.

9.1 Experiment Design and Software Development

A combination of third-party and self-developed software is used for ALAP. These are combined to form a software architecture and system infrastructure that enables implementation with reasonable effort. The resulting framework could be used to scaffold future alignment implementations for cultural heritage information. A methodology similar to rapid prototyping and agile development is used in ALAP. Traditional software development processes are biased towards larger projects with many developers. This section discusses why certain design decisions were chosen for ALAP.

Architectural and infrastructural decisions influence the effectiveness and efficiency of a whole software system. Leser and Naumann [168] provide a comprehensive overview of different architectures and discuss their advantages and disadvantages. Architectural designs range from being highly differentiated and distributed to simple and monolithic, which play on their various strengths in different situations. Only a few local, infrastructural elements have been developed for ALAP, but these can be extended at later stages of the development process. The design of ALAP concentrates on the offline entity resolution of well-defined sets of object representations because these architectures tend to be much more simplistic.

Comparison operations will be implemented that are supported by machine learning and other data mining techniques. These are most efficient for the information that has been “physically” accumulated in the memory systems of a dedicated system, which must be considered jointly. The algorithms that are applied to the data require frequent memory transactions that would considerably slow down if they were being performed over a network. However, it has been

mentioned that entity resolution can be parallelized by distributing partitions of the data to different systems.

To maintain high technical independence, the services that support the entity resolution process run on one physical machine. These comprise, for example, a relational database system for background storage and a Servlet container for user interface implementation. Currently, the framework does not need access to third-party services running at remote sites, e.g., for translation or knowledge representation. Thus, the implementation of ALAP does not make use of external background knowledge. All software components have been developed on one single desktop system and run with reasonable performance. In particular, this is meant to keep the running time of the overall entity resolution process under 12 hours, which allows for a minimal form of interactivity. Additionally, it is helpful to break the process down into single steps that can be completed in less than one hour. By breaking the process down into single steps, the intermediate results can be analyzed, which allows for the fine-tuning of single components.

In order to implement the ideas and intuitions that have been developed so far, multiple software components have been implemented. This is reflected in the package structure of the software components that have been developed as part of this dissertation project. Different packages have been created to organize the software for extracting, exploring, normalizing and matching. The methods that have been implemented range from rule-based information extraction to semantic exploration by latent semantic analysis. Each software component will be introduced in the order that it is used in ALAP. Functional units can be called locally, so there is no need to transfer information over a network.

It has already been emphasized that entity resolution is an interdisciplinary endeavor with various fields of research that need to be considered. To avoid unnecessary software development, a short overview of the available toolkits that provide useful functionality for entity resolution has already been given. In particular, the existing open source software that implements methods of natural language processing, machine learning, linear algebra, statistics and similarity metrics has been assessed. It turns out that high quality open source software is available in the field of machine learning. Not all of the components that are mentioned in the following paragraphs have been used, and the role of single libraries for ALAP is explicitly mentioned.

There are a number of tools available that support natural language and text processing. For ALAP, the most interesting tools are those that provide for entity extraction. A widely used framework, GATE (General Architecture for Text Engineering), has been described by Cunningham, Maynard and Bontcheva [73]. It has been made available under the GNU General Public License by the University of Sheffield. GATE comprises the ANNIE (A Nearly-New Information Extraction)

system, which can be used for entity extraction. ANNIE supports JAPE (Java Annotation Pattern Engine), an engine that allows for the formulation of patterns that are similar to the regular expressions that are backed by auxiliary data structures. GATE provides the means to construct complex text analysis workflows, and these workflows produce documents that are tagged with XML.

Further frameworks like MALLET (MACHINE Learning for Language Toolkit) and LingPipe are also interesting. MALLET has been described by McCallum [178]; it provides a Java library that implements a set of language and text processing algorithms. This functionality includes information extraction, classification and clustering as well as topic modeling. Although Ling Pipe is being developed by the commercial organization Alias-i [2], it can also be used under a royalty-free license. It allows for multiple types of named entity recognition by using rules, dictionary entries and statistical models. However, because suitable background vocabularies were not available at the time the experiment was being developed, a more simplistic approach has been pursued. Thus, the information extraction component relies on straightforward regular expressions.

Different tasks of the entity resolution approach can be supported by machine learning toolkits that are less focused on natural language processing. Braun, Sonnenburg and Ong [45] maintain a list of open source software for machine learning application programming. The list currently comprises around 380 entries,⁴⁰ and among these entries, two tools have been extensively used for ALAP: WEKA and Apache mahout.

Hall et al. [128] have developed and documented WEKA, which is an open source suite of algorithms for machine learning. These algorithms can either be combined into complex workflows with the help of a user interface or be directly used in Java source code. WEKA provides an efficient implementation of the C4.5 decision tree algorithm, supports decision forests and can be extended for the use of Support Vector Machines. Additionally, comprehensive documentation is available for both the user interface and source code. Therefore, WEKA will be used to implement the entity resolution component. Another toolkit has been developed with the support of the Apache Foundation: Mahout [242]. Mahout is a software library that provides algorithms for machine learning with a particular focus on scalability. To that end, Mahout has been tightly interwoven with Apache Hadoop [243], which is an open source library for distributed computing. At the moment, the suite seems to be biased towards mining user behavior in commercially relevant context. ALAP has implemented Canopy Clustering by the Mahout project in combination with Apache Solr [244].

In addition to open source projects that have already been presented, there are a few others that deal with entity resolution in the narrower sense. Software

⁴⁰As of February 13th, 2012.

libraries like SimMetrics and SecondString focus on providing string similarity metrics. SimMetrics was funded by the Engineering and Physical Sciences Research Council, and has been maintained as open Source Software by Chapman [51]. It comprises algorithms to determine simple edit distances and further enhancements like Levenshtein and Jaro/Winkler. But it also provides token-based and other distance functions like Soundex. Cohen, Ravikumar and Fienberg [63] describe a hybrid distance metric that has become part of the SecondString project. Similar to SimMetrics, SecondString provides a library of various string metrics, and it also includes the aforementioned soft TF/IDF algorithm.

ALAP employs algorithms that are provided by SimMetrics, and it uses the soft TF/IDF for the matching component. David et al. [79] present the Align API, which is geared towards the alignment of entity representations that form schemata. The Align API together with the Alignment Server provide the means to document and manage ontology alignments. Bernstein et al. [28] describe Sim-Pack, which is also geared towards the alignment of ontologies. Additionally, it provides wrapper functions for other projects including SecondString and SimMetrics.

Certain parts of ALAP require the performance of highly complex mathematical operations on very large amounts of data. Simple exploratory data mining has been applied to acquire a better idea of how large the overlap between Arachne and Perseus is. This approach could also be used to evaluate the relevance of sets of data elements for the alignment process. Originally started by Gentleman and Ihaka [114], the R Project for Statistical Computing has developed a free software environment for statistical computing and visualization. The tools of R have been used for certain analytical tasks throughout ALAP.

Eaton [93] has developed and documented GNU Octave, an interpreted language for numerical operations. Octave has been used to test complex matrix operations in order to achieve a better understanding of them. A very fast and efficient solution is required for production use in the Java programming environment. A set of high performance Java libraries that support linear algebra operations has been developed as part of the Colt project under the auspices of CERN [49]. Colt's efforts to achieve competitive or superior performance, especially in comparison to many other toolkits, make it an attractive choice for Java development. Moreover, Colt has proven its efficiency in large scale research projects for nuclear research.

Jurgens and Stevens [155] have developed the S-Space package, a set of tools that deals with semantic spaces and distributional semantics. Among these tools are algorithms that implement Latent Semantic Analysis for large textual corpora. However, in ALAP, Latent Semantic Analysis was implemented by using Colt directly. The term-document frequency matrix has been preprocessed by TF/IDF. The similarity between strings can then be calculated by transforming them into

the feature space and determining the cosine distance.

Another relevant aspect for this experiment is how to find efficient ways to represent internal data in a way that makes it accessible to the tools being used. As mentioned, an increasing amount of cultural heritage information is being represented according to Semantic Web standards. At first glance, it seems to be helpful to adopt these concepts and techniques for the internal data representation of the implemented entity resolution framework. This approach could avoid costly transformations and provide a straightforward way to manage data. However, in many cases, very simplistic forms of representation turned out to be more efficient and effective. LSA and Canopy clustering require more complex proprietary information representation. However, many machine learning algorithms only require simple tabular forms and represent their results in tables too. In fact, additional overhead would be created if there was a constant mediation between complex frameworks like RDF and simple tables.

Both Arachne and Perseus rely on relational databases as background storage, among others. Multiple caching-like mechanisms have been built on top of these relational databases to make the systems more efficient. Although the information of Arachne is currently being made available as RDF, comprehensive semantic information is still unavailable due to data quality issues. Thus, for the purposes of ALAP, means have been implemented to extract high quality information from both Arachne and Perseus. In some cases, regular expressions have been used to get to the internally structured information that was residing in the database field. For the time being, this is an efficient way to acquire relevant, high quality data without the hassle of decomposing the RDF/XML to conform to CIDOC CRM.

The process of entity resolution has been introduced as a workflow that breaks down the whole process into functional units that rely on each other. After extracting data from Arachne and Perseus, the first step was to create a binary representation of it, which allows for efficient clustering. The result of the clustering was then stored as comma-separated values in simple text files. Then a list of all the relevant comparisons for each Arachne and Perseus record that share the same cluster was created via relational operations in a relational database, etc. It turned out that very long text files with comma-separated values are a good way to represent intermediary results. Most operations of the alignment process do not need random access to the files; instead, they can be streamed to the main memory, chunk by chunk, without the need to create complex indexing structures.

Although Semantic Web techniques are not the first choice for internal data representation, they are relevant for sharing the results of the alignment process. The goal of ALAP is to assess and enhance the quality of entity resolution. This can be achieved by making use of different forms of visualization and iteratively enhancing software components. Nevertheless, the generated alignment informa-

tion should be shared because it could be immediately useful for other alignment projects. Because this information will most likely be used in a Semantic Web context, it should be represented as RDF triples in a standardized vocabulary.

However, it has already been mentioned that the use of OWL and RDFS vocabularies for this task is harmful because they tend to be too specific and strict. Therefore, it seems to be more adequate to use the coreference bundles that have already been introduced. Another aspect that could be interpreted both as an intermediate and final result of ALAP is the representation of semantic spaces for Latent Semantic Analysis. This information could be valuable for a wider audience that performs information retrieval and alignment tasks. Throughout ALAP, preliminary semantic spaces have been generated from labeled coreference information. Thus, the space itself – which is represented as a set of matrices resulting from singular value decomposition – should also be published in the future. This could be done with the serialization mechanism of Java or as comma-separated values.

CIDOC CRM specializes in modeling information about the relations between cultural heritage entities as graph-like structures. The advanced approaches to entity resolution that make use of contextual information and relations between entities have already been described. Thus, it is probable that RDF/XML will gain importance as a way of sharing information and of feeding it to information alignment systems. Furthermore, the need for data structures that can efficiently represent information that is modeled as a graph will become more pressing. However, there may be more efficient, implicit ways to represent information than RDF/XML. The way that algorithms operate on graph structures determines efficient representation and access. Skiena [232] suggests a few questions be asked with regard to how efficient the graph structure design is: How many nodes and edges are there? What is the density? What types of algorithms are used and do they need to modify the graph?⁴¹

ALAP builds on prior work that has been completed for the information integration of Arachne and Perseus. It quickly became obvious that the integration of entity representation would be the largest challenge once the schema matching and mapping had been solved. In particular, the fact that both data sources use different national languages to describe entities needs to be considered for software development. Nevertheless, a cursory analysis of manually picked co-referring entity descriptions revealed useful information for automatic alignment. It appears that the successful alignment of the archaeological entity descriptions from Arachne and Perseus is at least promising and not impossible.

The software components for ALAP have been developed in an iterative man-

⁴¹More information can also be found in an online repository that is maintained by the author [231].

ner. In the earlier stages, the co-reference was manually determined and simplistic scripts were written to validate the first intuitions. As more knowledge about the data and alignment methodology was acquired, the work became more systematic and the first foundations of the software development process were implemented. Although many knowledge discovery projects consider entity resolution to be a problem of data cleaning, it also requires a whole knowledge discovery workflow for itself, as argued in chapter 6. The following sections describe the issues that needed to be solved and the components that were implemented to establish ALAP.

Statistical summaries are a useful tool to estimate the overlap between the data sources with respect to content. An exploratory data analysis step has been laid out as the foundation for making decisions, such as selecting features of entity descriptions that are useful for alignment. In this case, exploratory data analysis describes both ad hoc and more systematic analyses of the data models, schemata and contents of Arachne and Perseus. Activities in this stage range from the manual inspection and comparison of database schemas to the submission of queries which identify overlapping content. Additionally, statistical summaries have been compiled for parts of entity descriptions that seem to be promising for automatic information alignment. One important outcome of the analysis is the knowledge that unveils the actual relevancy of information objects for the subsequent alignment tasks. Finally, an initial set of labeled data was compiled to help with bootstrapping the decision model.

The use of different national languages for Arachne and Perseus was considered in the process of feature selection. Thus, the features that are as language independent as possible, such as information about the dimensions of an archaeological object, were preferred. Unfortunately, the amount of language independent features present in Arachne and Perseus are rather limited. Another way to determine the similarity of entity representations is to focus on features that are part of a controlled vocabulary. In this case, terms could be resolved into a canonical representation before comparison. However, neither are specialized vocabularies publicly available nor the needed resolution services for them. Some vocabularies are rather minimal and could be translated with little effort, like the type of an archaeological entity or a description of entity's materials.

In general, a growing number of vocabulary terms have been represented by making use of the global identifiers in cultural heritage information systems. For example, accession numbers are recorded in both Arachne and Perseus for a certain amount of objects. For most of the objects, the accession number together with the name of the collection results in a unique identifier in most cases. There are certain exceptions to this rule if collections have already been merged or if the names of certain catalogs need to be considered. In the course of ALAP, accession numbers have played a major role in the creation of a set of labeled pairs of

entity descriptions, which were used to bootstrap the aligner. Also, geographic information like findspots have been matched with gazetteers to enrich the local information with globally unique identifiers. This can be very helpful in situations where multiple names have been used for an entity that cannot be treated with similarity metrics.

A number of entity description aspects make extensive use of Greek and Latin names, as well as modern English and German names, when referring to entities. For example, short descriptions of archaeological objects and information about findspots often refer to Greek and Latin names. Information about the collection that an object belongs to is recorded together with modern place names (e.g., Museum of Fine Arts, Boston). It turns out that many of the names belonging to this category can be resolved by using similarity metrics because they tend to only feature minor spelling deviations. The following sections will also deal with how to treat the selected features in more detail.

After identifying a set of relevant features, a certain level of data quality must be established. To that end, common data quality problems have been analyzed, and the means to raise data quality in problematic cases have been implemented. For the task of entity resolution, a good criterion for assessing data quality is to see if the extracted data elements form correct and usable input for the aligner. For example, fundamentally different principles have been used for Arachne and Perseus to represent bibliographic information. Therefore, methods of information extraction have been applied to the raw data to extract information that is well-suited for determining similarities.

A couple of the (machine learning) models that were introduced turned out to be successful in entity resolution environments. The tradeoff between the complexity of a model and how well it fits in the case of entity resolution models has been emphasized and discussed. Some projects have reported good experiences with Support Vector Machines, and others have applied complex statistical models. However, these approaches introduce a complexity that may be perfectly adequate for particular problems but that could be overkill for a first alignment experiment. Therefore, decision trees seemed to be an appropriate beginning for ALAP. They can be fed with categorical or cardinal values as input, and they produce robust results even if data is missing. Multiple extensions like *random forests* do exist and have favorable computational characteristics.

This section has reflected on the design decisions that were made for ALAP and how they have affected the process of software development. Four main topics were addressed: architecture / infrastructure, third-party software, internal data representation and the overall integration strategy. System designers working with entity resolution for cultural heritage information should consider the several alternatives: the various architectures that range from highly distributed system

components, which are highly autonomous, to more simplistic and monolithic architectures. Software components can either be implemented in-house or re-used if they have already been developed in another context. However, the latter often has the disadvantage of being a *worse fit*. A decision must also be made with regard to the different forms of information representation, which are either highly efficient or adhere to community standards (or both). On the basis of the decisions that have been made, a suitable integration strategy must be developed and implemented.

One major decision that was made for ALAP was to restrict the needed infrastructure to a local machine and to stick with an architecture that allowed for further expansion. Whenever it is possible and reasonable, third-party software was used to reduce the implementation effort. It turned out that high-quality open source software that implements different functions of the project was available. Semantic Web technology could be important for the interfaces of the systems that communicate with the “outside world”. The architecture of ALAP could be redesigned with reasonable effort to accommodate this task. For internal information representation, the data structures were chosen because they best support the operations being performed on the data.

9.2 Exploratory Analysis of Information Sources

It has been emphasized that the exploratory analysis of the models, schemata and data makes up the largest part of this work. In the following, the steps that were taken to analyze the information sources in an explorative manner will be elaborated. In the first step, multiple heuristics were used to determine the semantic overlap between the two information systems with respect to their records and attributes. This step was also helpful for selecting a set of features that would be useful for the matching process. Another aim of exploratory data mining is to measure the quality of the features that have been selected and that need to be extracted from each information system. The details of information extraction and quality measurement will be dealt with in more detail in the following. Also, criteria need to be derived in order to develop a model that will align the data from each information system.

9.2.1 Information Sources Overview

One of the first steps of ALAP was to perform an exploratory analysis of Arachne and Perseus. At the beginning of this analysis, both information systems were examined from a bird’s eye view. This included looking at the data models and how the information system’s content was organized into database schemas. Additionally, high level summaries of the actual content were compiled to decide on

the relevant parts. The following paragraphs give a short overview of the technical backgrounds of Arachne and Perseus and the way in which content is managed in these databases.

Arachne relies on a relational database as its central element of data management. The relational database is complemented with a number of components that provide for enhanced usability and efficiency. Additional software components implement interfaces for human users, and third-party applications have been introduced to effectively and efficiently deliver the data of Arachne for further processing. These comprise multiple index structures that have been established to provide fast access to all data objects. Arachne is dedicated to the effective management of archaeological entities in a highly contextualized manner. Different categories of entities, such as single objects, buildings and sites, can be explicitly linked to improve search and navigation. Currently, Arachne hosts about 250,000 objects that are associated with about 840,000 images in about 13 top-level categories [108].

The Perseus Digital Library Project provides, among others, digital collections of “primary and secondary sources for the study of ancient Greece and Rome [196].” The majority of the material in Perseus is text, which is presented to users in a reading environment that provides navigational features as well as contextual information. For example, this information comprises cross-references and translations of a particular passage of text. The cross-references are supported by a relational database that stores information about entities. These entities have already been extracted in preliminary information extraction efforts. The Perseus collection includes 5,859 objects, sites and buildings, which are presented to users for browsing. The art and archaeology database of Perseus is also built around a relational database.

Table 4 lists the estimated number of entities for different categories that can be extracted from Arachne and Perseus with reasonable effort. Reasonable effort means that entity descriptions are available at least in a partially pre-structured and normalized manner. Thus, this can be seen as the number of entities that will be extractable in the course of ALAP. In particular, the numbers for Arachne could be much higher if additional effort were to be given to information extraction, but this task is beyond the scope of the first alignment experiment. To align information, it is helpful to have a large overlap between the types of entities in Arachne and Perseus to be integrated. Additionally, entity descriptions should cover similar aspects in a dense manner, as this makes extracting relevant information for machine learning much easier.

The amounts should be interpreted as estimations of the magnitude because there may be some schematic elements that were not considered. “Site” comprises geographic entities that have explicitly been marked as topographical units.

	Arachne (public)	Perseus (a & a)	Perseus (texts)
sculpture	41 486	2 003	n/a
vase	4 600	1 909	n/a
coin	289	1 305	n/a
building	6 165	424	n/a
gem	539	140	n/a
site	4 704	78	n/a
place	(13 469)	(691)	941 335
lemma	n/a	n/a	100 564
person	218	n/a	251 984

Table 4: Estimated number of extractable entities for Arachne and Perseus.

“Place” mainly stands for the findspots of archaeological objects. The amounts of depositories and collections would be less; these are not illustrated in the table. Additionally, Arachne and Perseus treat the dates that also include data ranges and periods as entities. This information could be aligned manually if it were organized into controlled vocabularies. To align the sophisticated date information with fuzzy modifiers is beyond the scope of ALAP. The bracketed amounts refer to the entities that include textual descriptions, which may impede the alignment process.

It seems that sculptures and vases have the highest expected overlap of entity descriptions, 2 003 (sculpture) and 1 909 (vases). Although this amount would be much higher for places, ALAP treats these entity types as complementary information. These entity descriptions only have sparse descriptions and include additional information in external systems like gazetteers. Therefore, only limited descriptive information is available for determining the similarity of entity descriptions and learning good decision models.

ALAP focuses on information that has been extracted from Arachne and Perseus. Thus, the choice and parameterization of single components that make up the entity resolution framework consider multilingual information. It would be interesting to extend the approach to cover more material from additional projects that deal with comparable entity descriptions.

Arachne and The Beazley Archive actively contribute information to the CLAROS Project. CLAROS has developed a user interface that can be used to access the information for searching and browsing that has been provided by different project partners. Both Arachne and The Beazley Archive contribute a fair amount of entity descriptions for ceramics and / or pottery. This corpus of entity descriptions could be used to evaluate the architecture and workflow of ALAP.

Perseus Digital Library focuses on textual information that has been enriched with complementary information. Although a high number of entities have been

extracted from the texts of the Perseus collection, they do not seem to be suitable for ALAP. The majority of entity descriptions that have been extracted either refer to persons or to places. In fact, these descriptions are not dense because they are names, and are, therefore, too sparse for training sophisticated decision models. However, it would be beneficial to extract the names that refer to material objects, as well as the entities, and to use this information as contextual information.

Table 4 should be understood as an approximation which depends on a number of preliminary decisions. A more detailed analysis would reveal the additional means needed to enhance the entity extraction rate. However, the presented numbers can be seen as the minimum number of entities that can be extracted with reasonable effort and as a good indicator of expected entity resolution success. Because of the involved complexity, installing additional means to extract the structured entity descriptions for material objects could be seen as a research project on its own.

Since Arachne and Perseus use the relational data model to store names and structured descriptions of entities, it was possible to have standardized access to the data and to perform straightforward mapping of database schemata for entity retrieval and matching. The initial exploratory analysis of entity type partitions, which considered the amount of associated descriptive information, revealed that the descriptions of sculpture and ceramics were promising for ALAP. Future follow-up projects could also try to apply the methodology of ALAP to comparable collections. Additionally, it would be interesting to explore whether information that has been extracted from texts can also be aligned.

9.2.2 Initial Training Data Generation

A number of projects have used machine learning models to decide whether two entity descriptions refer to the same entity or not. Some problems in computer science require one to approach very complex systems by first establishing less complex systems. This helps to attenuate the dilemma of causality. This problem, which also applies to statistical learning, is commonly referred to as the *chicken or the egg problem*. Almost all approaches to entity resolution rely on a certain amount of a priori knowledge about the entity descriptions to adequately parameterize matching components. Different heuristics are used to identify possible links between Arachne and Perseus with the aim of compiling a training set to bootstrap the machine learning models. This step can be considered to be part of exploratory data analysis since it leads to a deeper understanding of how both information systems relate to each other.

Arachne and Perseus organize information about archaeological objects with the help of relational databases. Therefore, an obvious approach to align the databases would be to find the links by formulating SQL commands that per-

form joins between these information systems. Listing 1 shows an SQL command that can be used as a fairly good guess to find links between records of Arachne and Perseus. It describes a query that spans the database tables in Arachne and Perseus, and then returns pairs of primary keys that are very likely to hold information about the same entity.

The two criteria that have been used in this example are the accession number and the part of the depository information that contains the name of the city. Arachne records this information in the database field “start”, and in Perseus, the city is part of a string in the database field “collection”. Although there are differences in how some place names are written, their pronunciation is very similar. Therefore, the Soundex algorithm has been used to perform the matching in the early stages of ALAP.

Listing 1: An SQL command that finds records that a links by their depository.

```
1 SELECT ps_objektid, id FROM objekt
2 LEFT JOIN ortsbezug ON ps_objektid = fs_objektid
3 LEFT JOIN ort ON fs_ortid = ps_ortid
4 LEFT JOIN artifact ON accession_number like invnr
5 WHERE SUBSTR(SOUNDEX(stadt),1,3)
6 LIKE SUBSTR(SOUNDEX(collection),1,3)
7 AND title IS NOT NULL
```

The Soundex value of “Athens, National Archaeological Museum” is “A35...” (record from the Perseus art and archaeology database) and the Soundex value of “Athen” is “A35...” (record from the Arachne database). However, this approach is only the first attempt because of its obvious shortcomings. Unfortunately, if a place name in German begins with a different character than in English, the Soundex value will be different and the link will not be discovered: “Köln” is “K450” and “Cologne” is “C245”. Additionally, in the Perseus database field “collection”, the place name is a suffix in a number of cases and not a prefix. Consequently, the Soundex value of “Museum of Fine Arts, Boston” is “M25...” (Perseus), but the Soundex value of “Boston” is “B23...” (Arachne). Records that bear these place attributions cannot be correctly assessed – not even if they were to contain a lot of matches. However, the result of this command revealed 45 links that can be further examined.

The shortcomings of this approach have already been mentioned. However, Soundex is the only similarity metric that is implemented by MySQL 5.1; additional methods are therefore needed to perform approximate string matching.

One way to access the desired functionality in SQL is to implement additional “user defined” functions. MySQL is capable of loading object files that contain compiled C or C++ code for usage within SQL statements [228]. However, Gravano et al. [121, 122] argue that the implementation and usage of these functions can be challenging for multiple reasons. Oftentimes, functions like similarity metrics need to be applied to the cross-product of two tables, making it inefficient.

Therefore, the authors propose using auxiliary tables to implement positional q-grams because they eliminate the need to implement user defined functions.

Another way to deal with the limitations of database management systems like MySQL is to write code in a more powerful programming language and use the database as a background system. This approach is usually advantageous insofar as the libraries used for approximate string matching are already available. In this case, a division of labor occurs between a system that is good at storing and retrieving massive amounts of data and a system that is good at complex data manipulations. A matching framework would connect to a database and fetch data objects for further comparison, so that the amount of records to consider remains reasonably low. Additionally, different customized metrics could be applied to different data objects, making the introduction of conditional decisions possible.

To be sure, these approaches can be further elaborated and extended. For example, certain steps of preprocessing like clustering or the creation of index structures could be performed to make training set generation more efficient. An iterative approach for training set generation starts with rather simple means but can evolve by smoothly transiting into a more complex entity resolution architecture. These iterative steps could be the introduction of prescreening: they could group possible candidates together or they could continue to add more aspects of the entity descriptions for comparison over time.

A combination of the abovementioned approaches was used to bootstrap the machine learning process of ALAP. The amount of training data was gradually increased by applying different and increasingly sophisticated methods. At each step, all pairs of entity descriptions that refer to the same entity with a high probability were manually revised. This is important because wrong training data deteriorates the quality of the learned models. Following this approach, an overlap between about 140 records was revealed. These records have at least five attributes that qualify for determining their similarity. The training sets that have been compiled by the described approaches were used for preliminary experiments with different machine learning models. Hence, the quality and amount of training data increased with each iteration.

Because of the iterative approach of the experiment, it was difficult to clearly separate the exploratory analysis of data sources from data mining. Instead, the architecture of the entity resolution system emerged in an evolutionary manner. During this process, the exploratory parts and the application of complex models were continually differentiated. Additionally, both the discovered knowledge and the knowledge discovery methodology evolved as additional information about the kinds of data that became available. This included the process of parameterizing multiple components of the architecture so that they can function properly together. If one component or the composition of the training data changes, the

effects of this on subsequent components can be unexpectedly large. Thus, to control the iterative process, only one change at a time should be made to each iteration.

9.2.3 Evaluating Extraction Success

The quality of data mining depends on whether relevant information with an adequate level of quality is available. Machine learning techniques rely on training data, which must adequately represent the features of the data set for further analysis. The methods of exploratory analysis can be applied to identify aspects of entity descriptions that are relevant and useful for machine learning. In order to get a preliminary idea of the expected success of the knowledge discovery process, a systematic evaluation of the (expected) information extraction performance can be helpful. This section introduces the methodology of initial evaluation and discusses the results of the experiment.

A number of statistics have been compiled to determine how well information can be extracted from Arachne and Perseus. The analysis starts by looking at all entity descriptions that are available in both Arachne and Perseus, regardless of the expected effort for extraction. In the future, similar overviews should be produced to partition the entity descriptions that share certain features, like buildings or reliefs. As for the overall data set, the extraction success is also measured for the training set. Accompanying statistics reveal additional information about the relevancy of particular entity description aspects.

The aim of information extraction has been described as acquiring as much information as possible that is usable for entity resolution. This is helpful for estimating the success of future alignment experiments and for getting additional details on evaluating the relevancy of entity description aspects. Table 5 illustrates the results of naive information extraction that has been applied to all Arachne and Perseus data records. The dimensions of a material archaeological entity are easy to compare because they are represented as cardinal values. At the same time, it seems that less than half of the entity descriptions comprise information about the dimensions after information extraction has been performed. This situation seems to be quite bad for bibliographic information, but is much better for information about geographic associations. It would be valuable to extend this analysis to cover the partitions of the provided information, like the descriptions of buildings, topographical units, etc.

The approach of creating a set of pairs of entity descriptions that refer to the same entity has been described. The summary statistics that have been generated for Arachne and Perseus can also be applied to this set of training pairs. Table 6 shows the extraction ratio for the entity descriptions that make up the sample that was itemized by the contributing information system. Similar to the whole

	Arachne (70542 records)		Perseus (2003 records)	
	with content	extracted	with content	extracted
accession number	39057 (55%)	39057 (55%)	2003 (100%)	2003 (100%)
bibliography	50081 (71%)	46111 (65%)	1741 (87%)	1696 (85%)
findspot	35087 (50%)	35087 (50%)	1424 (71%)	1424 (71%)
height	37348 (53%)	35501 (50%)	1538 (77%)	1538 (77%)
location	70261 (100%)	70261 (100%)	1885 (94%)	1885 (94%)
summary	65187 (92%)	65187 (92%)	1841 (92%)	1841 (92%)

Table 5: Extraction ratio for Arachne and Perseus.

	Arachne (459 records)		Perseus (302 records)	
	with content	extracted	with content	extracted
accession number	428 (93%)	428 (93%)	302 (100%)	(100%)
bibliography	404 (88%)	393 (86%)	278 (92%)	275 (91%)
findspot	270 (59%)	270 (59%)	236 (78%)	236 (78%)
height	304 (66%)	298 (65%)	267 (88%)	267 (88%)
location	459 (100%)	459 (100%)	301 (100%)	301 (100%)
summary	448 (98%)	448 (98%)	292 (97%)	292 (97%)

Table 6: Extraction ratios for sample pairs of entity descriptions of Arachne and Perseus.

data set, these entity descriptions must be able to provide rich information that can be exploited for data mining. A further complication was the problem with missing information: if information is missing from one entity description, the entity resolution system is prevented from determining similarities.

Because values that are either missing or not extractable impede the determination of similarity values, joint statistics for the training set were compiled as well. For the activity of entity resolution, labeled pairs represent the ordered set of similarity values that resulted from the application of similarity metrics. Table 7 shows the result of applying summary statistics to the training data of this ordered set. The probability of successfully resolving entities is higher if the number of missing values is low. The lowest similarity value that has been determined is zero percent, which occurs if the set contains a number of non-matches. The mean should be lower if the number of labeled non-matches outweighs the number of labeled matches. A high standard deviation is favorable because it indicates that the similarity values are less ambiguous and tend towards zero or 100 percent.

The unpartitioned analysis of the extraction success for Arachne and Perseus shows results that vary from 38

These results are also confirmed by the analysis of the training set. Many values

	missing	min	max	mean	s/d
accession number	0	n/a	n/a	n/a	n/a
bibliography	0	0	91	14.332	19.49
findspot	0	0	100	68.479	35
height	274	1	100	87.874	23.917
location	0	0	100	13.456	26.541
summary	0	0	100	13.949	19.354

Table 7: Quality of the training sample (632 pairs).

are missing for bibliographic information, and those that are available are biased towards low similarity values. For information about the height of entities, it seems to be the other way around. These values are biased towards high similarity values. Thus, the dimensions of entities are less useful. The observation indicates that the considered entities are approximately of equal height. The short description of an entity only has a few missing values but seems to be biased towards low similarity values. Only limited statistical information is available for the accession numbers because it is modeled as a categorical value. The most valuable contribution came from the geographic information of each entity of Arachne and Perseus.

Exploratory data analysis was applied to shed light on different aspects of the structure and content of Arachne and Perseus. Mining unprocessed information helped to identify relevant aspects, find common data glitches and decide for suitable machine learning models. Furthermore, exploratory analysis can be applied to the intermediate results of the entity resolution process to tune parameters and debug applied software. The approach that has been described in this chapter was used to assess information extraction success and to optimize the extraction models. Geographic information is highly relevant for ALAP, and the other contributions remain to be explored in more detail.

9.2.4 Data Extraction, Cleaning and Normalization

Different aspects of entity descriptions were considered for ALAP. In the beginning, they were selected intuitively for further analysis, in particular, to assess their contribution to model generation. Those preferred aspects are those that are either represented as cardinal values or that are as language independent as possible. In the course of ALAP, each aspect was analyzed in more detail to optimize extraction results and to estimate its contribution to entity resolution. The following paragraphs describe exactly how these entity resolution aspects were explored.

Various information on the structure of entity descriptions in Arachne and Perseus was collected. First, to extract useful information from the information systems, it had to be assessed how well the provided information fit the machine

learning models. The survey of common data quality problems was used to craft simple, regular expressions for extracting relevant information. After an optional step of normalization, these extracted granular data elements were then compared to each other. An external schema according to the vocabulary of the CIDOC CRM was specified for entity resolution. This could be helpful for future developments if high quality information becomes available and is structured according to this vocabulary.

The different approaches for extracting information from unstructured or semi-structured sources have been introduced and discussed above. They range from simple, rule-based extraction methods that are supported by lists of terms to complex methods that make use of statistical learning. A compromise had to be found for ALAP: between complexity and effectiveness that has been decided in favor of simplicity. In practice, the means for debugging have been established to craft and optimize these rules in an iterative manner.

For example, logs that show raw data together with the extracted values in a synoptical way have been produced for random inspection. In addition, automatic tests have been introduced that enforce a certain syntax of the extracted information. However, it is difficult to identify cases where the syntax of the extraction result is correct, yet the wrong information elements have been extracted (false positives). Therefore, the extracted information will probably still contain erroneous information that negatively affects model generation.

In certain cases an additional transformation step may be necessary to make information elements comparable. Essentially, these elements are responsible for establishing a syntactical order of extracted elements and for ensuring semantic comparability that is optimal for the application of similarity metrics. The steps of quality analysis, information extraction and transformation of information cannot be clearly separated. Transformations must either be implemented as part of the extraction rules or as an additional step after extraction. The latter approach is necessary in cases where the rule language does not provide adequate means for transformation. Application of multi-lingual vocabularies would be helpful at this stage and could be introduced in the course of follow-up projects.

An exploratory analysis of Arachne and Perseus has been used to identify aspects of entity descriptions that are helpful for entity resolution. After having extracted granular and comparable information from these entity descriptions, statistical summaries can be used again to gain further insights. This analysis can now focus on the actual content to generate more accurate statistical summaries. These summaries help estimate the possible contribution of each entity description aspect for model generation. Of course, improved data quality analysis and advanced data scrubbing could further enhance the results of this analysis.

On the basis of the information that has been acquired so far, each feature will

be discussed with regard to its contribution to entity resolution. The focus will be on interpreting the data that has been produced by the exploratory analysis of each entity description aspect. This has also been helpful for understanding the major factors that influence entity resolution performance. For example, fields that have almost no values in common or that only have very similar values do not contribute greatly to entity resolution. Entity resolution aspects should be syntactically and semantically comparable, and the corresponding values should show high entropy and share a certain amount of values.

For the time being, source information was directly extracted from Arachne and Perseus. In the future, it would be helpful to have access to a standardized data model and schema like the CIDOC CRM, for example, serialized as RDF. It has been mentioned that Semantic Web concepts emphasize correct syntax and clearly defined semantics. Thus, it is necessary to first have large amounts of data available that conforms to high data quality standards. A certain level of data quality has already been established by the data quality and extraction components of ALAP. Thus, reflecting on possible external schemas seems to be useful for making this information accessible for other players in the future.

Software has been re-used or developed for different parts of the described workflow. The Java packages `align.extract` and `align.evaluate` contain classes that implement functionality for accessing raw data from Arachne and Perseus and evaluating the extraction process. Information can either be passed through as raw data or it can be filtered by an additional information extraction algorithm. The former is particularly interesting for the purpose of debugging. Additionally, functionality has been implemented that serializes the extracted information to files for creating statistical summaries and visualizations. The package `align.transform` contains classes that can transform extracted information to achieve normalized and comparable information. Another relevant package is `align.explore`, which was useful in the early stages of the experiment. It supported the process of building a preliminary training set. The package `align.workbench` assembles functionality that presents the matching decisions to end-users for review.

The methodology used for developing and monitoring data extraction and for the cleaning and normalization process has been discussed in this section. Software was developed that is able to extract relevant and useful information from the sources. The iterative development process was guided by manual inspection, looking at summary statistics and automatic syntactical examination. Although the application of more complex methods could enhance the extraction performance, a compromise needs to be found between complexity and practicability.

It is difficult to clearly separate the functional elements of the information extraction and normalization process. In particular, the boundaries between extraction and normalization are fuzzy. Exploratory analysis provides input for

further enhancements for the extraction process and benefits from improved data quality at the same time. Although the extraction performance can be further enhanced, the discussion in the previous section indicates that a reasonable amount of information is available for experimenting with different machine learning models. The introduced procedure was applied separately to all aspects of each entity description.

9.3 Discussion of Considered Entity Description Aspects

The previous sections in this chapter have described and discussed a number of design decisions that have guided the development of ALAP. In contrast, the following sections focus on presenting, interpreting and discussing the achieved results. The exploratory analysis and information extraction of several entity description aspects are evaluated. The information elements that will be considered by no means represent all relevant entity resolution aspects and should be rounded off in the future. Since (exploratory) data mining is an iterative process, additional relevant information could probably be revealed if additional time and effort were invested in doing so. Only systematic data quality problems are mentioned in the following sections. They can be eliminated by crafting more sophisticated extraction rules. A couple of other problems, such as corrupt XML data, can only be manually resolved because they cannot be treated systematically.

The entity description aspects used in ALAP were chosen intuitively. The focus of this experiment was the process of, first, making intuitive decisions and, second, elaborating on these decisions by compiling statistical summaries. Further exploration of the database schemas could certainly reveal additional information that would be useful for entity resolution. For example, information about the creation of a described entity could be helpful if it is represented as a cardinal value. However, a considerable amount of information is represented in a way that cannot be easily treated by certain similarity metrics. Information about the material of a physical object could also be interesting if a controlled vocabulary becomes available.

9.3.1 Bibliographic Information

A considerable amount of information that is managed by cultural heritage information systems is not born-digital but digitalized by evaluating traditional literature and sources. The provenance of this information is recorded by documenting the bibliographic resource that describes an entity. Usually, references to further complementary sources of information are also recorded in the considered literature. Bibliographic references have the advantage that the combination of author names, titles and date of publication can easily be evaluated by similarity metrics

across languages. This is due to the fact that a considerable amount of publications are referred to in the original language of publication. The idea to use bibliographic references to match archaeological entities is related to the Context Attraction Principle, which has been described in section 8.3.2. If two artifacts are described by the same authors, they are more likely to be the same entity than if they do not have any publication information in common.

A cursory analysis of the way that bibliographic references have been represented in Arachne and Perseus revealed that cleaning and normalization is necessary. Arachne uses citations according to the guidelines of the German Archaeological Institute, which give a detailed description of the bibliographic reference. In almost all cases, Perseus uses the name of the author together with the year of publication to refer to a monograph. Multiple references have been bundled in a database field by introducing an internal structure with the help of XML and TEI. For Arachne, the authors and the year of publication need to be extracted from the whole DAI citation. For Perseus, the bundled citations need to be separated for adequate comparison. The lowest common denominator seems to be the combination of the author name together with the year of publication.

The information extraction component tries to identify the names of the authors together with the year of publication for bibliographic references. These are then concatenated as a simple list for later application of similarity metrics. If everything works fine, the information extraction will be carried out as follows. For Arachne, “M. Galinier, La colonne Trajane et les forums impériaux (2007)” is a citation that, according to the guidelines of the German Archaeological Institute, should be extracted as “Galinier 2001”. An example from Perseus is “<P><bibl>Boardman 1985a, pp. 104-105</bibl>; <bibl>Stewart 1990, pp. 154-155</bibl>; <bibl>Ridgway 1981, pp. 16-26</bibl></P>”. It should be extracted as “Boardman 1985 Stewart 1990 Ridgway 1981”.

After having extracted the pieces of information that are shared by Arachne and Perseus, the resulting concatenated strings can be compared by using a soft TF/IDF similarity metric. The similarity method is advantageous because it ensures that even if there are spelling deviations in author names, they do not harm the comparison too much. If there is a larger number of authors and years mentioned in both entity descriptions, the similarity will be larger. The problem with the similarity metric, however, is that variations in the publication year are treated the same as spelling variations in author names, and this can decrease the matching quality. Additionally, the order of the concatenated elements does not matter for determining the similarity, so the association of author and date will not be preserved.

Since bibliographic references are represented as nominal values, simple frequency counts were used for the analysis shown in figure 25. It seems that both

information systems are not referring to the same bibliographic references very frequently. Even if “Richter” is referring to the same person in Arachne and Perseus, the date of publication is still different. This information alone could indicate that the overlap is not very high. But for the contribution of bibliographic references for entity resolution, this is not necessarily bad news. If the frequency of tokens is low, this is more likely to be a good indicator of entity descriptions that refer to the same entity in the world.

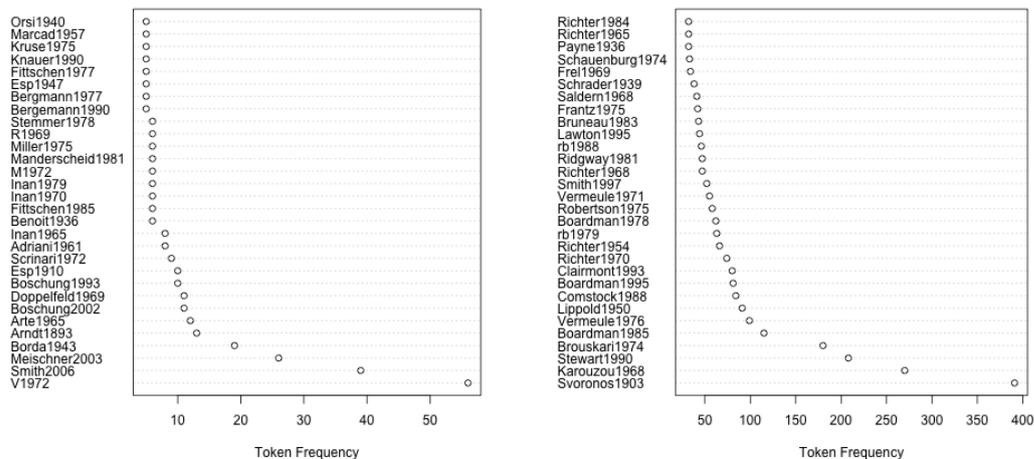


Figure 25: Frequency plots of locations that are attributed to objects in Arachne and Perseus.

The process of data cleaning and information extraction results in structured information that could be published. At the same time, an entity resolution system could consume this structure from third-party information sources. Listing 2 shows the involved classes of the CIDOC CRM as well as the relations between them that are considered for the comparison of bibliographic references. Since Perseus does not record the full title of a bibliographic reference, only author names and publication years have been considered for the time being. However, the bibliographic information of Perseus could possibly be resolved into a full citation with the help of a bibliographic information system.

Listing 2: The extracted information expressed about bibliographic references serialized as Turtle.

```

1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2 @prefix ecrm: <http://erlangen-crm.org/101001/> .
3
4 <http://arachne.uni-koeln.de/entity/123>
5   a ecrm:E22_Man-Made_Object ;
6   ecrm:P67i_is_referred_to_by <http://arachne.uni-koeln.de/entity/234> .

```

```

7
8 <http://arachne.uni-koeln.de/entity/234>
9   a ecrm:E31_Document ;
10  ecrm:P94i_was_created_by
11    <http://arachne.uni-koeln.de/entity/234/creation> .
12 <http://arachne.uni-koeln.de/entity/234/creation>
13   a ecrm:E65_Creation ;
14   ecrm:P14_carried_out_by <http://arachne.uni-koeln.de/entity/345> ;
15   ecrm:P4_has_time-span <http://arachne.uni-koeln.de/entity/456> .
16
17 <http://arachne.uni-koeln.de/entity/345>
18   a ecrm:E39_Actor ;
19   ecrm:P131_is_identified_by
20     <http://arachne.uni-koeln.de/entity/345/appellation> .
21 <http://arachne.uni-koeln.de/entity/345/appellation>
22   a ecrm:E82_Actor_Appellation ;
23   rdf:value "Ridgway"^^<http://www.w3.org/2001/XMLSchema#string> .
24
25 <http://arachne.uni-koeln.de/entity/456>
26   a ecrm:E52_Time-Span ;
27   rdf:value "1981"^^<http://www.w3.org/2001/XMLSchema#gYear> .

```

Information about bibliographic references was considered for ALAP because it is frequently recorded in archaeological information. Additionally, this information tends to be language independent because the combination of author and year of publication does not need to be translated. However, according to the comparison of token frequencies, the overlap between Arachne and Perseus was not very high. On the other hand, this can be good news for actual entity resolution performance because tokens with low frequencies contribute a higher quality of information.

To put it in a nutshell, bibliographic information has favorable features for the performance of entity resolution. At the same time, usable information can be extracted with reasonable effort by applying regular expressions to the DAI guidelines and the strings that have been obtained from XML.⁴² The first analysis revealed that the amount of overlapping information is low. A number of ways to enhance the analysis were mentioned, and could be considered for future development iterations.

9.3.2 Collection and Depository Information

The types of archaeological entities (e.g., sculpture, ceramic) considered for ALAP are typically curated in museum contexts. This means that they are part of a museum collection, and their depository, whether current or former, is known and recorded. Thus, the Context Attraction Principle is also relevant for collection information. Several entity descriptions indicate that the referenced entities are

⁴²Bibliographic references in Arachne follow the citation guidelines of the German Archaeological Institute (i.e., DAI guidelines). For Perseus, bibliographic references are structured as XML.

being, or have been, curated by the same museum. This can be an additional hint that the same entity is being described. In analogy to bibliographic information, names of museums are often not translated and tend to be language independent.

A cursory data quality analysis revealed that a slight reorganization of collection information for Arachne is necessary. The database schema of Arachne provides multiple elements that organize information about the depository of an entity. Perseus represents information about the collection of an object as one single string. Here, the representation of collection information in Perseus does not follow any regular syntax. For instance, sometimes the name of the city where a museum is located has been represented as a prefix and sometimes as a suffix.

Thus, the fields of Arachne have been concatenated for the time being: all the information in Arachne and Perseus is represented as one string. The three information elements, “Athen“, “Panagia Gorgoepikoos“ and “Kleine Metropolis“, have been concatenated as “Athen, Panagia Gorgoepikoos, Kleine Metropolis” for Arachne. For Perseus, the strings “Museum of Fine Arts, Boston” or “Berlin, Antikenmuseen” have been left untouched. For downstream steps of processing, it may be useful to use a gazetteer to extract the city name for Perseus.

Due to the use of different national languages and syntactical variations, the application of soft TF/IDF seemed to be suitable to deal with spelling variations. Future versions of the alignment framework could invest further effort in information extraction and resolve the location information to a canonical identifier. Additionally, a gazetteer could provide different names for a place, which would enhance the application of similarity metrics.

Information about the collection – how a material entity is curated – has been represented as nominal values. Thus, a frequency analysis seemed to be the most suitable statistical summary, which is illustrated in figure 26. Although the frequency diagrams do not take into account the values that could not be extracted, some observations seem to be significant. Apparently, Arachne has introduced the virtual place name “verschollen”, which marks lost material entities. Overall, the focus seems to be on archaeological objects that are curated by museums in Rome.

On the other hand, Perseus seems to have a strong focus on material objects that are curated by the National Museum in Athens. Although the amount of entities is much higher for Arachne, there seems to be a significant overlap between the larger collections of curated objects. These collections include, for example, “Athens, National Museum”, “Rome, Musei Capitolini”, “Rome, Museo Nazionale delle Terme”, “London, British Museum”, “Paris, Musée du Louvre” and “Copenhagen, Ny Carlsberg Glyptothek”. As mentioned, it would be useful to have a synopsis of the *shared* values that appear most frequently. Of course, this is also true for collection information.

Listing 3 shows the data model elements that organize the extracted place

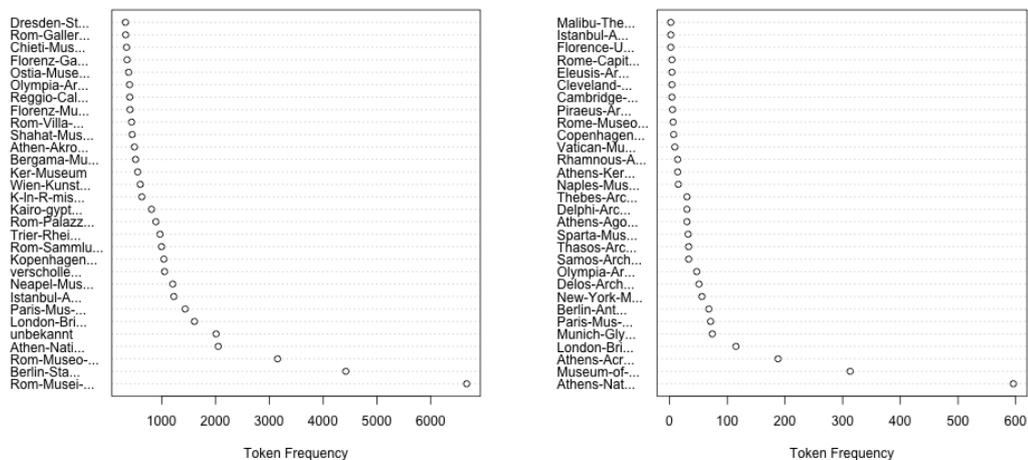


Figure 26: Frequency plots of collection and depository locations that are attributed to objects in Arachne and Perseus.

information. Perseus provides place information with two hierarchy levels (i.e., museum, city), and Arachne provides an additional level (i.e., country). Since the hierarchical information has not yet been extracted for Perseus or concatenated for Arachne, there are still two comparable strings. Additionally, Arachne provides multiple identifiers for certain place names that have not yet been modeled.

Listing 3: The extracted information expressed about collection and depository locations serialized as Turtle.

```

1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2 @prefix ecrm: <http://erlangen-crm.org/101001/> .
3
4 <http://arachne.uni-koeln.de/entity/123>
5   a ecrm:E22_Man-Made_Object ;
6   ecrm:P53_has_former_or_current_location
7     <http://arachne.uni-koeln.de/entity/234> .
8
9 <http://arachne.uni-koeln.de/entity/234>
10  a ecrm:E53_Place ;
11  ecrm:P87_is_identified_by
12    <http://arachne.uni-koeln.de/entity/234/appellation/1> .
13
14 <http://arachne.uni-koeln.de/entity/234/appellation/1>
15   a ecrm:E44_Place_Appellation ;
16   ref:value "Athen, Panagia Gorgoepikoos, Kleine
17     Metropolis"^^<http://www.w3.org/2001/XMLSchema#string> .

```

Geographical information is highly relevant for information integration in cultural heritage contexts. Information about the collection of an object implicitly covers geographical information. It seems that Arachne and Perseus describe a

considerable amount of material objects that are curated in the same collections. Since the names of these collections are not translated into the national language of an information system and place names have orthographical similarities, a soft TF/IDF metric should be experimented with. Information extraction and normalization were only implemented to the degree that it makes information elements from both systems comparable.

The statistical summary reveals overlapping content, by exclusively showing the values that have a high frequency. However, while collection information is helpful for blocking, it probably does not contribute much to distinguishing objects from a certain collection. Thus, the matcher must rely on other features for a larger group of entities that are curated within the same collection. But collection information in combination with an accession number can form an identifier that is unique in the majority of cases; it can be used to bootstrap ALAP. Future iterations of experiment development should consider how to make information extraction more granular. This would include exploiting the hierarchical relations of places.

9.3.3 Findspot Information

The collection that a material entity is curated in usually has a distinct place and, therefore, geographic information attached to it. This information turns out to be valuable for partitioning entity descriptions into buckets that hold descriptions that co-refer to the same entity with a higher probability. But it has also been mentioned that this does not contribute to distinguishing a considerable amount of entities that belong to the same collection. Many entity descriptions also cover information about the find place. This information can be useful because it has the same favorable features for entity resolution as the collection information.

In Arachne, the database fields “Fundkontext”, “FunddatumObjekt” and “HerkFundKommentar” seem to record information about the findspot of an object in addition to other “findcircumstances”. For historical reasons, multiple redundant schema elements can be found in Arachne for recording information about findspots. However, for the time being, only the original fields are considered because they contribute more information. The extraction component could include about 7900 more values if it were to be enhanced.

In many cases the database field “Fundkontext” contains the name of a city together with the name of a country (“Aquileia, Italien”). There are some field values that also mention a find date (“Athen, Agora (1931) Griechenland”). In some other cases there is also a short description attached (“Aquileia, Grabungen im Bereich des Cassis-Eigentums., Italien”). The database field “FunddatumObjekt” often has rather exact values (August 1926), but sometimes has values that are more vague (“In der ersten Periode der Ausgrabungen”). The field “HerkFundKommentar” contains a more detailed description of the circumstances of

the discovery (“gefunden “in der Nähe der Gräber 12 und 18” (Inventar)”).

Perseus records information about the findspot in a single database field. Often, the name of a city is separated from the name of a site with a comma (“Athens, Acropolis”). In a number of cases, information about the find date is also mentioned as a year (“Minorca (before 1833)”). Many place names are accompanied by further textual descriptions (“Delphi (in 1894 and 1895 around the monuments of the Sacred Way)”).

For the time being, no complex information extraction has been performed for findspots. Since Perseus puts information in one database field, which is distributed to multiple fields in Arachne, this has been concatenated for later comparison. Subsequently, a soft TF/IDF similarity metric was applied to the prepared information. By doing this the spelling variations of the mentioned place names were able to be captured. However, the mentioned find dates should be treated in a different way in the future because an exact match of years would probably lead to better results.

Figure 27 shows the most frequent findspots for Arachne and Perseus. The collection of Perseus seems to have a strong focus on Greek material, especially on the material objects that have been discovered in Athens. Greek material also has the highest frequency of findspots in Arachne, but Roman material seems to be represented slightly better, if compared to Perseus. Additionally, there seems to be a large amount of objects in Arachne where the findspot is not recorded, most likely because it is “unknown”.

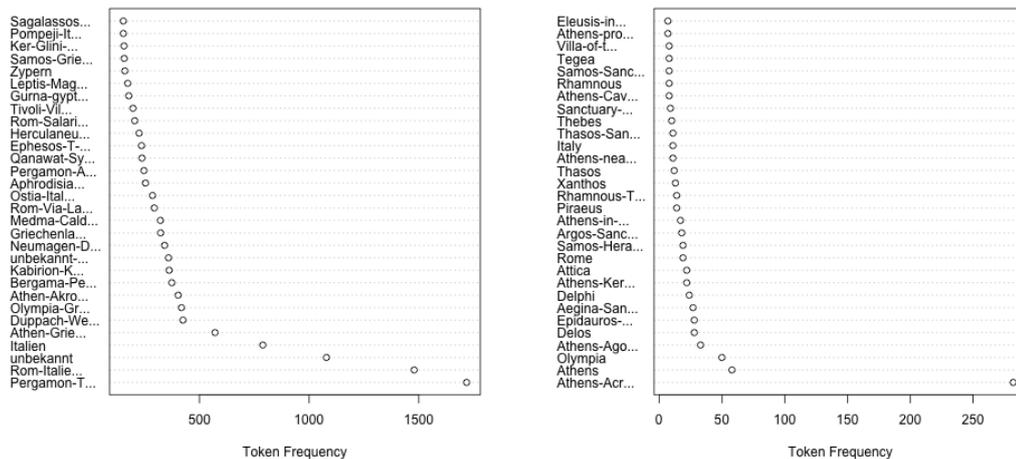


Figure 27: Frequency plot of findspots that are attributed to objects in Arachne and Perseus.

Listing 4 shows the external schema for information about the findspot of a

material object. However, with the current data cleaning and information extraction setup, it is not possible to adequately represent the needed information in a granular manner. Therefore, the event of finding a material thing has been modeled and annotated with a note. The event itself has been further differentiated as an excavation event by using the CRM typing mechanism.

Listing 4: The extracted information expressed about findspots serialized as Turtle.

```

1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2 @prefix ecrm: <http://erlangen-crm.org/101001/> .
3
4 <http://arachne.uni-koeln.de/entity/123>
5   a ecrm:E22_Man-Made_Object ;
6   ecrm:P12i_was_present_at <http://arachne.uni-koeln.de/entity/234> .
7
8 <http://arachne.uni-koeln.de/entity/234>
9   a ecrm:E5_Event ;
10  ecrm:P3_has_note "Athen, Agora (1931)
11    Griechenland"^^<http://www.w3.org/2001/XMLSchema#string> ;
12  ecrm:P2_has_type <http://arachne.uni-koeln.de/entity/345> .
13
14 <http://arachne.uni-koeln.de/entity/345>
15   a ecrm:E55_Type ;
16   rdf:value "excavation"^^<http://www.w3.org/2001/XMLSchema#string> .

```

The information about the findspot of an archaeological object is a valuable complement to other geographic information. However, data quality in the sense of usability for ALAP was rather limited in many cases. A soft TF/IDF metric was applied for comparison, which attenuated some but not all of these shortcomings. The exploratory analysis revealed that Arachne and Perseus have a slightly different focus; however, a reasonable amount of overlapping entity descriptions can still be expected.

If additional effort went into the information extraction component for findspots, the quality of the results could be significantly enhanced. For the time being, not all relevant and identified schema elements have been exploited to adequately organize the represented information. But the approach that has been chosen to consider information about find circumstances is helpful for aligning a significant number of entity descriptions. And the data quality situation for geographic information will certainly improve in the near future because larger projects are working on establishing a gazetteer infrastructure for the humanities.

9.3.4 Accession Numbers

Accession numbers are very helpful for the alignment of cultural heritage information because they often have unique identification characteristics. In combination with information about a specific curating museum and / or collection, the accession number can be used as the distinguishing feature. Thus, accession numbers are especially relevant for compiling a set of training pairs to bootstrap the ma-

chine learning components. This section describes how accession numbers are used to drive the information alignment experiment.

The data quality for accession numbers is usually rather high. For example, in Arachne the database record with the id “226982” (“Pferd eines Dreigespann”), which was curated in the “Antikenmuseum der Universität” in “Leipzig”, has the inventory number “TK 115”. Additionally, an old inventory number “T 298 c” has also been recorded. An example for Perseus is the record with the id 1870 (“Herakles and the centaurs”), which was curated in the “Museum of Fine Arts” in Boston, and the accession number “84.67a-b”.

However, in some cases the way that accession numbers are represented slightly differs for Arachne and Perseus, making a strict comparison impossible. And there are some cases where the collection name and the accession number match, yet the entity description does not seem to refer to the same entity. For example, the Perseus record with the id 3864 (“The Atarbos Base”) shares the inventory number 1338 with the Arachne record 135088 (“TerrakottarelieF”), and both belong to the collection of the Acropolis Museum in Athens.

Both Arachne and Perseus have dedicated schema elements that organize information about accession numbers. Therefore, the values have been taken as is – no additional information extraction was performed on them. Since the syntax of representation tends to vary, a soft TF/IDF similarity metric with a very high threshold was chosen. This allows entries with varying punctuation or token order to be matched. Preparing a frequency analysis of the accession numbers does not seem to be useful because they function as identifiers in most cases.

Listing 5 shows the schema elements according to the vocabulary of the CIDOC CRM that adequately represent the accession numbers. It has been mentioned that accession numbers in combination with the collection name can uniquely identify material objects in most cases. In many cases, additional identifiers have been assigned to material objects that are applicable in more general contexts. These identifiers include, for example, catalogue numbers and URIs that are used in representations for the Semantic Web.

Listing 5: The extracted information expressed about accession numbers serialized as Turtle.

```
1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2 @prefix ecrm: <http://erlangen-crm.org/101001/> .
3
4 <http://arachne.uni-koeln.de/entity/123>
5     a ecrm:E22_Man-Made_Object ;
6     ecrm:P1_is_identified_by <http://arachne.uni-koeln.de/entity/234> .
7
8 <http://arachne.uni-koeln.de/entity/234>
9     a ecrm:E41_Appellation ;
10    rdf:value "TK 115"^^<http://www.w3.org/2001/XMLSchema#string> .
```

Because accession numbers uniquely identify cultural heritage objects, they are

useful for bootstrapping the alignment process. In combination with geographic information and / or a collection name, they can unambiguously identify many archaeological entities. However, there are cases where accession numbers do not help identify the entity descriptions that co-refer to an entity in the world, which leads to wrong decisions if the training set exclusively contains co-referring pairs with matching accession numbers.

Accession numbers are extractable and comparable with almost no effort. And if the shortcomings, which have been discussed above, are considered, the comparison can provide valuable information for the alignment process. Accession numbers in combination with collection information were successfully used in the course of exploratory analysis. They are important for compiling a training set, semi-automatically. Using this set with the machine learning component helps to assess the role and contribution of the additional entity description aspects.

9.3.5 Object Dimensions

The dimensions of cultural heritage objects are usually part of the documentation in cultural heritage information systems. Object dimensions are partially recorded as cardinal values, which can be compared by using adequate mathematical operations. Not only can an exact distance between measurements be determined but also an exact ratio, which makes comparison more tolerant for very high values. This section describes how the object dimensions in Arachne and Perseus can be represented and exploited.

Arachne provides dedicated schema elements that organize information about the height, breadth and depth of a material object. For example, the database record with the identifier 85 (“Kopf einer Statue der Kybele”) has a height of 35 cm, a breadth of 21 cm and a depth of 19 cm. 53,211 database records provide information about the height, 31,896 about the breadth, 18,881 about the depth and 3,311 about the diameter of a material object. The records give the overall dimensions, and there is an additional free text database field for recording additional measurements.

Perseus only provides one database field that has a certain internal structure for recording information about the dimension of an object. For example, the database record with the identifier 2021 (“Goddess (Persephone?) seated in elaborate throne”) has sophisticated dimension information attached: “H 1.51m, H of face 0.18 m, H of head and neck 0.33 m, H of base 0.06 m, Base 0.90 X 0.70 m, H of throne 1.245, W of throne 0.69 m”. This field seems to record both the overall dimensions of an object and additional measurements. In this case it is necessary to extract information that is comparable to the dimension information of Arachne.

No sophisticated information extraction has been established for Arachne be-

cause the dimension information is stored in a database field that is dedicated to it. Perseus documents multiple measurements for a certain dimension. Thus, the extraction mechanism strives to obtain the highest measurement because this is probably the overall height of the object. Since Perseus uses a different unit to record the measurement, an additional normalization step has been introduced. For example, the value “151 cm” would be extracted from the field value, which was mentioned in the preceding.

Since the object dimensions are represented as cardinal values in Arachne and Perseus, a ratio was determined for the height of the objects. This is beneficial for addressing all sizes of objects in the comparison, because small objects will only have small deviations in the object’s size, while larger objects will have a higher deviation. However, Perseus uses additional means to describe the size of an object, like “Miniature” or “Colossal”. For a comparison, it would be useful to transform these additional descriptors into a numeric value before proceeding.

The width, height and diameter of an object can also be a good feature to compare in the future. Figure 28 shows the histograms of measurements in Arachne and Perseus for object heights of up to 3000 mm.

Arachne describes one object as having a height of 0, described by the record 47997: “Grabrelief eines Mannes mit Knaben” (<http://arachne.uni-koeln.de/item/objekt/47997>). This is obviously unintentional; someone probably used 0 to express the fact that the height was unavailable. The first quartile is at 250 mm, the mode at around 273 and the median at 420 mm. The expected value is around 660 mm, with a standard deviation of 556 mm. The third quartile (75%) is at 870 mm, so that 50% of all described objects are between 250 mm and 870 mm in height. There are also a number of mild and extreme outliers. One of the outliers is the object described by the Arachne record 151581, “Obelisk, sog. Solarium Augusti” (<http://arachne.uni-koeln.de/item/objekt/151581>), which is more than 21 meters in height.

The minimum value is 14 mm, the first quartile is at 270 mm, the median is at 500 mm and the expected value is at 2426 mm. Because the third quartile is at 1000 mm, 50% of all described objects are between 270 and 1000 mm in height. The maximum value is 2549000 mm, which was obtained from a database record with the ID 3846, describing two fragments. This is probably an encoding or extraction error, as the number is far too large to reflect the height of one of the fragments.

The histograms of Arachne and Perseus indicate a similar frequency distribution. It seems that both information systems describe entities of approximately the same height. This could be interpreted as an indicator of a high overlap of content. At the same time, this could also be an indication that the height probably does not contribute much information to the actual entity resolution. The statistics also

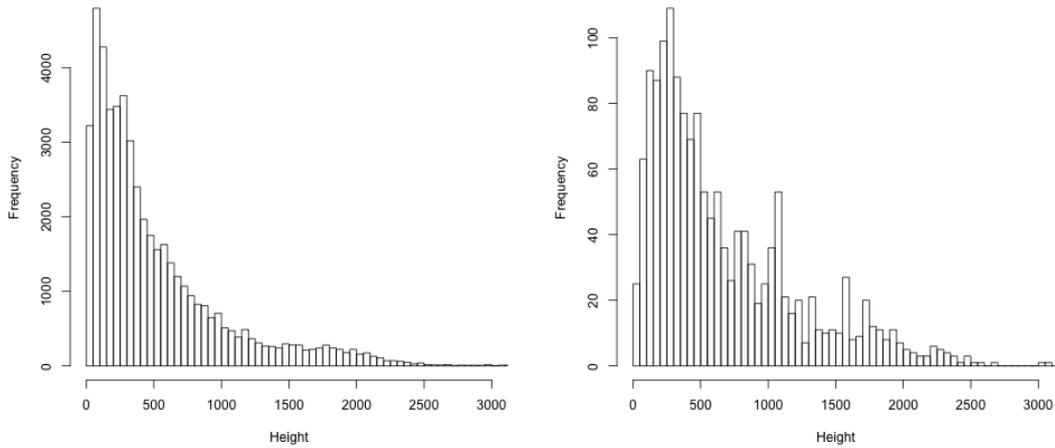


Figure 28: Histograms for the distribution of measured heights up to 3000 millimeters for material objects described by Arachne and Perseus.

reveal some outliers like a height of 0 or objects larger than 2 meters, which could be falsely categorized or data glitches. The 5th percentile is at 44 mm in Arachne and approximately 94 mm in Perseus. The 95 percentile is at 1930 in Arachne and at 1720 mm in Perseus. This may indicate a rather small number of significant outliers that should be looked at. The alignment process should take these facts into consideration by ignoring the extreme outliers and giving the values with high density lower weights.

Arachne provides granular values for the overall height of a material object, and it was possible to extract a considerable amount of comparable values from Perseus. Listing 6 shows an example entity description represented for publishing. The current model only expresses exact measurements in centimeters. It could be extended in the future to represent qualitative measurements like “miniature” or “colossal”. Since the extraction mechanism cannot guarantee that the semantics of values have been interpreted correctly, published data could be erroneous. Therefore, the extraction mechanism that is used should be documented so that the external users of the data can estimate its reliability.

Listing 6: The extracted information about the height of a material object serialized as Turtle.

```

1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2 @prefix ecrm: <http://erlangen-crm.org/101001/> .
3
4 <http://arachne.uni-koeln.de/object/1>
5     a ecrm:E22_Man-Made_Object ;
6     ecrm:P43_has_dimension

```

```

7         <http://arachne.uni-koeln.de/object/1/dimension/height> .
8 <http://arachne.uni-koeln.de/object/1/dimension>
9     a ecrm:E54_Dimension ;
10    ecrm:P2_has_type <http://arachne.uni-koeln.de/types/dimension/height> ;
11    ecrm:P91_has_unit <http://arachne.uni-koeln.de/units/cm> ;
12    ecrm:P90_has_value "105"^^<http://www.w3.org/2001/XMLSchema#integer> .
13
14 <http://arachne.uni-koeln.de/units/cm>
15     a ecrm:E58_Measurement_Unit .
16
17 <http://arachne.uni-koeln.de/types/dimension/height>
18     a ecrm:E55_Type .

```

The extraction and normalization of information that records the dimensions of material objects can be performed with reasonable effort. However, since the quality of the extraction component depends on the accuracy of the used extraction rules, the process is potentially unreliable. This should be reflected by the data model that is used to share the information with other parties. Statistical summaries can be used to assess the data quality, which is sufficiently high. Additional data cleaning could enhance the process but would involve additional manual review by experts, which is beyond the scope of ALAP. However, establishing better data quality management both for Arachne and Perseus bears considerable potential for enhancing future entity resolution efforts.

Since the dimensions of an object are represented as cardinal values, more sophisticated statistical summaries were able to be compiled for the experiment. The shapes of the frequency distribution plots were compared and the outliers were analyzed to identify data glitches. For the time being, only the heights of material objects were considered, which could be extracted and normalized with reasonable effort. However, future development of ALAP should also consider additional measurements like the depth, diameter and so on. The actual contribution of dimension information for alignment work will be explored in more depth later in an analysis of the models that were generated by machine learning.

9.3.6 Short Entity Description

Another interesting entity description aspect is the brief full-text description of an archaeological entity. Because the short description is expressed in the national language of each information system, it is difficult to compare. However, the statistical summary reveals that tokens like “parthenon”, “athena” and “asklepio” are frequently used, and these can be treated with similarity metrics. This section introduces and discusses the method that has been used to exploit this information for entity resolution.

It is difficult to perform a data quality analysis on a field that – per definition – represents information as full-text. However, certain features in this type of description can be identified that may improve or inhibit the extraction and com-

parison of useful information. For example, a database record in Arachne with the ID 5383 describes an entity with the short description “Kopf der myronischen Athena”. And in Perseus, the database record with the ID 1915 describes an archaeological entity with the short description “Athena and the other gods fighting the giants”. A title has also been provided, “Gigantomachy pediment”. Obviously, these descriptions contain information that needs to be translated for proper comparison, like “Kopf”, “fighting” and “pediment”.

It would be desirable to extract only those features that are language independent or to introduce at least a simple form of translation. The names mentioned in these short descriptions could also be matched with controlled vocabularies that provide multiple appellations of entities. But for the time being, only stop words have been dropped and the rest of the description has been incorporated without further processing. The intuition is that if descriptions share names of equal entities, the results of similarity metric application will be significantly higher despite the noise.

To address the abovementioned problems, a soft TF/IDF metric has been applied to the short, full-text descriptions of entities. The order of the tokens does not matter, and the tokens that vary in spelling because of different national languages will be considered. Of course, this approach will fail in cases where the deviation in spelling is rather high, like for “Cologne” and “Köln”. However, similarity metrics that are backed by controlled vocabularies could be used for future alignment framework implementations to approach these problems.

Figure 29 gives an overview of the tokens that are most frequently used to describe archaeological objects. The frequency of certain tokens provides the first overview of the core area that is described by the records of a database. The figure also gives an overview of the tokens that are most frequently used to describe archaeological objects in Perseus. Both Arachne and Perseus seem to have a strong focus on figures and statues with heads that are male or female. This is another indication for a strong overlap of content. In Perseus, tokens like ‘perhaps’ and ‘two’ often occur in the descriptions, indicating a certain amount of uncertainty, but also that database records can describe groups of objects.

Among the 30 most frequently used tokens for describing archaeological objects in these information systems, at least 11 overlap (athena - athena, knab - youth, herm - herm, mannlich - male, weiblich - female, torso - torso, fragm - fragment, frau - woman, mann - man, kopf - head, statu - statu). Although not all of these overlaps can be adequately matched by applying similarity metrics, they suggest a high overlap beyond the obvious focus of both information systems. Similar to the previously mentioned entity description aspects, it would be interesting to create a summary for those tokens that have a high similarity for Arachne and Perseus.

Listing 7 shows how short descriptions have been treated in ALAP. It is difficult

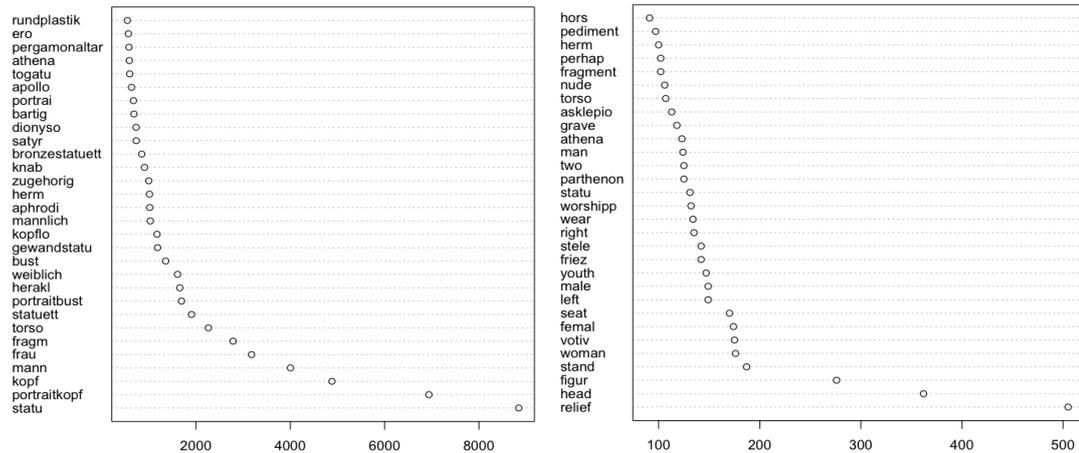


Figure 29: The frequency of tokens in the short descriptions of the archaeological objects in Perseus.

to create a straightforward set of schema elements for the short entity description because it is represented as full-text. Therefore, the schema for representing information that has been extracted from the short description could become arbitrarily complex. Any number or type of entity could be expressed with natural language, and the complex semantic relations between them could also be expressed. However, both entity types and the relations that are frequently mentioned in short descriptions could be modeled and extracted by more complex means of information extraction in the future. At the present, the short description is modeled as a container of information that has not yet been explicitly structured.

Listing 7: The extracted information from the short description of a material object serialized as Turtle.

```

1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2 @prefix ecrm: <http://erlangen-crm.org/101001/> .
3
4 <http://arachne.uni-koeln.de/object/1>
5   a ecrm:E22_Man-Made_Object;
6   ecrm:P3_has_note "Kopf der myronischen
   Athena"^^<http://www.w3.org/2001/XMLSchema#string>;

```

Because of the involved complexity, information extraction was not performed on the mentioned short summaries, which are represented as full-text. For the statistical analysis, the descriptions were decomposed as tokens, and then treated by a stemmer. Because the short summaries, which are part of the entity descriptions, are represented in multiple national languages, there is obviously a difficulty with the determination of similarities. However, a considerable number of names like “Athena”, “Rome” and “Herm” are used in these summaries. In these cases, the application of similarity metrics was useful.

The summary indicates that Arachne and Perseus describe comparable entities, and that a reasonable overlap between them can be expected. However, it is difficult to assess the actual contribution of short entity descriptions without the application of information extraction. An evaluation of the generated model could reveal additional information about the relevance of the summary for entity resolution. Additional means of information extraction should be installed in the course of future developments, so that the comparisons can be more specific and less disturbed by noise.

9.4 Entity Resolution and Management

Different software components are developed to establish a framework for ALAP. These components can partially rely on proprietary software development and additional external software libraries, if appropriate. In particular, third-party software is used to build the infrastructural elements necessary to support indexing and partitioning. The following sections describe the functional components of the framework and how they interact in more detail. To this end, the output of different components are discussed and the success of each component is assessed.

The major components of this framework implement functionality to partition datasets, to compare entity descriptions, to decide on matches and non-matches and to visually represent the results of the entity resolution efforts. Throughout ALAP, these functional components have been developed to a certain level of maturity, which allows for the generation of meaningful results. However, there is considerable potential for enhancements, for instance, by putting additional effort into software development. Components that allow for more parameterization of single components and for better control over the workflow have yet to be developed.

9.4.1 Partitioning of Datasets

The number of comparisons that must be performed for entity resolution can be overwhelming if real-word data sets are used. For (interactive) debugging and parameterization of the workflow, the framework must be able to produce meaningful results in a reasonable amount of time. Before an effective and efficient alignment framework can be developed, the information must be partitioned. By establishing preliminary means, the scalability of the framework can be dramatically improved in order to handle huge datasets. The partitioning of datasets is implemented by relying on a combination of third-party software components, which have been iteratively parameterized. The following section introduces these components and describes their mode of interaction.

The different approaches that allow for effective and efficient partitioning of datasets, such as indexing and windowing, are considered. As mentioned, canopy clustering seems to be a rather promising method, as it has successfully been used in large-scale entity resolution contexts. Additionally, an implementation of canopy clustering exists as part of the Apache Mahout machine learning library, which is partially implemented on top of Apache Hadoop. Apache Mahout provides tools to convert a Lucene index into a Hadoop sequence file, which makes it convenient for further processing in the context of these frameworks. Another platform that has been used is Apache Solr, a wrapper that extends the functionality of Apache Lucene.

A running Solr server was used to create a Lucene index of the information that was acquired from Arachne and Perseus. The fields considered for partitioning are the depository of an entity, the inventory number and the short entity description. To achieve better clustering results, the information was pre-processed at index time to perform certain transformations. In particular, the preprocessing consisted of tokenizing the input strings into bigrams and transforming all uppercase characters to lowercase. After the indexing was performed, each entity description was represented by these term vectors, which represent the fields that have been considered. Apache Mahout provides means for mapping a Lucene index onto a Hadoop vector file. The latter was used to create vectors for all selected features for machine learning.

After creating a representation of the Lucene index as a Hadoop vector file, the data was used as input for Mahout. In the present case, a canopy clusterer has been instructed to determine distances of indexed entity descriptions by using the Squared Euclidean Distance Metric. The thresholds that are used to define clusters have been set to 300 and 200 to achieve reasonable cluster sizes. The result of the clustering process was again represented as a Hadoop sequence file, and the actual clusters were extracted with the help of a cluster dumper, which is part of Apache Mahout. Listing 8 shows an excerpt of the dumped clustering information as a CSV file.

Listing 8: An excerpt of the dumped clustering information as CSV file.

```
1 ...
2 ngTitle ,143 ,arachne ,120296
3 ngTitle ,143 ,arachne ,120290
4 ngTitle ,143 ,arachne ,130540
5 ngTitle ,143 ,perseus ,2060
6 ngTitle ,143 ,perseus ,2327
7 ngTitle ,143 ,perseus ,3335
8 ...
```

A CSV file such as this one has been created for each entity description aspect that was considered for partitioning. These files were used to compile a set of pairs

that will be used later for the actual matching. The first step was to combine each entity description from Arachne with each entity description from Perseus. The resulting pairs of each cluster were then merged, causing each pair to exist only once. The same procedure was performed to merge the results of different clustering runs. With the current parameterization of the partitioning workflow, the number of pairs that need to be considered is 3 203 292.

The performance of the partitioning relies on adequately adjusting several parameters. Different transformations can be applied at the time of index creation, such as stemming or the production of n-grams. These transformations determine the way that entity descriptions are represented as vectors. In particular, canopy clustering can be parameterized in multiple ways. For example, different values for the thresholds influence the quality of the clustering. Larger thresholds result in larger clusters with fewer false non-matches. At the same time, the increased cluster size may lead to inefficient subsequent processing. Clustering is rather resource intensive, and runtime can add up to several hours in an alignment experiment, so tuning these parameters with intermediate results can be laborious.

A palpable reduction of entity description pairs that need to be compared can be achieved by using the results of preprocessing and clustering. However, the quality of the clustering process has not been quantified yet. Not only is the overall reduction of comparisons important, but other features like homogeneous cluster sizes and the similarity of entities within the same cluster should be quantified as well to support parameterization. For the time being, parameters have been set and adjusted in a qualitative manner by manually inspecting the generated clusters. In the future, it would be helpful to consider the mentioned statistics for the results of clustering, which cover information about the size and homogeneity of clusters. This would also support the (semi-)automatic adjustment of parameters.

9.4.2 Comparison of Entity Descriptions

A vital part of ALAP is the actual comparison of entity description aspects. These comprise, for example, the dimensions of an object and the mentioned bibliographic references. Exploratory analysis reveals valuable information about the role of these aspects in the process of entity resolution. Canopy clustering dramatically reduces the number of necessary comparisons. Therefore, a deeper inspection of the remaining entity descriptions that have been flagged as potentially co-referring is now possible. This section describes how the entity descriptions can be compared to determine the actual similarity.

To determine the similarity of entity descriptions, single aspects were first compared in isolation and then combined by machine learning. Since the application of similarity metrics to over three million pairs of entity descriptions after partitioning is still resource intensive, each similarity has been determined individually.

The similarity for each aspect has been calculated in a serial manner to achieve shorter iterations and to enable better parameterization. Listing 9 shows an excerpt of a log file, which records the comparison results of the short summaries for inspection and improves fine tuning.

Listing 9: An excerpt of the log file that records comparison results for the short descriptions in Arachne and Perseus.

```

1 ...
2 10.0,'Portraitkopf des Augustus',,'of statue bust Head or Domitian
   from portrait '
3 3.0,'Portraitkopf des Augustus',,'African with Head origin , a origin
   thick , young of curly Libyan, Portrait head hair perhaps man'
4 5.0,'Portraitkopf des Augustus',,'of girl Julio-Claudian neck Head
   Roman a Portrait the period and'
5 0.0,'Portraitkopf des Augustus',,'Girl with girl Corinth Young from
   scalloped coiffure '
6 5.0,'Portraitkopf des Augustus',,'of Julio-Claudian bust a Bust
   Portrait the period clean-shaven man'
7 6.0,'Portraitkopf des Augustus',,'wavy of with Bronze Roman a hair
   Bearded man portrait '
8 0.0,'Portraitkopf des Augustus',,'bald , of Head Republican Late older
   a head man'
9 44.0,'Portraitkopf des Augustus',,'Posthumous of Augustus Bust
   portrait '
10 ...

```

Although almost all aspects of an entity description have been treated with a soft version of TF/IDF, the threshold for the fuzzy matching of tokens has been adjusted. For example, higher thresholds work well for inventory numbers that have only slight syntactic deviations, and lower thresholds seem to work well with short textual descriptions of entities. Tokens have been created by applying a German Analyzer of the Lucene framework, and the distance metric has been applied to the resulting string of tokens. The functionality for soft TF/IDF has been bundled in a class that also supports to save the document corpus to achieve additional efficiency. In addition, the class allows for multilingual tokenization and is also ready to be used in combination with Latent Semantic Analysis. For cardinal values like the dimensions of entities, the ratio of one measurement to the other was determined.

Methods that rely on the idea of dimensionality reduction by principal component analysis (PCA) have also been considered for dealing with the problem of multiple national languages. The method of Latent Semantic Analysis (LSA), which implements the principal component analysis, has already been described in section 6.1.3. Although it is not yet part of the developed framework, LSA was evaluated for entity description comparisons. The sets of instances that were either hand-picked or resulted from early iterations of the alignment framework have

been used to create a rudimentary semantic space. To that end, a term-frequency matrix has been generated by concatenating aspects of entity descriptions that refer to the same entity. Listing 10 shows how the results of feature comparison changes for some example entity descriptions if LSA is used in addition to TF/IDF. The listing is by no means a representative sample, but it shows that it can significantly raise the similarity for matches.

Listing 10: An excerpt of the log file that records results of the application of LSA.

```

1 Comparing
2   'Statuette einer Kore'
3 and
4   'of Statue Peplos Kore maiden'
5 ==> TF-IDF + LSA: 0,98
6 ==> TF-IDF: 0,23
7
8 Comparing match
9   'Kopf eines Mannes'
10 and
11   'with defeated Head tongue protruding a lips. warrior, from pursed
      of his Aegina Warrior between'
12 ==> TF-IDF + LSA: 0,31
13 ==> TF-IDF: 0,00
14
15 Comparing nonmatch
16   'Karyatide von den Inneren Propylaeen in Eleusis'
17 and
18   'of Meleager Torso head Fogg and'
19 ==> TF-IDF + LSA: 0,14
20 ==> TF-IDF: 0,00

```

By decoupling the comparison processes for aspects of an entity description from each other, all similarities can be determined in under an hour, per aspect. However, it was mentioned that the comparison of entity descriptions leaves room for improvement. The only similarity metric that has been applied to nominal values so far is a soft version of TF/IDF. The evaluation of the log files shows that correct and helpful results can be achieved by this approach. For example, by applying more specialized metrics like Soundex to names of authors, the quality of the results could be further enhanced.

Additionally, the application of Latent Semantic Analysis is promising in a multilingual environment. However, the creation of the semantic space is currently limited to a small corpus (the labeled set). This renders it impossible to apply it as a similarity metric to documents that contain tokens that are not part of the corpus. Thus, the application is currently limited to a fraction of entity description pairs, and many missing values still exist. In the future, it would be interesting to test the technique with a larger corpora of labeled data and to exchange the

semantic spaces between these different projects.

To determine the similarities of entity description pairs, a workflow was established to enable interactivity. This was achieved by determining the similarity of each aspect individually and in a serial manner. This method allowed for the interactive tuning of parameters. However, the current setup leaves room for further enhancements, which should be considered for future work. Additionally, the results of Latent Semantic Analysis are promising but still fall short because of the small set of overlapping records that are currently available.

9.4.3 Deciding on Matches and Non-Matches

The previous sections have discussed activities that deal with acquiring relevant information for generating a decision model. This model can be applied to a set of unlabeled entity description pairs to make a decision on whether two entity descriptions refer to the same entity. The number of pairs that need to be considered is considerably reduced by making use of partitioning methods. Distance metrics are applied both to the training instances and the instances that are classified by the decision model. This section describes and discusses generating and applying a decision model for actual entity resolution.

After the similarities between the different entity description aspects have been determined, the results can be joined and transformed into a Weka ARFF file, as shown in listing 11. The Attribute-Relation File Format (ARFF) was developed for the Machine Learning Project by the Department of Computer Science at The University of Waikato. It is an ASCII file that lists the instances that share a set of features; these instances can then be used with the Weka machine learning software. The following paragraph will provide a basic overview of how the file is structured.⁴³

Listing 11: Determined similarities of pairs in the sample represented as an ARFF file.

```
1 @relation pairs
2
3 @attribute arachneId string
4 @attribute perseusId string
5 @attribute class {match,nonmatch}
6 @attribute bibSim numeric
7 @attribute heiSim numeric
8 @attribute sumSim numeric
9 @attribute accNumberMatch {YES,NO}
10 @attribute finSim numeric
11 @attribute locSim numeric
```

⁴³Please refer to the online documentation for a more thorough description. There is more information on ARFF available at the Weka Wiki [253].

```

12
13 @data
14 3434,2135,nonmatch,0,?,0,NO,30,0
15 173958,2270,nonmatch,43,94,33,NO,0,0
16 1116,1958,match,0,100,42,YES,55,62
17 148562,2901,nonmatch,0,?,0,YES,18,0
18 5098,2117,match,16,100,30,NO,87,37
19 50267,2135,nonmatch,0,?,0,NO,30,0
20 153158,3219,match,12,100,24,YES,84,0
21 50383,2135,nonmatch,0,?,0,NO,30,0
22 2906,3386,nonmatch,0,96,0,NO,100,0
23 2790,3390,nonmatch,59,100,9,NO,100,0
24 50249,2135,nonmatch,0,?,14,NO,30,0
25 2803,3456,match,0,?,22,YES,100,0
26 [...]

```

Line 1 contains the `@relation` keyword that defines the name of the relation that is represented by the file. Lines 3 to 11 contain the `@attribute` declarations that define the name and data type of each attribute. In this example the identifiers are represented by two string values. They serve as an identifier for the pair of entities that need to be compared. Identifiers have only been included for human readability and will not be filtered beforehand so that they can be included in the process of machine learning. For the time being, only two classes have been defined: `match` and `nonmatch`. If the identifier denotes a pair of identifiers that resolve to the same entity, they are labeled with the class “`match`”. If they do not resolve to the same real-word entity, they are labeled with the class “`nonmatch`”. The remaining attributes hold the similarities that were established by the application of similarity metrics.

Different alternative machine learning techniques that provide suitable decision models have been evaluated: a naive bayes classifier, a decision tree learner, a random forest learner and a support vector machine. The Weka environment has provided a framework to perform these machine learning experiments. The following paragraphs give a short summary of the preliminary experiences that have been made with these different techniques.

Naive Bayes is the model that has most resemblance to the original formalization by Fellegi and Sunter. It has the advantage of its inherent simplicity and favorable computational characteristics, and it performs reasonably well in many settings, like the identification of spam emails. After generating the machine learning model with available training data, 47606 pairs of entity descriptions were labeled as matches. It seems that the recall is very high in this case in favor of a rather low precision. The number of labeled pairs is disproportionally high and may be an indication of a lack of applicability in this context.

The features of decision tree learning have already been discussed and explored in detail for ALAP. The application of decision trees following model generation

resulted in 455 matches, which is the least amount of matches in comparison to the other models. Although the decision tree has been pruned to avoid the problem of overfitting, the number of matches seems to be too low. However, like Bayesian models, decision trees have a favorable computational complexity and the number of matches seems to be more than adequate. Additional details will be presented on how decision trees were used in ALAP later on.

An extension of the decision tree idea is the random forest, which combines a potentially high number of decision trees to achieve better results. With a random forest of 10 trees that consider 3 random features, 1168 pairs were labeled as matches. With 100 trees and 5 random features, the number can be slightly reduced to 973. This amount of automatically labeled data can be manually inspected in a reasonable amount of time. However, it is difficult to visualize the learned model for inspection and further tuning.

Support Vector Machines belong to a more recent set of algorithms that can be very powerful for classification if they are used properly. Chang and Lin [50] have developed a library that implements comprehensive functionality for the application of Support Vector Machines. This implementation has been integrated into the Weka environment by EL-Manzalawy and Honavar [97] because it is much faster than Weka SMO (Sequential Minimal Optimization). By applying Weka LibSVM (WLSVM), 1 364 831, results have been labeled as matches for a Support Vector Machine that uses a radial basis function as kernel and standard parameterization. Of course, this is a result that cannot be used for validation by a domain expert at all.

However, Hsu et al. [145] have observed that naive and unparameterized application of Support Vector Machines usually lead to unsatisfactory results. Support Vector Machines can be influenced in many ways by selecting different kinds of machine types (C-SVM, nu-SVC etc.) and kernel functions (linear, polynomial etc.). Additionally, the whole process and the kernel function in particular can be parameterized in many ways. Because of the high number of parameter combinations that are possible, the authors propose conducting simple scaling, starting with radial basis kernel and using cross validation to find suitable values for the needed parameters. These comprise in particular the penalty parameter of the error term and the γ variable of the kernel function. The parameterization is driven by sticking to a “grid search” method, which sequentially tries out different combinations and ranks them by the achieved prediction accuracy. Bilenko and Mooney [34] showed that Support Vector Machines can significantly outperform decision trees in particular with limited training data. Thus, this method should be considered for further development of ALAP.

The decision tree model has been used as a classifier for ALAP. In particular, the assessment of how well single entity description aspects contribute to a correct

decision model has been made easier this way. The fact that decision tree models are represented in a way that can easily be interpreted by humans is helpful in this situation. An entity description aspect that triggers counterintuitive decisions at some part of the tree can be an indication of an error in one of the preceding steps of the alignment workflow. For example, it could mean that wrong information has been extracted from the information sources or that training pairs have erroneous labels. Listing 12 shows the result of the model generation step as a decision tree.

Listing 12: The decision tree model after performing the training.

```

1 INFO - J48 unpruned tree
2 _____
3
4 accNumberMatch = YES
5 |   finSim <= 23
6 |   |   locSim <= 30: nonmatch (29.0/1.0)
7 |   |   locSim > 30: match (3.0)
8 |   finSim > 23
9 |   |   heiSim <= 66
10 |   |   |   finSim <= 55
11 |   |   |   |   locSim <= 19: nonmatch (9.08/2.39)
12 |   |   |   |   locSim > 19: match (7.2/1.0)
13 |   |   |   |   finSim > 55: match (9.4/0.23)
14 |   |   |   |   heiSim > 66: match (196.31/8.07)
15 accNumberMatch = NO
16 |   locSim <= 13: nonmatch (331.0/6.0)
17 |   locSim > 13
18 |   |   sumSim <= 38: nonmatch (23.0/3.0)
19 |   |   sumSim > 38: match (24.0/3.0)
20
21 Number of Leaves   :         8
22
23 Size of the tree   :        15

```

Discussing and evaluating decision trees can be difficult because minor changes in the training set can result in trees that are dramatically different. Thus, the process of finding errors in the training data needs to deal with very different trees in each iteration. However, the current decision tree that is shown in listing 12 does not bear any features that are obviously counterintuitive. In all cases, higher similarities lead to leaves that denote a class of matching instances. And lower similarities lead to subtrees that perform tests on further aspects or to leaves that denote a class of non-matching instances. However, the tree seems to be rather large and complex compared to the versions that were generated in earlier stages of the experiment. For some reasons, certain aspects like the findspot are repeatedly tested at different hierarchy levels. These observations could indicate that further effort should be put into verifying the training set.

The fact that the accession number has been helpful for compiling a training

set confirms the observation that it seems to be an aspect that generated a high information gain. But the model only decides for a positive match in combination with high similarities for the findspot, the dimensions and the depository. If the accession number does not match, the model decides for a match if the depository and the short description have high similarities. This illustrates how decision trees can achieve a certain robustness against missing and erroneous data. A considerable number of pairs belonging to the class of matches have significantly similar heights. Some matches do not have the same accession numbers. This could either be the result of imputed values due to missing data or erroneous data in the information systems. Bibliographic references do not seem to play an important role in entity resolution. Either there are only few pairs with high similarity for bibliographic information or there is a high correlation with another aspect that is more suited for achieving high information gain.

ALAP used a machine learning approach to make a decision on whether a pair of entity descriptions are a match or not. Different popular machine learning models were evaluated with varying success. Although Bayesian learning is closest to the original approach of Fellegi and Sunter, it did not perform very well. It seems that the entity resolution problem is too complex for Bayesian modeling in combination with the used data. Reasonable and helpful results were achieved by generating decision tree models. Random forests seem to be less prone to overfitting for the data that has been extracted from Arachne and Perseus.

However, to get a first impression, it is easier to evaluate the appearance of one single decision tree. Therefore, a single decision tree model is a reasonable choice for ALAP. A Support Vector Machine may significantly outperform the decision tree approach due to the small training set. Further resources should be assigned to this aspect for future experiment development. It turns out that in all cases it is very important to maintain a clean and error-free training set. Single errors can have a huge impact on the quality and performance of some models.

After evaluating different machine learning models and making a preliminary decision in favor of decision trees, the architecture of ALAP is nearly complete. The structure of the current decision tree suggests that the preceding steps provide reasonable data quality for the training set. Cross-validation reveals that 94.62 percent of entity description pairs are correctly classified. Of course, this value is only relevant if the training set adequately represents the whole population.

9.4.4 Visual Representation and User Interaction

Fellegi and Sunter propose classifying pairs of entity descriptions into three classes: certain matches, certain non-matches and possible matches. According to the authors, those pairs that have been classified as possible matches are declared subject to manual review. This approach could also be applied to the result of the classifi-

cation that was discussed in the previous section. It would certainly help domain experts to review the decisions of machine learning models. The number of entity pairs in need of review could be narrowed down. A suitable environment – one that provides an appropriate user interface – could then visualize the results. This section describes the approach that has been chosen for ALAP and recommends relevant steps for further work.

Glaser et al. [118] have proposed a user interface that helps with managing the coreference bundles that have been described above. This interface can retrieve and display the metadata that is associated with a particular reference, so that domain experts can make an informed decision on whether a group of references should belong to the same bundle. These are certainly important functional requirements that should be considered for ALAP, too. However, a system that is able to handle URI bundles has not yet been established. This is due to the fact that it is not directly related to the preparation of information for matching and has been postponed for the time being. Glaser et al. do not describe in detail the process of bundle identification by domain experts within the described environment. But it seems that full-text searches can be used to find similar references that are represented as individual singleton bundles, so that they can be joined to larger bundles.

In the course of ALAP, certain matching decisions have been made automatically by calculating similarities and feeding the results to machine learning algorithms. There are a number of machine learning models that can output the probability distribution of the class relationships of a classified instance. This additional information could be used to weight the results so that uncertain matches are presented to experts for further review first. For the time being, a rather rudimentary approach has been chosen to verify the decisions made by the machine learning component. Image 30 shows a rudimentary user interface that has been implemented in order to present pairs of entity descriptions to expert users. Entity descriptions are displayed together with associated contextual information in a tabular form.

The user interface shows an identifier for each labeled pair, allowing it to be traced through the workflow. Pairs that are tagged with a green color have already been verified by a “domain expert” as belonging to the class of matches. If they have been tagged by a red color, this means that an expert has revised the decision made by the machine learning component and assigned the pair to the class of non-matches. Pairs of entity descriptions that are not marked by green or red have been proposed by the machine learning model to be referring to the same entity but have not yet been reviewed.

The color can be changed during the process of verification by simple clicking on the identifier. For each pair certain entity description aspects have been included

Source ID	Source Description	Target ID	Target Description
3399	3399 13510 Naples, Museo Archeologico Nazionale Sculpture		Baumarmenik Rundplastik
3399	3399 13510 Naples, Museo Archeologico Nazionale Double herm Thucydides with Herodotos and Sculpture		Neapel, Museo Archeologico Nazionale Doppelherme mit Portrait des Thukydides Inschrift Rundplastik Portrait
16777	3218 152037 Thebes Archaeological Museum with nude male clenched Ptoon at from Sanctuary figure, Kouros his hands Standing sides Sculpture		Fethiye, Archäologisches Museum Weibliche Gewandstatue Rundplastik
19058	3474 2800 Museum of Fine Arts, Boston Nude (Mercury) Hermes Sculpture		Boston, Museum of Fine Arts Oberarm einer Hermes - Statue Rundplastik
19058	1965 1008 Athens, National Archaeological Museum of Statue Themis Sculpture		Athen, Nationalmuseum Statue der Themis Rundplastik
15583	2879 1001 Athens, National Archaeological Museum worshippers Hygieia Relief showing four Epione, and (?) Sculpture		Athen, Nationalmuseum Kopflose Statue der Hygieia Rundplastik
19058	3588 135192 Athens, Acropolis Museum Erichthonios nurse(?) Erichthonios, (or son from the and Iakchos, Pandrosos son, Athena Frieze Demeter Mother group perhaps Erechtheion Sculpture		Athen, Akropolis-Museum Fragment vom Fries des Erechtheion mit einer Frau mit Kind Rundplastik
15583	2918 1078 Athens, National Archaeological Museum one stele from Alexos (of figures) of Sounion original showing four Monument naiskos Grave man Deep Sculpture		Athen, Nationalmuseum Portraitkopf eines bärtigen Mannes Rundplastik Portrait
19054	3561 135191 Athens, Acropolis Museum 3/4-view to left, seated a wearing chiton Enthroned from the and figure, figure Frieze female himation. Sculpture		Athen, Akropolis-Museum Fragment vom Fries des Erechtheion mit einer Frauenfigur Rundplastik
18923	2427 41267 London, British Museum Carisur 3 Laphi Metope South Parthenon, and Sculpture		London, British Museum Metope Süd 3 des Parthenon Bauarmenik Relief
19058	2468 12963 Naples, Museo Archeologico Nazionale of Ares Ludovisi an youth Torso athletic Sculpture		Neapel, Museo Archeologico Nazionale Torso des Ares Rundplastik
244	3390 2908 Museum of Fine Arts, Boston to with neck a hair, turning and of The middle-aged curly his Portrait left short, clean-shaven head man Sculpture		Boston, Museum of Fine Arts Herakles - Kopf Rundplastik
18858	3914 938 Athens, Acropolis Museum over carrying figure or calf Calf-bearer Moschophoros, shoulders Male Sculpture		Athen, Akropolis-Museum Statue eines Mannes mit einem Kalb auf seinen Schultern, sog. Kalbträger Inschrift Rundplastik
	3016 29844 Athens, Acropolis Museum		Athen, Akropolis-Museum

Figure 30: A rudimentary user interface for verifying the decision of the machine learning model.

so that an expert can get a first impression of whether a pair describes a match or a non-match. Additionally, a link has been provided that can be used to navigate back to the entity descriptions in their original contexts. Figure 31 shows how entities can be inspected in their “original” digital context by making use of the provided hyperlinks.

The implemented user interface has helped with comprehending the decisions that were made by the machine learning model. The decisions can be analyzed and interpreted with respect to the calculated similarities for single aspects of an entity description. This has been particularly important for understanding why unexpectedly good or bad classification results were generated. By looking at the entity descriptions in their original contexts and inspecting the intermediary results of the entity resolution workflow, errors can be found and opportunities for optimizations can be discovered. The following paragraphs will introduce several examples of successful as well as counterintuitive matching examples and suggest a way to analyze the the factors that have influenced the decision of the matcher.

A portrait herm of Themistocles is described both by Arachne and Perseus.⁴⁴ Although Perseus does not record an inventory number for the entity, the matching framework has been able to suggest that both representations describe the same object. Since the accession number does not match for these objects, due to missing data in the Perseus record, the decision depends on additional values.

⁴⁴<http://arachne.uni-koeln.de/item/objekt/14124> and <http://www.perseus.tufts.edu/hopper/artifact?name=Ostia+Themistokles&object=Sculpture>.

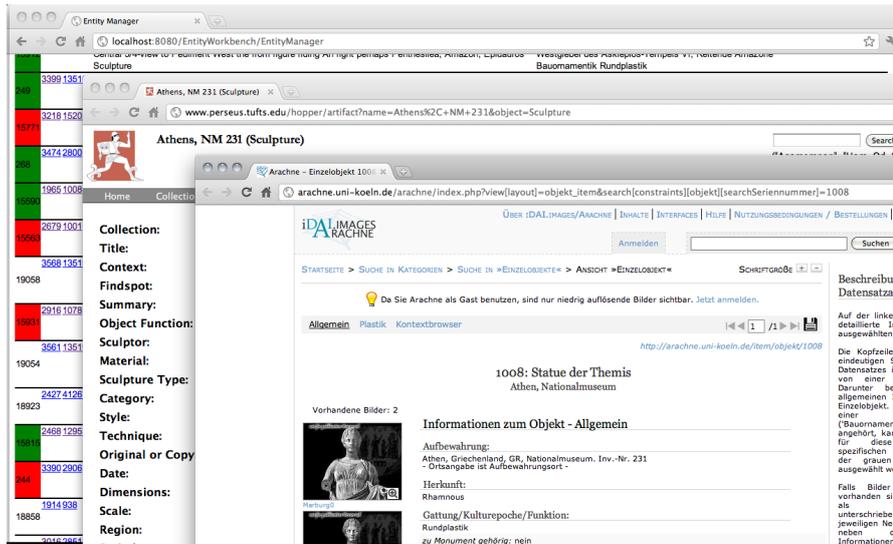


Figure 31: By using the links to the original information sources, entities can be evaluated in their “original” digital context.

According to the decision model, the similarity for the names of the depository needs to be higher than 13 percent and the similarity for the string that makes up the short description needs to be higher than 38 percent. For this particular pair of entity descriptions, the similarity for the short description string is 61 percent (‘Portraitherme des Themistokles’ and ‘of Herm Themistokles portrait Ostia’) and the similarity for the depository name is 105 percent (‘Ostia, Museo Ostiense’, ‘Ostia Museum’).⁴⁵

The same is true for the South Metope 4, which shows the head of Lapith. It is also described both by Arachne and Perseus.⁴⁶ This entity has also been identified as describing the same object without using the accession number. The similarity for the depository name is 53 percent (‘Kopenhagen, Nationalmuseet’ and ‘Copenhagen, National Museum’) and the similarity for the short description is 59 percent (‘Metope Süd 4 des Parthenon, Kopf eines Lapithen’ and ‘of 4, Head Lapith Metope youth head South Parthenon’).

Both Arachne and Perseus describe an object with the accession number 4754 in the collection of the National Archaeological Museum, Athens.⁴⁷ But the entity description does not to refer to the same entity. The short description of the

⁴⁵The soft TF/IDF measure can result in similarity values that are greater than 105%

⁴⁶<http://arachne.uni-koeln.de/item/objekt/80808> and [http://www.perseus.tufts.edu/hopper/artifact?name=Parthenon+SM.4+\(Copenhagen\)&object=Sculpture](http://www.perseus.tufts.edu/hopper/artifact?name=Parthenon+SM.4+(Copenhagen)&object=Sculpture).

⁴⁷<http://arachne.uni-koeln.de/item/objekt/146939> and <http://www.perseus.tufts.edu/hopper/artifact?name=Athens,+NM+4754&object=Sculpture>.

Arachne object 146939 says “Ostgiebel des Asklepios-Tempels IV.1, Oberteil einer weiblichen Figur”, and the short description of the Perseus object Athens, NM 4754 says “Base of Kroisos”. The current version of the decision model would not classify this pair of entity descriptions as referring to the same entity. However, this circumstance was also counterproductive in the first iterations of the bootstrapping process, where accession numbers were used extensively.

The situation seems to be more problematic for a pair of entities from Arachne and Perseus that have an identical accession number yet do not refer to the same entity.⁴⁸ In this case, the current decision model decides for a match. This is due to the fact that the similarity of the findspot names is lower than 23 percent (only 18 percent for ‘Athen, Griechenland’ and ‘Athens, Acropolis (found in 1888 to the East of the Parthenon although it is nearly the same place).’) and the similarity of the depository names is higher than 30 percent (78.0 percent for ‘Athen, Akropolis-Museum’ and ‘Athens, Acropolis Museum’).

Because the accession number is almost the only way to identify museum entities in a unique way, this outcome was not expected at all. It could be due to an error that occurred while numbering the objects or that happened at the time of the entity description creation. But since this issue seems to occur more frequently, other reasons, such as the pooling of different museum collections, seem to be more probable. When it comes to training a statistical model for entity resolution, these situations must be carefully considered. If the accession number reliably identifies entities in the majority of cases, a few bad cases may diminish the quality of the model.

In ALAP, rudimentary means were established to review the results of the machine learning component. Domain experts can rely on a user interface to get to relevant information and to make an informed matching decision. More complex means have been developed in other projects that require more sophisticated data models for the internal representation of coreference information. These should be included in future versions of the alignment implementation, especially after additional information sources have been considered for alignment. Currently, the results are not weighted and sorted according to the probability of belonging to the class of matches or non-matches. This can be remedied by using a machine learning model that allows for retrieving the needed information.

Certain requirements have been addressed in this section that allow for revising the decisions of the applied machine learning model. One requirement is a user interface that, among other things, provides means to correct coreference relations. In future versions of the alignment architecture, it should also be possible to deal with coreference bundles, which have been discussed in section ??.

⁴⁸<http://arachne.uni-koeln.de/item/objekt/41229> and <http://www.perseus.tufts.edu/hopper/artifact?name=Athens,+Acropolis+52&object=Sculpture>.

This is particularly important if more than two information systems are to be linked. Additionally, the user interface should provide means to import and export coreference information to enable sharing with other projects. This requires the implementation of appropriate back-end functionality and additional application programming interfaces.

Different pairs of entity descriptions have been highlighted in this section. They have been analyzed with respect to where and why entity resolution functions or fails. The analysis revealed that erroneous or counterintuitive data can either decrease the quality of the machine learning model or lead to wrong results at the time of model application. Although the learned model is robust against missing or erroneous data to some degree, the revealed problems can lead to false positives and negatives. Another reason for unsatisfactory results is the application of similarity metrics that do not work well due to bad data preparation. In other situations, they may be completely inadequate, and would need to be revised as well as refined.

9.5 Discussion and Future Development

State-of-the-art methods and techniques that are used in the field of knowledge discovery were evaluated and adapted for ALAP. The steps of exploratory data analysis, entity resolution and coreference management were described and critically discussed. Furthermore, the entity description aspects that have been chosen for ALAP were analyzed in greater detail. This section discusses the results of the experiment and the conclusions that can be drawn from them. Certain topics such as parameterization, workflow automation and control, making use of additional background knowledge and the question of generalizability are particularly important for further research.

Most of the techniques that have been used for ALAP depend on adequate parameterization to achieve results of high quality. For example, a suitable threshold needs to be set for similarity metrics like soft TF/IDF; further thresholds need to be determined for canopy clustering and certain machine learning models, which require very complex parameterization. In the case of applying Support Vector Machines to data, unparameterized models lead to unsatisfactory results in the majority of cases. The two basic approaches to the parameterization of machine learning techniques, or a combination of these, may lead to enhanced results: manual and automatic parameterization. Manual parameterization could be helpful in settings where a certain amount of experience with both the applied techniques and the data is present in the project.

But in the course of ALAP, it has been demonstrated that the misconfiguration of machine learning techniques can lead to results that are hardly useful. Thus, in certain cases and in situations where the parameterization heavily depends on the

peculiarities of large data sets, an automatic approach seems to be more helpful. Automatic parameterization should rely on an evaluation of the performance that can be achieved by specific parameterization and iteratively adjusting parameters. For Support Vector Machines, the optimization problem has been introduced as a systematic two-dimensional search through the parameter space for the maximal target function value.

The target function for each machine learning algorithm has a different topology, which determines the method to be used for optimization. It seems that the search space should be structured in a way that allows for localizing local or global optima, according to the example described by Hsu, Chang and Ling [145]. It has been mentioned that Support Vector Machines have been shown to outperform decision trees in entity resolution settings. Thus, they should be applied in the course of further alignment efforts by using the proposed “grid search” approach for iterative parameter refinement.

Currently, the entity resolution workflow is only partially automatized, and the necessary steps need to be manually triggered in correct order. This has been helpful because many iterations are necessary for a single component to achieve helpful results by debugging and parameterization. Thus, the process of generating a set of labeled entity description pairs is reminiscent of creating a software build. Here, dependencies are described between different function calls, so that only certain parts of a software build need to be updated if small changes have been made. If parameters have been changed at a certain step of the entity resolution workflow, those intermediary results that are part of a dependency subtree should be generated.

Parameterization should not be limited to isolated components of the entity resolution architecture but should also be applied to the workflow itself. Single components of the workflow need to be exchangeable, so that experiments with different methods or the application of multiple methods at the same time are possible. For example, it would be helpful to allow for changing the similarity metrics or to apply multiple metrics at the same time. Different machine learning models are helpful in different contexts, and different combinations can result in more robustness in certain contexts. Of course, the central parameterization of single components and the entire workflow requires a certain amount of standardization and wrapping of third-party software libraries.

The different aspects of the entity resolution experiment can be enhanced by a combination of evaluation and optimization. In this experiment, single components of the entity resolution architecture are evaluated by manual inspection and by compiling simple statistics. The determined information is then used to correct errors in the source code or to readjust parameters. Advantages of automatizing the latter have already been discussed and should be considered for future

enhancements of the alignment project.

It has also been mentioned that entity resolution projects have difficulties with quantifying the effectiveness of the applied methods. This quantification usually depends on comparing the results of entity resolution to pre-labeled reference collections that are specialized for particular domains. For domains like cultural heritage in general and archaeological data in particular, these reference collections are not yet available. Thus, it would be desirable to develop methods that allow for the quantifiable evaluation of effectiveness and efficiency, instead of deriving qualitative estimations. Since the decisions of the machine learning model will be examined by a domain expert, the results obtained can be used to form such a reference collection.

Such a reference collection would allow for more meaningful evaluations of entity resolution success. For the time being, only false positives can be analyzed, and it would be desirable to also have access to false negatives. Then state-of-the-art evaluation methods like precision, recall, f-measure and their enhancements, which have been developed in the field of information retrieval, could be applied. However, the value of explorative statistics and qualitative evaluation should not be underestimated. In the course of ALAP, they have helped with obtaining valuable information for debugging and optimization.

For pragmatic reasons, which have already been explained, the use of external sources was avoided in the course alignment experiment. In some cases, appropriate and useful sources were not yet available in the necessary formats. Nevertheless, determining the similarity of entity description aspects has worked unexpectedly well for the considered features. By making use of unstructured background knowledge and structured background knowledge that may come from a remote service, the quality of the entity resolution process can be further enhanced. It would also be interesting to evaluate the performance of token normalization by simple translation between English and German. To this end, information could possibly be drawn from Wikipedia (by exploiting language links) or Geonames (by harvesting multilingual names for places).

However, this experiment has proved that similarity metrics work well in many situations, even if the multilingual information still needs to be aligned. But the method of applying similarity metrics to match pieces of information relies on the assumption that the orthographic similarity of tokens correlates with their semantic similarity. This leads to problems in cases where identical tokens have a different meaning (homographs) or different tokens have identical meaning (synonyms). It has been mentioned that more and more information is being published online in a highly structured form, for example, with Semantic Web technology. These ways of explicitly modeling semantic relations between concepts that are represented by names should be exploited to bring these cases under control in the

future.

The scalability of the developed entity resolution approach is important for future development. Further efforts for evaluating the generalizability of the entity resolution architecture that has been developed for ALAP seem to be promising. A couple of influencing aspects have been identified that could function as indicators of the generalizability, and these will be discussed in the following paragraphs.

Generalizing the alignment architecture would certainly entail considering huge amounts of entity descriptions. Means need to be further explored that make the whole process more efficient – runtime is already a couple of hours with data from Arachne and Perseus alone. However, there are several possibilities for gaining additional efficiency, which have already been identified. Some algorithms for extracting, transforming and comparing entity descriptions have not yet been optimized. Canopy clustering is important for achieving an efficient workflow. Similar to other machine learning models, an evaluation and optimization of the used parameters could lead to smaller cluster sizes and a further reduction of comparison operations. Additionally, partitioning and comparison has not yet been distributed to several machines to make use of massive parallelization.

The heterogeneity of information to be considered for alignment is another indication of its generalizability. As argued, different forms of representation (e.g., the data model and the database schema) require certain semantic transformations. But the types of the described entities must also be comparable in the sense that they share similar features. In most cases, different cultural heritage information systems represent descriptions for different types of entities, like sculptures, buildings, topographical entities, etc. If these share several entity types, ensemble methods for machine learning could be used to train different matchers for different entity types. For collections of entity descriptions that thematically overlap but do not share the same types of entities, coreference resolution is an inadequate approach. In these cases link discovery could be an interesting approach to explicitly relate information from different information systems. A short introduction to the methods and aims of link discover has already been given.

If an initial assessment reveals that different information systems share comparable content, the amount of overlapping information should be assessed in more detail. Methods of exploratory data analysis have been introduced – among other things – as a way of estimating this overlap. Additional heterogeneity within the entity descriptions heavily affects the methods of comparison that need to be applied. Similarity metrics work best if only one national language needs to be considered, and they work reasonably well in certain contexts for information that is multilingually represented. In this case, generalizability could be limited if it is missing unstructured and structured background knowledge from external sources, for example.

The described entity resolution framework was developed in particular for ALAP. It leaves room for further optimization in several areas, particularly, with respect to the generalizability of the approach. By finding means to automatize the parameterization both of single components and the whole workflow, existing components could be enhanced and compared to alternatives. To that end, means must be established that allow for the evaluation of various intermediary results and of the framework's overall success. While the former seems to be achievable by the application of optimization techniques, the latter is still difficult for the reasons that have been discussed in this section. Furthermore, by resorting to (external) background knowledge, the effectiveness of similarity metrics can be enhanced. The latter aspect is also important for the generalizability of the framework, which depends on the aspects of scalability and heterogeneity.

ALAP has successfully aligned a significant amount of entities: The current training set consists of 633 entries (241 matches and 392 nonmatches).⁴⁹ There is plenty of room for optimizing the current implementation of the entity resolution framework. However, the application of this framework sheds light on new challenges for other cultural heritage information systems. Different entity types require the comparison of different features, which must be learned, for example, by using different machine learning models. If required background knowledge is not available and cannot be easily established, or if additional national languages must be considered, entity resolution performs badly. However, approaches such as the application of Latent Semantic Analysis create information that can be used by the framework itself and by other projects as well.

⁴⁹As of April 24, 2012.

10 Beyond Entity Resolution

The previous sections discuss the approach taken to entity resolution for ALAP. As this experiment began, information had not yet been fully integrated in this way by former efforts. This is why additional means were introduced to align entities and integrate overlapping cultural heritage content. However, it was also argued that linking information by de-duplicating database records is not an end in itself.

A user scenario was developed in section 2.3, which conveys a shared view of how linked information is valuable to end-users. Having aligned the available information, the use cases derived from this user scenario are directly supported. Additionally, further use cases can be indirectly supported by deriving additional information from aligned entities. This section reflects on the challenges of processing and conveying integrated information. It also goes into the different forms of primary and secondary uses of aligned information and discusses its relevance.

The alignment information that was generated by applying the framework to this experiment can be used as a basis for linking and fusing information in the future. This information should be kept in some kind of data structure that is accompanied by one or more indexes. It would be useful to have some kind of endpoint in the future that enables other information systems to query this discovered information. Another way to use this information would be to enrich the existing graph-like structures that include pieces of information that reference other information. “Linked Data” currently does this by using different link types, and in the case of aligned entities, this is traditionally known as “owl:sameAs”.

But certain use cases may require the information from different information systems to be merged, not only by linking the information from different sources but by fusing it as well. An entity description can be enriched with additional information by fusing the description aspects from different sources. But information that has been provided by different cultural heritage information systems will also contain conflicts and errors. Thus, additional means are necessary to resolve the conflicting and contradicting elements in entity descriptions from different sources.

However, these conflicts are not always the result of erroneous data entry, as was discussed earlier. Certain features of entities change over time or differ in relation to their context (i.e., they have different roles in different contexts). To address this problem, the provenance of information could be tracked down and the coreference bundles, which have already been introduced, could be used. In any case, information systems that contain information drawn from multiple sources and then present this information to users must figure out how to deal with these problems.

Figure 32 shows the browser prototype that was developed in the early stages of ALAP. Each described entity is represented as a thumbnail image that contains a short description. Additionally, a couple of user interface elements have been

provided to improve search and navigation. These elements comprise pagination, faceted browsing and full-text queries.



Figure 32: A browser relying on CIDOC CRM data.

The figure shows that the short summaries of the entity descriptions currently being displayed are in English. But the tokens' facets, which can be seen on the right side, are in German because the number of entities that have been extracted from Arachne is much higher. In an environment that makes use of integrated information, such as ALAP, a canonical form of presentation and processing needs to be found. In this case, search and navigation should be possible in English or German, and the presentation of entity descriptions should take into account the problems that have been described above.

CLAROS [17] follows a similar user interface metaphor and provides additional forms of visualization and interaction, such as a map and a timeline. Moreover, in the course of the project, the fusion of vocabularies is intended and has already begun for appellations of time periods that have been mapped onto distinct years. The results of information integration can also be made visible to end users by providing widgets. Widgets become part of the original user interface and provide additional information and links to other information systems. For example, Barker, Isakson and Simon [15] describe the aforementioned project PELAGIOS, which provides this form of visualization.

A number of methods that are suitable for establishing useful background knowledge were used throughout ALAP. As argued, vocabularies that have been crafted top-down will not properly represent the realities of common data sets. Therefore, the information that has been linked by supervised machine learning should be used to extract multilingual vocabularies. The extracted vocabularies

could be encoded in a machine readable way (i.e., as SKOS), so that they can be easily transmitted, shared and exploited by other projects.

Currently, the entity resolution framework mainly considers entity description aspects that are quite granular and short textual descriptions. But cultural heritage information systems like Arachne and Perseus also use longer textual descriptions to describe entities. Thus, a valuable outcome of entity resolution could be parallelized textual information that can be used to train models for terminology and topic extraction. For example, Chatterjee, Sarkar and Mishra [53] describe a way of analyzing co-occurrences for terminology extraction and building cross-lingual dictionaries.

Throughout ALAP, methods like Latent Semantic Analysis were applied to multilingual data. Variations of these methods seem to also be applicable to the information in ALAP. Additionally, latent topic analysis not only is helpful for entity resolution but also for clustering operations. In this case, entity descriptions that have a high degree of similarity with respect to their topic structure can be shown together to enhance navigation. And the tokens that are used in entity descriptions can also be clustered to create vocabularies that enhance searching content.

Zhang, Mei and Thai [265] propose an approach for cross-lingual latent topic extraction that deals with unaligned textual information. Although entity resolution does not seem to contribute to this approach, it could be usefully applied to entity resolution in the future. A combination of approaches that make use of both controlled vocabularies with rich semantic annotations and probabilistic models for information that is represented in textual form seems to be useful. These probabilistic models can support and enhance the results of rigid formalized methods in the future.

The majority of entity description aspects that were considered for entity resolution do not contain full-text information. Therefore, state-of-the-art machine translation methods were not used in ALAP. Yet a couple of entity description aspects have been represented as longer textual descriptions, as discussed. For example, an entity description in Perseus “Portrait of a man” refers to the same entity as a description in Arachne, which states “Portraitkopf eines Mannes (Maximianus Herculeus?)”.

Furthermore, both information systems provide additional information about the condition of the entity in the form of a short description: “Fragment(e); Nase, Mund, Kinn und Hals fehlen; in zwei Fragmenten erhalten und zusammengesetzt; gelbe Patina” and “Comprised of two fragments preserving some of a male head: missing the nose, mouth, chin, and much of the left side fo[sic!] the face. The battered surfaces have acquired a yellowish patina.” Machine translations function well with textual material that includes a good amount of parallel text, but they

would probably fail in highly specialized domains like archaeology. But it seems that cross-lingual vocabulary extraction and topic analysis could enhance these models, making them extremely useful in the future.

The approaches that have been discussed can enhance future entity resolution and linking efforts. It is difficult to properly reflect on the terminology used in the data if it is in the form of a structured vocabulary that has been manually crafted. Thus, cross-lingual terminology structures and topic models should be deduced from the information that has already been aligned to reflect real data in cultural heritage information systems. In the future, a combination of both approaches would probably be optimal for establishing usable background knowledge.

The results of ALAP should be viewed as a starting point for future research. That which has been achieved can be considered as part of a bootstrapping process that needs to be expanded upon. The various ways to methodically extend the architecture and to integrate additional background knowledge have been discussed and should continue to be considered in the future. Aligning entity representations for material objects like sculptures will also help with aligning complementary material and immaterial entities like buildings, reproductions or places. For example, according to the Context Attraction Principle, entity descriptions that refer to the same building are more likely referring to the same entity.

In the course of ALAP, several means were introduced to measure the degree of information quality. Information quality is relevant in environments that strive for massive information integration because it influences the success of the alignment process. The means established for ALAP were used to gain and maintain a certain degree of information quality for its intended use. The information that has been extracted for the entity resolution framework could also be useful in other contexts and should be published for re-use. Other information systems may then implement, for example, special finding aids that rely on this structured and top-shelf data.

The focus of ALAP was to discover (coreference) relations between entity descriptions. Moreover, other methods that have been explored in the field of knowledge discovery and data mining could contribute valuable information. Link prediction is a task of knowledge discovery that can either be done by analyzing features of entities themselves or by observing the existing links. It would also be interesting and helpful to explore methods that predict links between the entities themselves and enrich the entity descriptions with this information. Link mining is also viewed as a discipline that makes use of the existing links between information elements in order to derive new knowledge. A virtuous cycle between mining links from existing information and exploiting links information to derive new information should be established in the future.

ALAP itself was developed in an iterative fashion – by pursuing gradual im-

provements. With each iteration, the quality of the available information improves as well as the knowledge of the applied methods and techniques. Thus, the components that are described as being part of the entity resolution framework are also part of this iterative process. As discussed in this section, different scenarios may benefit from aligned cultural heritage information. Additionally, a number of involved techniques can also contribute valuable information to the endeavor of information alignment itself.

On the one hand, the entity resolution workflow benefits from the information that is generated by each iteration. On the other hand, other projects can make use of the generated information. The entity resolution workflow should consider information that is provided by other projects as well. In the course of ALAP, a preliminary corpus of background knowledge was established in the form of explicit links and implicit topic space information. This knowledge – provided that it will be more comprehensive in the future – could also be used in other contexts to enhance, for example, information retrieval, vocabulary extraction and machine translation.

However, the corpus of aligned entities is still too small to achieve satisfying results in the mentioned fields. To enhance the quality of the provided background information, it is important to work towards aligning more entities of different types and to consider additional multilingual information systems in the future. Then, the gained knowledge becomes more valuable for the above-mentioned purposes. In particular, both explicitly structured forms of representation like Semantic Web technology and implicitly unstructured forms of representation like semantic topic spaces could be beneficial.

Additionally, means were established that are suitable to measure and enhance information quality, supporting the described alignment tasks. The information alignment methodology that was successfully applied can also be used for other knowledge discovery tasks beyond entity resolution. For example, clustering is indispensable if the aim is to gain efficiency for entity resolution. Furthermore, it can be used to present users with entity descriptions that are possibly related without performing strict entity resolution.

In summary, useful information was generated at different steps of the described alignment architecture. Although this knowledge was mainly used to feed different components of the entity resolution framework in ALAP, it could also be used as background information in other contexts. Future opportunities to apply this knowledge and to use this methodology to implement scenarios beyond entity resolution have been discussed throughout this section. The application of knowledge discovery techniques to cultural heritage information could unleash considerable research potential.

11 Summary and Outlook

Scientists dealing with cultural heritage information, in many relevant and productive scenarios, depend on the processing and provisioning of interoperable systems. Foundational tasks like source criticism become more seamless with such systems, and additional scientific questions can be tackled by them as well. Almost all forms of advanced information manipulation such as knowledge discovery and visualization benefit from interoperability. Consequently, efforts to align the information from Arachne and Perseus, primarily by mapping the application data models and schemas to the CIDOC CRM, began in 2007 and are ongoing. ALAP, which has been described and discussed in the previous chapters, strives to tackle the challenge of aligning the cultural heritage entities that are documented in these systems. Not only should the models and schemas that organize entity descriptions be represented in a way that enables sharing and re-use, but the entity descriptions should be treated in the same manner.

However, interoperability is not only a technical challenge. It involves diverse stakeholders like political, legal and organizational actors. Therefore, distributed information systems are usually crafted in a way that seeks a compromise between the needs of different stakeholders. Additionally, to establish interoperability, increased formality to information systems must be introduced, on the level of the schema and the instance. At the same time, this is required both before and after the information has been aligned. Due to the involved effort, smaller groups should connect with related projects that share the same pressing needs. More importantly, clearly defined scenarios and use cases should be followed when integrating information systems; this is necessary / must be in place to produce goal-driven system development and a realistic assessment of costs.

The interoperability of multiple (cultural heritage) information systems should be seriously pursued by researchers in the future, which is no easy task. To establish interoperability, it is fundamental that the information be aligned in a way that can be exploited by different actors, which includes technical systems. Therefore, resolving the cultural heritage entity descriptions that have been extracted from Arachne and Perseus is at the core of the described alignment experiment. More specifically, ALAP focused on the required workflow to facilitate resolving entity descriptions in the field of cultural heritage information management. A number of topics that are either directly or indirectly related to aligning cultural heritage information were critically discussed.

Recent developments in knowledge representation and processing such as the Semantic Web and the CIDOC CRM were evaluated for this task. Entity resolution itself is described as being an integral part of the knowledge discovery workflow. Additionally, the role of data mining methodology was highlighted, which is comprised of approaches that make use of machine learning to prepare the information

for subsequent processing, such as classification and clustering. Additionally, the theoretical implications for entity alignment were considered as well as the representation and sharing of information about sameness and difference.

These considerations form the foundation of the entity resolution framework used here, and they enabled the implementation and execution of ALAP. The essential components of this framework, such as partitioning, model selection, model generation and performance evaluation, were discussed with respect to their applicability to cultural heritage information. A clear separation was made between the approaches that were already considered for ALAP and the advanced topics that remain to be implemented in the future. By considering these advanced topics and allocating appropriate resources, the quality of the current entity resolution approach could be significantly enhanced.

Each aspect of the knowledge discovery workflow was further analyzed by describing, discussing and interpreting unprocessed information, statistical summaries and generated statistical models. The opportunities and challenges were discussed, and recommendations for future enhancements were given. Finally, the applications of the generated alignment information and the background knowledge in different contexts beyond entity resolution were contemplated. It was argued that the methodology being used for ALAP would probably also be beneficial in other contexts, such as in knowledge discovery and information retrieval for cultural heritage information. The following paragraphs present the findings of the discussed topics in greater depth.

The purpose of ALAP was to analyze the characteristics of (semi-)automatic information alignment in the field of cultural heritage. Information systems that strive to process integrated information must deal with complex architectures and infrastructures that are distributed per definition. Centralized information systems reduce administrative tasks and result in higher efficiency due to reduced latency for data provisioning. On the other hand, distribution allows for parallel processing that can enhance efficiency if the task is parallelizable. To match entity descriptions, the best of both approaches should be combined to determine similarities.

However, the biggest obstacle for establishing such systems is semantic heterogeneity. Therefore, an important concept for information integration is semantic correspondence, which needs to be discovered between schema and data elements. Entity descriptions that are organized along a standardized schema must be matched and mapped to enable further exploitation. To that end, the foundations of information matching and mapping were illuminated and state-of-the-art methods were studied in light of cultural heritage. Since the internal data models and database schemas of Arachne and Perseus have already been mapped onto the CIDOC CRM (serialized as RDF) by other projects, ALAP was strongly biased

towards integrating the descriptions of archaeological entities.

Research centered on Semantic Web concepts is biased towards formally representing information and defining a system of strict inference for it. However, the information that is subject to mapping and matching is often not formalized in a way that can be accessed by these strict formalizations. Semantic Web technology is mainly concerned with description logic, which can be seen as a set of extensions to first-order logic. In particular, very limited means are implemented to support inductive inference, which would enable the identification of implicit coreference relations. To match entity descriptions, types of inference that have only recently been applied to distributed information are often required.

However, the scope of the Semantic Web is still subject to different interpretations, ranging from classical formalizations and reasoning to newer “soft” approaches. The importance of Semantic Web concepts will increase in the future as inductive and fuzzy reasoning are investigated in more detail. Linked Open Data was also mentioned, which describes the most prominent method of publishing and exchanging formalized information on the Web. Semantic Web concepts were not considered for internal information representation or processing in ALAP. But these concepts could become useful for consuming external background knowledge and for publishing the information that is generated by the entity resolution framework itself.

Representing, sharing and manipulating cultural heritage information requires adequate data models and schemas. For this reason, the CIDOC CRM was introduced as a reference model that aims at guiding the implementation of domain-specific models and schemas. It is considered for a number of projects that must deal with digital material in the fields of classics and archaeology. Semantic Web concepts are used to implement the CRM. But, unfortunately, different interpretations and technical implementations of the CRM introduce additional heterogeneity, which must be dealt with. The function of the CRM in ALAP depended on the role of Semantic Web concepts.

The methodology that was used to implement the entity resolution framework is mainly rooted in the fields of information retrieval, knowledge discovery and data mining. Entity resolution is commonly considered as a form of data cleaning. But interpreting entity resolution as a full knowledge discovery task in itself is useful. Therefore, it should be implemented along the principles of knowledge discovery. The methods that are relevant for entity alignment in the cultural heritage area were discussed, and relevant software components were identified. Since a large part of relevant information is represented in textual form, specific concepts and models from information retrieval turned out to be helpful.

Regular expressions, or simple forms of entity extraction, were used to get hands-on structured information for consecutive processing. In the majority of

cases, TF/IDF in combination with soft term matching was used to determine the similarity of encountered entity features. Additionally, Latent Semantic Analysis was explored to attenuate the problem of heterogeneous vocabularies. However, according to the principle of “garbage in – garbage out”, the data mining process relies on a sufficient degree of data quality. This aspect was controlled by analyzing and scrubbing data with the help of exploratory data mining techniques and manual inspection.

Furthermore, philosophical implications were discussed which should be considered for future elaboration of the entity resolution framework. The fact that ontological entities tend to change over time has implications for the comparison and management of cultural heritage entity descriptions. Among other things, only those features that are comparable during a particular period of time and with respect to a certain context should be considered for comparison. If co-referring entity descriptions are discovered, the coreference information should be represented in a way that is semantically adequate. The approach of using *owl:sameAs* to express the identity of things is too rigid in many information integration settings. Bundling entity descriptions referring to entities that share a close causal relationship should be preferred over stating strict identity.

Different state-of-the-art entity resolution frameworks were used as a foundation for the elaboration of a suitable alignment architecture. These frameworks comprised software components that were arranged in a way to implement a particular entity resolution methodology. Due to the vastness and the heterogeneity of the material, a good understanding of the data cannot be established by manual inspection alone. Therefore, an architecture that relies on semi-automatic model generation as opposed to hand-crafted models was used. Canopy clustering was brought into play to reduce the number of needed comparisons. Subsequently, the presorted entities were treated by a matcher that uses decision tree models. These models allow for manually inspecting the decision process.

The aforementioned efforts served to elaborate a foundation on which to conduct ALAP. The purpose of this experiment was twofold: to test the elaborated approach to align cultural heritage information and to estimate the potential of future information alignment efforts in that area. In particular, practical experience was acquired by implementing the approaches that were first elaborated theoretically. Most importantly, the expected success was assessed by using realistic information.

The preliminary analysis revealed that the amount of information about sculptures (> 41 000 for Arachne and > 2 000 for Perseus) and vases (> 4 500 for Arachne and > 1 500 for Perseus) that can be extracted with reasonable effort was promising. Additionally, the number of needed comparisons for aligning information about sculptures posed an interesting scalability problem. A number of

heuristics including q-grams and the comparison of accession numbers were used in an iterative manner to compile an initial training set for bootstrapping a statistical model. An overlap of about 140 records was discovered, each containing at least five attributes that qualify for determining similarities.

As expected, for the unpartitioned data sets, it was easier to acquire unstructured and semi-structure information (short summary: 92% for Perseus and Arachne) for comparison than to extract highly structured values (height: 50% for Arachne and 77% for Perseus). The extraction results were slightly better for the actual sample data set that was used to build the statistical model (98% and 65% for Arachne; 97% and 88% for Perseus). The preferred entity description aspects were those that are either represented as cardinal values or that are as language-independent as possible. The Context Attraction Principle served as a guiding principle for the alignment work. The majority of features were compared by using the soft TF-IDF weight, because their content contains orthographic similarities. These features comprise information about bibliographic references, depositories, findspots, accession numbers and dimensions. The short descriptions of entities were also analyzed.

Because the amount of entities to be compared was rather high, means were implemented to make the alignment more efficient. Canopy clustering was therefore supplied with information about the depository, inventory number, and the short description of entities. A minimum amount of interactivity was established by decoupling the comparison process as much as possible. This enabled iterative parameterization and the analysis of intermediate results. Additionally, a simple user interface was provided so that the domain experts can identify and correct false positives. Erroneous or counterintuitive data harms both model generation and application. However, by inspecting and correcting a number of discovered false positives, these effects were alleviated.

The elaborated entity resolution approach helped to explore information alignment for Arachne and Perseus. In the future, it would be useful to establish further automatization, such as analysis and parameterization, because these tasks still require significant manual intervention. In order to cover a larger set of information, the framework should be further generalized by adapting and enhancing similarity measures and machine learning models. Approaches like Latent Semantic Analysis are demanding with respect to implementation and computational complexity. But they are promising if multiple national languages and varying vocabulary must be dealt with.

Information alignment is not an end in itself but makes a number of scenarios possible, such as information visualization, interaction, statistical analysis and information retrieval. The aligned information forms valuable background knowledge for the information alignment itself. In the course of ALAP, a preliminary

corpus of background knowledge was established, which still needs to be further elaborated. Additionally, it may be desirable to broaden the scope of the applied methodology to other fields such as link prediction, which strives to identify possibly related entities without performing strict resolution. The application of such knowledge discovery approaches to cultural heritage information could unleash considerable research potential.

A proper strategy to evaluate the performance of the preliminary entity resolution framework is still missing. Nevertheless, the results are promising and there is no indication that one should refrain from putting additional effort into enhancing the framework. Rather, different ways to improve the workflow have been identified, in particular, concerning information extraction, determination of similarities and model generation. In the future, advanced means of information extraction should also be applied to information that is represented as longer text. Considering additional background information that is domain-specific can significantly boost the effectiveness of similarity metrics.

To further enhance the suggested framework, more advanced machine learning models should be employed because these represent classification rules for entity resolution better than decision trees. Ensemble learning is a promising approach for evaluating multiple types of entities. According to this approach, different learners or even different learning models can be combined. Then, various machine learning models can assume responsibility for a matching decision in different matching contexts. Such a context consists of the type of entities to be matched and of the certain constellations of frequently occurring description patterns.

In summary, a number of methods that are genuinely developed in data mining research turned out to be central for semi-automatic information alignment. Therefore, ALAP was organized along the workflow of knowledge discovery and data mining. In the course of the experiment, these methods and techniques were applied to cultural heritage information. Unlike in many other domains, information in the field of cultural heritage tends to be highly heterogeneous and, in some cases, extremely vague.

A considerable amount of information about the humanities is represented in textual or semi-structured form. Methods that are developed as part of text mining research can be directly applied to these forms of representation to achieve useful results. However, the quality of the statistical models that are used for entity resolution depend, to a high degree, on information that is comparable on a rather granular level. Therefore, information extraction should always be applied first, if possible.

The methodology of knowledge discovery has considerable potential for enriching humanities information in general and cultural heritage information in particular. ALAP explored how knowledge discovery could be applied to multiple

information sources in order to predict related information. However, this is only a preliminary step to enable the further re-use of information. The main purpose of information alignment is to provide a more comprehensive information source for searching, browsing and other forms of processing. For example, Manovich [171] forms the notion of “cultural analytics” to describe a set of techniques that can be applied to very large amounts of cultural data.⁵⁰ Although the focus is on visual analytics, the questions that Manovich raises certainly are transferrable to the cultural heritage domain as well.

A growing amount of information is being made publicly available according to Semantic Web concepts and the principles of Linked Data. Information that is represented in these ways often exhibits a rich graph-like structure. Against this background, entity resolution can be described as discovering a particular type of relation between entity descriptions: coreference. Link mining, on the other hand, is also described as a suitable way of enriching entity descriptions by exploiting documented links between the entities themselves. In order to generate information for the Semantic Web, methods that allow for link discovery, either by evaluating described features or existing links, are particularly interesting.

Therefore, a considerable amount of humanities and cultural heritage information is made available in a structured way, and projects are becoming aware of the role of data quality management. But at the same time, information that is represented in more implicit forms like generated statistical models and semantic spaces can significantly support this process. Each form of representation can benefit from the other. This process of semantic accumulation can be driven by the application of knowledge discovery methodology. For example, statistical models can be used to predict the features of entities as well as the links between them. Semantic spaces can be used to find topics and to explicitly relate described topics.

Matching related information from different humanities information sources is a rather resource intensive endeavor. But at the same time, information alignment is an important enabling factor for advanced forms of information processing. Therefore, the activities of entity resolution in particular and knowledge discovery in general should not be driven by arbitrary goals. Information management and processing need to be implemented with well-defined aims in mind. For example, the process of software development should be focused on meeting user needs (i.e., being discovered and analyzed by scenarios and use cases) as best as possible.

These needs influence the choice of data models, schemas and algorithms, which must be adequate for the task and efficient for processing at the same time. In many situations a compromise must be found between formality that involves additional complexity and challenges of efficiency. The same holds true for the actual content that is organized along data models and schemas. Erratic mining of re-

⁵⁰A short introduction to the topic of cultural analytics can be found at [172].

lated information, predicting, and documenting features of entities will most likely lead to a set of rich information, but that is nevertheless only partially relevant. Only the information that is relevant for a certain task should be aggregated. This fosters the effective and efficient processing of large amounts of information.

Aligning information from different humanities resources in general and cultural heritage information systems in particular becomes one of the most important research-enabling activities. However, due to the requirements and habits of information representation in these fields, information alignment is a rather challenging research area at this time. ALAP helped to study entity resolution workflows that are both effective and efficient. Initial experience was gained by analyzing the information sources with respect to the needs of information alignment. A framework was composed by choosing, parameterizing and combining adequate components. The interrelation between implicit and explicit forms of background knowledge representation played an important role and should be further investigated. Finally, the results that were achieved in the course of ALAP are promising, and additional resources should be allocated to enhance the explored methodology for cultural heritage information alignment.

References

- [1] Ian F. Alexander and Neil Maiden. *Scenarios, Stories, Use Cases: Through the Systems Development Life-Cycle*. John Wiley & Sons, August 2004.
- [2] alias-i. Lingpipe 4.1.0. <http://alias-i.com/lingpipe/>, 2008. Last retrieved August 20, 2012.
- [3] Amazon Web Services LLC. Amazon Web Services. <https://aws.amazon.com/>. Last retrieved August 14, 2012.
- [4] American Council of Learned Societies Commission on Cyberinfrastructure for the Humanities and Social Sciences. Our cultural commonwealth: The final report of the ACLS commission on cyberinfrastructure for the humanities and social sciences. <http://www.acls.org/cyberinfrastructure/OurCulturalCommonwealth.pdf>, 2006. Last retrieved September 20, 2012.
- [5] Rohit Ananthakrishna, Surajit Chaudhuri, and Venkatesh Ganti. Eliminating fuzzy duplicates in data warehouses. In *Proceedings of the 28th international conference on Very Large Data Bases*, pages 586–597, Hong Kong, China, 2002. VLDB Endowment.
- [6] Grigoris Antoniou and Frank van Harmelen. *A Semantic Web Primer*. MIT Press, 2nd edition, 2008.
- [7] Apache Mahout Community. Canopy Clustering. <https://cwiki.apache.org/MAHOUT/canopy-clustering.html>, February 2011. Last retrieved August 20, 2012.
- [8] Arachne. Die Skulpturen der Berliner Antikensammlung. <http://arachne.uni-koeln.de/drupal/?q=de/node/164>, September 2009. Last retrieved August 24, 2012.
- [9] Arachne. The Hellespont Project: Integrating Arachne and Perseus. <http://arachne.uni-koeln.de/drupal/?q=en/node/231>, 2012. Last retrieved August 20, 2012.
- [10] David Arnold and Guntram Geser. Research agenda for the applications of ICT to cultural heritage. http://public-repository.epoch-net.org/publications/RES_AGENDA/research_agenda.pdf, February 2007. Last retrieved August 20, 2012.

- [11] Austrian National Library. Europeana Connect. <http://www.europeanaconnect.eu/index.php>, October 2011. Last retrieved August 20, 2012.
- [12] Alison Babeu, David Bamman, Gregory Crane, Robert Kummer, and Gabriel Weaver. Named entity identification and cyberinfrastructure. *Research and Advanced Technology for Digital Libraries*, pages 259–270. Springer, 2007.
- [13] Ricardo Baeza-Yates. Relating content through Web usage. In *Proceedings of the 20th ACM Conference on Hypertext and Hypermedia*, HT '09, pages 3–4, New York, NY, USA, 2009. ACM.
- [14] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, May 1999.
- [15] Elton Barker, Leif Isaksen, and Rainer Simon. PELAGIOS. interlinking ancient world research resources through place. <http://de.slideshare.net/aboutgeo/pelagios-project-overview>, February 2012. Last retrieved August 24, 2012.
- [16] Rohan Baxter, Peter Christen, and Tim Churches. A comparison of fast blocking methods for record linkage. In *ACM SIGKDD '03 Workshop on Data Cleaning, Record Linkage, and Object Consolidation*, 2003.
- [17] Beazley Archive. CLAROS - Classical Art Research Centre Online Services – The Univeristy of Oxford. <http://www.clarosnet.org/about.htm>, July 2010. Last retrieved August 20, 2012.
- [18] Khalid Belhajjame, Helena Deus, Daniel Garijo, Graham Klyne, Paolo Missier, Stian Soiland-Reyes, and Stephan Zednik. Prov model primer. <http://dvcs.w3.org/hg/prov/raw-file/default/primer/Primer.html>, August 2012. Last retrieved August 20, 2012.
- [19] Michael K. Bergman. White paper: The Deep Web. Surfacing hidden value. *The Journal of Electronic Publishing*, 7(1), August 2001.
- [20] Tim Berners-Lee. Information management: A proposal. Technical report, CERN, March 1989.
- [21] Tim Berners-Lee. Semantic Web roadmap. <http://www.w3.org/DesignIssues/Semantic.html>, September 1998. Last retrieved August 20, 2012.

- [22] Tim Berners-Lee. The Semantic Web as a language of logic. <http://www.w3.org/DesignIssues/Logic.html>, August 1998. Last retrieved August 20, 2012.
- [23] Tim Berners-Lee. Linked data – design issues. <http://www.w3.org/DesignIssues/LinkedData.html>, June 2009. Last retrieved August 20, 2012.
- [24] Tim Berners-Lee, Yuhsin Chen, Lydia Chilton, Dan Connolly, Ruth Dhanaraj, James Hollenbach, Adam Lerer, and David Sheets. Tabulator: Exploring and analyzing linked data on the Semantic Web. In *Proceedings of the 3rd International Semantic Web User Interaction Workshop*, 2006.
- [25] Tim Berners-Lee, R. Fielding, and L. Masinter. Uniform Resource Identifier (URI): Generic syntax. <https://tools.ietf.org/html/rfc3986>, January 2005. Last retrieved August 20, 2012.
- [26] Tim Berners-Lee and Mark Fischetti. *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. Harper, San Francisco, CA, USA, 1st edition, 1999.
- [27] Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, May 2001.
- [28] Abraham Bernstein, Esther Kaufmann, Christoph Kiefer, and Christoph Bürki. SimPack: A generic Java library for similarity measures in ontologies. <https://files.ifi.uzh.ch/ddis/oldweb/ddis/research/simpack/index.html>, April 2008. Last retrieved August 20, 2012.
- [29] Patrick Beuth. PR-Agentur prahlt mit Manipulation von Wikipedia und Google. <http://www.golem.de/1112/88311.html>, December 2011. Last retrieved August 20, 2012.
- [30] Indrajit Bhattacharya and Lise Getoor. A Latent Dirichlet model for unsupervised entity resolution. In *SIAM International Conference on Data Mining*, 2006.
- [31] Indrajit Bhattacharya and Lise Getoor. Collective entity resolution in relational data. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1), March 2007.
- [32] Mikhail Bilenko. Adaptive blocking: Learning to scale up record linkage. In *Proceedings of the 6th IEEE International Conference on Data Mining, ICDM-2006*, pages 87–96, 2006.

- [33] Mikhail Bilenko, Raymond Mooney, William Cohen, Pradeep Ravikumar, and Stephen Fienberg. Adaptive name matching in information integration. *IEEE Intelligent Systems*, 18(5):16–23, 2003.
- [34] Mikhail Bilenko and Raymond J Mooney. Adaptive duplicate detection using learnable string similarity measures. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD-2003, pages 39–48, 2003.
- [35] Mikhail Bilenko and Raymond J. Mooney. On evaluation and training-set construction for duplicate detection. In *Proceedings of the KDD '03 Workshop on Data Cleaning, Record Linkage and Object Consolidation*, KDD '03, pages 7–12, 2003.
- [36] Mikhail Yuryevich Bilenko. *Learnable Similarity Functions and Their Application to Record Linkage and Clustering*. PhD thesis, Faculty of the Graduate School of The University of Texas at Austin, 2006.
- [37] Ceri Binding, Keith May, and Douglas Tudhope. Semantic interoperability in archaeological datasets: Data mapping and extraction via the CIDOC CRM. In *Proceedings of the 12th European conference on Research and Advanced Technology for Digital Libraries*, ECDL '08, pages 280–290. Springer-Verlag Berlin, 2008.
- [38] Ceri Binding and Douglas Tudhope. Using Terminology Web Services for the Archaeological Domain. In *Proceedings of the 12th European Conference on Research and Advanced Technology for Digital Libraries*, ECDL '08, pages 392–393. Springer-Verlag Berlin, 2008.
- [39] Chris Bizer. The D2RQ Platform – accessing relational databases as virtual RDF graphs. <http://d2rq.org/>, June 2012. Last retrieved August 27, 2012.
- [40] Christian Bizer, Tim Berners-Lee, and Tom Heath. Linked Data – the story so far. *International Journal on Semantic Web and Information Systems*, 5(3), 2009.
- [41] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. DBpedia – a crystallization point for the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154–165, September 2009.

- [42] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *The Journal of Machine Learning Research*, 3:993–1022, March 2003.
- [43] George Boole. *The Mathematical Analysis of Logic: Being an Essay Towards a Calculus of Deductive Reasoning*. Cambridge Library Collection - Mathematics. Cambridge University Press, 2009.
- [44] Paolo Bouquet, Heiko Stoermer, Claudia Niederee, and Antonio Maña. Entity Name System: The Back-Bone of an Open and Scalable Web of Data. In *Proceedings of the 2008 IEEE International Conference on Semantic Computing*, pages 554–561, Washington, DC, USA, 2008. IEEE Computer Society.
- [45] Mikio Braun, Soeren Sonnenburg, and Cheng Soon Ong. MLOSS. Machine Learning Open Source Software. <http://mloss.org/software/>, 2011. Last retrieved August 20, 2012.
- [46] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, October 2001.
- [47] Dan Brickley and R.V. Guha. RDF Vocabulary Description Language 1.0: RDF Schema. <http://www.w3.org/TR/rdf-schema/>, February 2004. Last retrieved August 22, 2012.
- [48] Vannevar Bush and Jingtao Wang. As we May Think. *Atlantic Monthly*, 176:101–108, July 1945.
- [49] CERN. Colt. <http://acs.lbl.gov/software/colt/>, September 2004. Last retrieved August 22, 2012.
- [50] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, April 2011. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>. Last retrieved August 22, 2012.
- [51] Sam Chapman. Simmetrics. <http://sourceforge.net/projects/simmetrics/>, June 2011. Last retrieved August 22, 2012.
- [52] Christian Charras and Thierry Lacroq. Sequence comparison. <http://www-igm.univ-mlv.fr/~lecroq/seqcomp/index.html>, February 1998. Last retrieved August 22, 2012.

- [53] Diptesh Chatterjee, Sudeshna Sarkar, and Arpit Mishra. Co-occurrence graph based iterative bilingual lexicon extraction from comparable corpora. In *Proceedings of the 4th Workshop on Cross Lingual Information Access*, pages 35–42, Beijing, China, August 2010. Coling 2010 Organizing Committee.
- [54] Jiangping Chen, Fei Li, and Cong Xuan. A preliminary analysis of the use of resources in Intelligent Information Access research. In *Proceedings 69th Annual Meeting of the American Society for Information Science and Technology (ASIST)*, volume 43, 2006.
- [55] Zhaoqi Chen, Dmitri V. Kalashnikov, and Sharad Mehrotra. Exploiting relationships for object consolidation. In *Proceedings of the 2nd International Workshop on Information Quality in Information Systems, IQIS '05*, pages 47–58, New York, NY, USA, 2005. ACM.
- [56] Zhaoqi Chen, Dmitri V. Kalashnikov, and Sharad Mehrotra. Exploiting context analysis for combining multiple entity resolution systems. In *Proceedings of The 35th SIGMOD International Conference on Management of Data, SIGMOD '09*, pages 207–218, New York, NY, USA, 2009. ACM.
- [57] Peter Christen. Automatic training example selection for scalable unsupervised record linkage. In *Proceedings of The 12th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, PAKDD'08*, pages 511–518, Berlin, Heidelberg, 2008. Springer-Verlag.
- [58] Peter Christen. Febrl: a freely available record linkage system with a graphical user interface. In *Proceedings of The Second Australasian Workshop on Health Data and Knowledge Management*, volume 80 of *HDKM '08*, pages 17–25, Darlinghurst, Australia, 2008. Australian Computer Society, Inc.
- [59] Peter Christen and Karl Goiser. Quality and complexity measures for data linkage and deduplication. In *Quality Measures in Data Mining*, pages 127–151. Springer, 2007.
- [60] CIDOC CRM SIG. The CIDOC CRM: Applications. http://www.cidoc-crm.org/uses_applications.html, July 2012. Last retrieved August 22, 2012.
- [61] Erin Coburn, Richard Light, Gordon McKenna, Regine Stein, and Axel Vitzthum. LIDO - Lightweight Information Describing Objects. Version

- 1.0. <http://www.lido-schema.org/schema/v1.0/lido-v1.0-specification.pdf>, November 2010. Last retrieved August 22, 2012.
- [62] Edgar F. Codd. A relational model of data for large shared data banks. *Communications of the ACM*, 13(6):377–387, June 1970.
- [63] William W. Cohen, Pradeep Ravikumar, and Stephen E. Fienberg. A comparison of string distance metrics for name-matching tasks. In *Proceedings of the IJCAI-03 Workshop on Information Integration on the Web (IIWeb-03)*, pages 73–78, 2003.
- [64] Cesare Concordia, Stefan Gradmann, and Sjoerd Siebinga. Not (just) a Repository, nor (just) a Digital Library, nor (just) a Portal: A Portrait of Europeana as an API. In *World Library and Information Congress: 75th IFLA General Conference and Council*, August 2009.
- [65] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. The MIT Press, 2 edition, 2001.
- [66] Thomas M. Cover. *Elements of Information Theory*. Wiley, New York, 1991.
- [67] Gregory Crane, David Bamman, Lisa Cerrato, Alison Jones, David Mimno, Adrian Packel, David Sculley, and Gabriel Weaver. Beyond digital incunabula: Modeling the next generation of digital libraries. In *Proceedings of the 10th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2006)*, volume 4172 of *Lecture Notes in Computer Science*. Springer, 2006.
- [68] Gregory Crane and Clifford Wulfman. Towards a cultural heritage digital library. In *Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries*, JCDL '03, pages 75–86, Washington, DC, USA, May 2003. IEEE Computer Society.
- [69] James M. Crawford and Benjamin Kuipers. ALL: Formalizing Access Limited Reasoning. In *Principles of Semantic Networks: Explorations in the Representation of Knowledge*, pages 299–330. Morgan Kaufmann Publishers, 1990.
- [70] Luka Crnkovic-Dodig. Classifier Showdown. <http://blog.peltarion.com/2006/07/10/classifier-showdown/>, July 2006. Last retrieved August 22, 2012.

- [71] Nick Crofts, Martin Doerr, Tony Gill, Stephen Stead, Matthew Stiff, and CIDOC CRM SIG. Definition of the CIDOC Conceptual Reference Model. http://www.cidoc-crm.org/docs/cidoc_crm_version_5.0.4.pdf, November 2011. Last retrieved August 22, 2012.
- [72] Aron Culotta and Andrew McCallum. Joint deduplication of multiple record types in relational data. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management, CIKM '05*, pages 257–258, New York, NY, USA, 2005. ACM.
- [73] Hamish Cunningham, Diana Maynard, and Kalina Bontcheva. *Text Processing with GATE*. University of Sheffield, Department of Computer Science, April 2011.
- [74] Richard Cyganiak and Anja Jentzsch. The Linking Open Data cloud diagram. <http://richard.cyganiak.de/2007/10/lod/>, September 2011. Last retrieved August 22, 2012.
- [75] DAI. Herzlich Willkommen! Deutsches Archäologisches Institut. <http://www.dainst.org/>, September 2008. Last retrieved August 22, 2012.
- [76] Mathieu d’Aquin, Marta Sabou, Martin Dzbor, Claudio Baldassarre, Laurian Gridinoc, Sofia Angeletou, and Enrico Motta. Watson: a gateway for the Semantic Web. In *The 4th Annual European Semantic Web Conference (ESWC 2007)*, June 2007.
- [77] Tamraparni Dasu and Theodore Johnson. *Exploratory Data Mining and Data Cleaning*. Wiley-Interscience, Hoboken, NJ, 2003.
- [78] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys (CSUR)*, 40:5:1–5:60, May 2008.
- [79] Jérôme David, Jérôme Euzenat, François Scharffe, and Cássia Trojahn dos Santos. The Alignment API 4.0. *Semantic Web*, 2(1):3–10, 2011.
- [80] Maurice de Kunder. The size of the World Wide Web (The Internet). <http://www.worldwidewebsite.com/>, August 2012. Last retrieved August 7, 2012.
- [81] Herbert Van de Sompel, Michael L. Nelson, Carl Lagoze, and Simeon Warner. Resource harvesting within the OAI-PMH framework. *D-Lib Magazine*, 10(12), 2004.

- [82] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified data processing on large clusters. In *Sixth Symposium on Operating System Design and Implementation*, OSDI '04, pages 137–150, December 2004.
- [83] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science (JASIS)*, 41(6):391–407, 1990.
- [84] Martin Doerr and A. Kritsotaki. Documenting events in metadata. In M. Ioannides, D. Arnold, F. Niccolucci, and K. Mania, editors, *The 7th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST)*. ICS-FORTH, Heraklion, Greece, 2006.
- [85] Xin Dong, Alon Halevy, and Jayant Madhavan. Reference reconciliation in complex information spaces. In *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*, SIGMOD '05, pages 85–96, New York, NY, USA, 2005. ACM.
- [86] Martin Dörr. The CIDOC CRM – an ontological approach to semantic interoperability of metadata. *AI Magazine*, 24(3):75–92, 2003.
- [87] Martin Dörr. Who we are. http://www.cidoc-crm.org/who_we_are.html, 2006. Last retrieved August 22, 2012.
- [88] Martin Dörr. Special Interest Group Members. http://www.cidoc-crm.org/special_interest_members.html, 2010. Last retrieved August 22, 2012.
- [89] Martin Dörr and Dolores Iorizzo. The dream of a global knowledge network - a new approach. *Journal on Computing and Cultural Heritage (JOCCH)*, 1(1):1–23, 2008.
- [90] Martin Dörr, Klaus Schaller, and Maria Theodoridou. Integration of complementary archaeological sources. In *Computer Applications and Quantitative Methods in Archaeology Conference (CAA 2004)*. Prato (Italy), April 2004.
- [91] John D. Dougherty, Susan H. Rodger, Sue Fitzgerald, and Mark Guzdial, editors. *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education, SIGCSE 2008, Portland, OR, USA, March 12-15, 2008*. ACM, 2008.

- [92] Halbert L. Dunn. Record linkage. *American Journal of Public Health*, 36(1):1412–1416, January 1946.
- [93] John W. Eaton, David Bateman, and Søren Hauberg. GNU Octave. <https://www.gnu.org/software/octave/octave.pdf>, February 2011. Last retrieved August 22, 2012.
- [94] Øyvind Eide. The Unit for Digital Documentation (EDD) system for storing coref information. <http://cidoc.mediahost.org/eddSystemCoref.pdf>, September 2008. Last retrieved August 22, 2012.
- [95] Øyvind Eide. What is co-reference? http://cidoc.mediahost.org/what_is_coref.pdf, 2008. Last retrieved August 22, 2012.
- [96] Øyvind Eide and Christian-Emil Ore. SIG:Ontologies. <http://wiki.tei-c.org/index.php/SIG:Ontologies>, November 2009. Last retrieved August 22, 2012.
- [97] Yasser EL-Manzalawy and Vasant Honavar. WLSVM: Integrating LibSVM into Weka environment, 2005. Software available at <http://www.cs.iastate.edu/~yasser/wlsvm>. Last retrieved August 22, 2012.
- [98] Ahmed K. Elmagarmid, Panagiotis G. Ipeirotis, and Vassilios S. Verykios. Duplicate record detection: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 19(1):1–16, 2007.
- [99] Douglas Engelbart. Augmenting Human Intellect: A conceptual framework. Summary report AFOSR-3233, Stanford Research Institute, Menlo Park, CA, October 1962.
- [100] Europeana. Thoughtlab : Think with us. <http://www.europeana.eu/portal/thoughtlab.html>. Last retrieved August 22, 2012.
- [101] Europeana. Europeana - Homepage. <http://europeana.eu/portal/>, April 2012. Last retrieved August 22, 2012.
- [102] Christos Faloutsos. *Searching Multimedia Databases by Content*. The Kluwer International Series on Advances in Database Systems; 3. Kluwer, Boston, 5th edition, 2002.

- [103] Bernard Favre-Bulle. *Information und Zusammenhang: Informationsfluß in Prozessen der Wahrnehmung, des Denkens und der Kommunikation*. Springer, Wien, 1st edition, April 2001.
- [104] Ivan P. Fellegi and Alan B. Sunter. A theory for record linkage. *Journal of the American Statistical Association*, 64(328):1183–1210, 1969.
- [105] Dieter Fensel. *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce*. Springer, New York, 2003.
- [106] Anthony G. Flew. *A Dictionary of Philosophy*. St. Martin's Press, New York, NY, USA, 2nd edition, 1984.
- [107] Forschungsgesellschaft Wiener Stadtarchäologie. VBI ERAT LVPA. <http://www.ubi-erat-lupa.org/>, November 2010. Last retrieved August 22, 2012.
- [108] Reinhard Förtsch. Arachne. Objektdatenbank und kulturelle Archive des Archäologischen Instituts der Universität zu Köln und des Deutschen Archäologischen Instituts. <http://arachne.uni-koeln.de/drupal/?q=de/node/186>, August 2007. Last retrieved August 14, 2012.
- [109] Reinhard Förtsch. Arbeitsstelle für Digitale Archäologie - Cologne Digital Archaeology Laboratory (ehem. Forschungsarchiv für Antike Plastik). <http://www.klassarchaeologie.uni-koeln.de/abteilungen/mar/forber.htm>, April 2012. Last retrieved August 22, 2012.
- [110] Gottlob Frege. *Begriffsschrift, eine der Arithmetischen nachgebildete Formelsprache des reinen Denkens*. Nebert, Halle, 1879.
- [111] Gottlob Frege. Über Sinn und Bedeutung. *Zeitschrift für Philosophie und philosophische Kritik*, 1892.
- [112] George W. Furnas, Thomas K. Landauer, Louis M. Gomez, and Susan T. Dumais. The vocabulary problem in human-system communication. *Commun. ACM*, 30(11):964–971, 1987.
- [113] Peter Gärdenfors. How to make the Semantic Web more semantic. In Achille C. Varzi and Laure Vieu, editors, *Formal Ontology in Information Systems: Proceedings of the Third International Conference (FOIS-2004)*, volume 114 of *Frontiers in Artificial Intelligence and Applications*, pages 17–34. IOS Press, 2004.

- [114] Robert Gentleman and Ross Ihaka. The R Project for Statistical Computing. <http://www.r-project.org/>, June 2012. Last retrieved August 22, 2012.
- [115] Lise Getoor and Christopher P. Diehl. Link mining: a survey. *ACM SIGKDD Explorations Newsletter*, 7(2):3–12, 2005.
- [116] Peter Gietz, Andreas Aschenbrenner, Stefan Budenbender, Fotis Jannidis, Marc W. Kuster, Christoph Ludwig, Wolfgang Pempe, Thorsten Vitt, Werner Wegstein, and Andrea Zielinski. TextGrid and eHumanities. In *E-SCIENCE '06: Proceedings of the Second IEEE International Conference on e-Science and Grid Computing*, pages 133–141, Washington, DC, USA, 2006. IEEE Computer Society.
- [117] Hugh Glaser, Afraz Jaffri, and Ian Millard. Managing co-reference on the semantic web. In *WWW2009 Workshop: Linked Data on the Web (LDOW2009)*, Madrid, Spain, April 2009.
- [118] Hugh Glaser, Tim Lewy, Ian Millard, and Ben Dowling. On coreference and the semantic web. Technical report, School of Electronics and Computer Science, University of Southampton, UK, December 2007.
- [119] Google. Freebase. <http://www.freebase.com/>. Last retrieved August 14, 2012.
- [120] Stefan Gradmann. Interoperability: A key concept for large scale, persistent digital libraries. <http://www.digitalpreservationeurope.eu/publications/briefs/interoperability.pdf>, September 2008. Last retrieved September 17, 2012.
- [121] Luis Gravano, Panagiotis G. Ipeirotis, Hosagrahar V. Jagadish, Nick Koudas, S. Muthukrishnan, Lauri Pietarinen, and Divesh Srivastava. Using q-grams in a DBMS for approximate string processing. *IEEE Data Engineering Bulletin*, 24(4):28–34, 2001.
- [122] Luis Gravano, Panagiotis G. Ipeirotis, Hosagrahar V. Jagadish, Nick Koudas, S. Muthukrishnan, and Divesh Srivastava. Approximate string joins in a database (almost) for free. In Peter M. G. Apers, Paolo Atzeni, Stefano Ceri, Stefano Paraboschi, Kotagiri Ramamohanarao, and Richard T. Snodgrass, editors, *Proceedings of the 27th International Conference on Very Large Data Bases (VLDB)*, pages 491–500. Morgan Kaufmann, 2001.

- [123] Benjamin N. Grosz, Ian Horrocks, Raphael Volz, and Stefan Decker. Description logic programs: Combining logic programs with description logic. In *Proceedings of the 12th International Conference on World Wide Web (WWW '03)*, pages 48–57, 2003.
- [124] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition – Special issue: Current issues in knowledge modeling*, 5(2):199–220, June 1993.
- [125] Heinz-Peter Gumm and Manfred Sommer. *Einführung in die Informatik*. Oldenbourg, München, 8th edition, 2009.
- [126] Marios Hadjieleftheriou, Amit Chandel, Nick Koudas, and Divesh Srivastava. Fast indexes and algorithms for set similarity selection queries. In *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, pages 267–276, Washington, DC, USA, 2008. IEEE Computer Society.
- [127] Monika Hagedorn-Saupe, Carlos Saro, Axel Ermert, and Lütger Landwehr. museumsvoabular.de. <http://museum.zib.de/museumsvokabular/>. Last retrieved August 14, 2012.
- [128] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The WEKA data mining software: An update. In *ACM SIGKDD Explorations Newsletter*, volume 11, pages 10–18, 2009.
- [129] Harry Halpin and Henry S. Thompson. Social meaning on the web: From Wittgenstein to search engines. *IEEE Intelligent Systems*, 24(6):27–31, 2009.
- [130] Jiawei Han and Micheline Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2nd edition, 2005.
- [131] Oktie Hassanzadeh, Fei Chiang, Hyun Chul Lee, and Renée J. Miller. Framework for evaluating clustering algorithms in duplicate detection. *Proceedings of the VLDB Endowment*, 2:1282–1293, August 2009.
- [132] Oktie Hassanzadeh and Renée J. Miller. Creating probabilistic databases from duplicated data. *The International Journal on Very Large Data Bases (VLDB Journal)*, 18(5):1141–1166, 2009.

- [133] Patrick Hayes and Brian McBride. RDF semantics. <http://www.w3.org/TR/rdf-mt/>, February 2004. Last retrieved August 22, 2012.
- [134] Patrick J. Hayes and Harry Halpin. In defense of ambiguity. *International Journal on Semantic Web and Information Systems*, 4(2):1–18, 2008.
- [135] John Hebel, Matthew Fisher, Ryan Blace, and Andrew Perez-Lopez. *Semantic Web Programming*. John Wiley & Sons, Hoboken, NJ, USA, 1st edition, April 2009.
- [136] Mauricio A. Hernández and Salvatore J. Stolfo. The merge/purge problem for large databases. *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data (SIGMOD '95)*, 24:127–138, May 1995.
- [137] Mauricio A. Hernández and Salvatore J. Stolfo. Real-world data is dirty: Data cleansing and the merge/purge problem. *Data Mining and Knowledge Discovery*, 2(1):9–37, January 1998.
- [138] Thomas N. Herzog, Fritz J. Scheuren, and William E. Winkler. *Data quality and record linkage techniques*. Springer, New York, NY, 2007.
- [139] Gerhard Heyer. *Text Mining: Wissensrohstoff Text Konzepte, Algorithmen, Ergebnisse*. W3L-Verlag, Herdecke, 2006.
- [140] Gerald Hiebel, Klais Hanke, and Ingrid Hayek. Methodology for CIDOC CRM based data integration with spatial data. In Francisco Contreras and F. Javier Melero, editors, *Proceedings of the 38th Conference on Computer Applications and Quantitative Methods in Archaeology*, April 2010.
- [141] Pascal Hitzler, Rudolf Sebastian, and Markus Krötzsch. *Foundations of Semantic Web Technologies*. Chapman & Hall/CRC, London, 2009.
- [142] Thomas Hofweber. Logic and Ontology (Stanford Encyclopedia of Philosophy). <http://plato.stanford.edu/entries/logic-ontology/>, October 2004. Last retrieved August 22, 2012.
- [143] Aidan Hogan, Andreas Harth, Jürgen Umbrich, Sheila Kinsella, Axel Polleres, and Stefan Decker. Searching and browsing Linked Data with SWSE: The Semantic Web Search Engine. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(4):365–401, December 2011.

- [144] Georg Hohmann and Martin Scholz. Recommendation for the representation of the primitive value classes of the CRM as data types in RDF/OWL implementations. <http://erlangen-crm.org/docs/crm-values-as-owl-datatypes.pdf>, February 2011. Last retrieved August 22, 2012.
- [145] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin. A practical guide to support vector classification. Technical report, Department of Computer Science, National Taiwan University, July 2003.
- [146] Edmund Husserl. Philosophie der Arithmetik. In *Psychologische und logische Untersuchungen*, volume 1. Pfeffer, Halle, 1891.
- [147] University of Glamorgan Hypermedia Research Unit. CIDOC CRM-EH ontology. <http://hypermedia.research.glam.ac.uk/resources/crm/>. Last retrieved August 22, 2012.
- [148] Getty Institute. The Getty Thesaurus of Geographic Names online. <http://www.getty.edu/vow/TGNSearchPage.jsp>, August 2007. Last retrieved August 14, 2012.
- [149] Leif Isaksen, Kirk Martinez, and Graeme Earl. Archaeology, formality & the CIDOC CRM. In *Interconnected Data Worlds: Workshop on the Implementation of the CIDOC-CRM, Berlin, Germany*, November 2009. Online available at <http://eprints.soton.ac.uk/69707/>. Last retrieved August 23, 2012.
- [150] J. Paul Getty Trust and ARTstor. CDWA lite: Specification for an XML schema for contributing records via the OAI harvesting protocol, July 2006.
- [151] Paul Jaccard. Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bulletin del la Société Vaudoise des Sciences Naturelles*, 37:547–579, 1901.
- [152] David Jao. “metric space” (version 10). Freely available at <http://planetmath.org/encyclopedia/MetricTopology.html>. Last retrieved August 23, 2012.
- [153] Matthew A. Jaro. Advances in record-linkage methodology as applied to matching the 1985 Census of Tampa, Florida. *Journal of the American Statistical Association*, 84(406):pp. 414–420, 1989.
- [154] Karen Spärck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21, 1972.

- [155] David Jurgens and Keith Stevens. The S-Space package: An open source package for word space models. In *Proceedings of the ACL 2010 System Demonstrations (ACLDemos)*, pages 30–35. The Association for Computer Linguistics, 2010. Online available at <http://code.google.com/p/airhead-research/>. Last retrieved August 23, 2012.
- [156] Vipul Kashyap and Amit P. Sheth. Semantic and schematic similarities between database objects: A context-based approach. *The International Journal on Very Large Data Bases (VLDB Journal)*, 5(4):276–304, 1996.
- [157] Michael Kifer and Harold Boley. RIF overview. <http://www.w3.org/TR/rif-overview/>, June 2010. Last retrieved August 23, 2012.
- [158] Aaron Kimball, Sierra Michels-Slettvet, and Christophe Bisciglia. Cluster computing for web-scale data processing. In Dougherty et al. [91], pages 116–120.
- [159] Manu Konchady. *Building Search Applications*. Mustru Publishing, Oakton, VA, USA, 1st edition, 2008.
- [160] Hanna Köpcke and Erhard Rahm. Frameworks for entity matching: A comparison. *Data & Knowledge Engineering*, 69(2):197–210, February 2009.
- [161] Hanna Köpcke, Andreas Thor, and Erhard Rahm. Evaluation of entity resolution approaches on real-world match problems. In *Proceedings of the VLDB Endowment*, volume 3, pages 484–493, September 2010.
- [162] Saul A. Kripke. *Naming and Necessity*. Harvard University Press, Cambridge, MA, USA, 1980.
- [163] Robert Kummer. *Towards semantic interoperability of cultural information systems – making ontologies work*. Master’s thesis, Universität zu Köln, August 2007. Online available at http://old.hki.uni-koeln.de/studium/MA/MA_kummer.pdf. Last retrieved August 29, 2012.
- [164] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML ’01)*, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

- [165] Karl-Heinz Lampe, Klaus Riede, and Martin Doerr. Research between natural and cultural history information: Benefits and IT-requirements for transdisciplinarity. *Journal on Computing and Cultural Heritage (JOCCH)*, 1(1), June 2008.
- [166] Justin Leavesley. Perfect or sloppy - RDF, Shirky and Wittgenstein. <http://www.justinleavesley.com/journal/2005/8/2/perfect-or-sloppy-rdf-shirky-and-wittgenstein.html>, August 2005. Last retrieved August 23, 2012.
- [167] Douglas B. Lenat, R. V. Guha, Karen Pittman, Dexter Pratt, and Mary Shepherd. Cyc: toward programs with common sense. *Communications of the ACM*, 33:30–49, August 1990.
- [168] Ulf Leser and Felix Naumann. *Informationsintegration: Architekturen und Methoden zur Integration verteilter und heterogener Datenquellen*. dpunkt-Verl., Heidelberg, 1st edition, 2007.
- [169] Vladimir I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10:707, 1966.
- [170] Thomas Lukasiewicz and Umberto Straccia. Managing uncertainty and vagueness in description logics for the semantic web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(4):291–308, 2008.
- [171] Lev Manovich. White paper: Cultural analytics: Analysis and visualizations of large cultural data sets. http://softwarestudies.com/cultural_analytics/cultural_analytics_2008.doc, May 2007. With contributions from Noah Wardrip-Fruin. Last retrieved August 23, 2012.
- [172] Lev Manovich. Software studies: Cultural analytics. <http://lab.softwarestudies.com/2008/09/cultural-analytics.html>, March 2012. Last retrieved August 14, 2012.
- [173] Catherine C. Marshall and Frank M. Shipman. Which Semantic Web? In *Proceedings of the Fourteenth ACM Conference on Hypertext and Hypermedia (Hypertext'03)*, pages 57–66, New York, NY, USA, August 2003. ACM.
- [174] Margaret Masterman and Yorick Wilks. *Language, Cohesion and Form*. Studies in Natural Language Processing. Cambridge University Press, Cambridge, December 2005.

- [175] Stefano Mazzocchi, Stephen Garland, and Ryan Lee. SIMILE: practical metadata for the Semantic Web. *XML.com*, January 2005. Online available at <http://www.xml.com/lpt/a/2005/01/26/simile.html>, Last retrieved August 29, 2012.
- [176] Andrew McCallum. Information extraction: Distilling structured data from unstructured text. *Queue - Social Computing*, 3(9):48–57, 2005.
- [177] Andrew McCallum, Kamal Nigam, and Lyle H. Ungar. Efficient clustering of high-dimensional data sets with application to reference matching. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '00)*, pages 169–178, 2000.
- [178] Andrew Kachites McCallum. MALLET: A machine learning for language toolkit. <http://mallet.cs.umass.edu>, 2002. Last retrieved August 23, 2012.
- [179] Deborah L. McGuinness and Frank van Harmelen. OWL Web Ontology Language overview. <http://www.w3.org/TR/owl-features/>, February 2004. Last retrieved August 23, 2012.
- [180] Matthew Michelson and Craig A. Knoblock. Learning blocking schemes for record linkage. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI '06)*, volume 1, pages 440–445, 2006.
- [181] Microsoft. Bing - Entwickler. <http://www.bing.com/developers/>. Last retrieved August 14, 2012.
- [182] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, New York, NY, USA, 1997.
- [183] Alvaro E. Monge and Charles Elkan. The field matching problem: Algorithms and applications. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 267–270, 1996.
- [184] Felix Naumann and Melanie Herschel. *An Introduction to Duplicate Detection*. Synthesis Lectures on Data Management. Morgan & Claypool Publishers, San Rafael, CA, 2010.
- [185] Gonzalo Navarro. A guided tour to approximate string matching. *ACM Computing Surveys (CSUR)*, 33(1):31–88, March 2001.

- [186] Theodor H. Nelson. *Literary Machines: The report on, and of, Project Xanadu concerning word processing, electronic publishing, hypertext, thinkertoys, tomorrow's intellectual revolution, and certain other topics including knowledge, education and freedom*. T. Nelson, Swarthmore, PA, USA, 1981.
- [187] Howard B. Newcombe, James M. Kennedy, S. J. Axford, and A. P. James. Automatic linkage of vital records. *Science*, 130:954–959, October 1959.
- [188] Peter Norvig and Stuart Russell. *Artificial Intelligence: A Modern Approach*. Prentice Hall International, 2 edition, 2003.
- [189] Philipp Nussbaumer and Bernhard Haslhofer. Putting the CIDOC CRM into practice - experiences and challenges. Technical report, University of Vienna, September 2007.
- [190] OCLC. About LC Name Authority File. <http://www.oclc.org/research/researchworks/authority/default.htm>. Last retrieved August 14, 2012.
- [191] Charles K. Ogden and Ivor A. Richards. *The Meaning of Meaning*. Trubner & Co, London, 1923.
- [192] Martin Oischinger, Bernhard Schiemann, and Günther Görz. An implementation of the CIDOC Conceptual Reference Model (4.2.4) in OWL-DL. In *The Annual Conference of the International Documentation Committee of the International Council of Museums*, 2008.
- [193] Ontotext AD. RDF(S), rules, and OWL dialects. <http://www.ontotext.com/rdfs-rules-owl>, 2011. Last retrieved August 23, 2012.
- [194] M. Tamer Özsu and Patrick Valduriez. *Principles of Distributed Database Systems*. Springer, 3rd edition, 2011.
- [195] Charles S. Peirce and Justus Buchler. *Philosophical Writings of Peirce*. Dover Publ., New York, NY, 1955.
- [196] Perseus Digital Library. Perseus Collections/Texts. <http://www.perseus.tufts.edu/hopper/collections>. Last retrieved August 14, 2012.
- [197] Barbara Pfeifer. Gemeinsame Normdatei (GND). http://www.dnb.de/DE/Standardisierung/Normdaten/GND/gnd_node.html. Last retrieved August 14, 2012.

- [198] Barbara Pfeifer. Personennormdatei (PND). http://www.dnb.de/DE/Standardisierung/Normdaten/PND/pnd_node.html. Last retrieved August 14, 2012.
- [199] Felix Pirson. Pergamon – Bericht über die Arbeiten in der Kampagne 2006. *Archäologischer Anzeiger*, pages 30 – 34, 2007.
- [200] Lucian Popa, Yannis Velegarakis, Mauricio A. Hernández, Renée J. Miller, and Ronald Fagin. Translating Web data. In *Proceedings of the 28th International Conference on Very Large Data Bases, VLDB '02*, pages 598–609. VLDB Endowment, 2002.
- [201] ProgrammableWeb.com. ProgrammableWeb – Mashups, APIs, and the Web as platform. <http://www.programmableweb.com/>. Last retrieved August 14, 2012.
- [202] Eric Prud'hommeaux and Andy Seaborne. SPARQL query language for RDF. <http://www.w3.org/TR/rdf-sparql-query/>, January 2008. Last retrieved August 23, 2012.
- [203] Sven Puhmann, Melanie Weis, and Felix Naumann. XML duplicate detection using Sorted Neighborhoods. In Yannis Ioannidis, Marc Scholl, Joachim Schmidt, Florian Matthes, Mike Hatzopoulos, Klemens Boehm, Alfons Kemper, Torsten Grust, and Christian Boehm, editors, *Advances in Database Technology (EDBT 2006)*, volume 3896 of *Lecture Notes in Computer Science*, pages 773–791. Springer Berlin / Heidelberg, 2006.
- [204] J. Ross Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.
- [205] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.
- [206] Erhard Rahm and Hong Hai Do. Data cleaning: Problems and current approaches. *IEEE Data Engineering Bulletin*, 23(4):3–13, March 2000.
- [207] Matthew J. Rattigan and David Jensen. The case for anomalous link discovery. *ACM SIGKDD Explorations Newsletter*, 7(2):41–47, December 2005.
- [208] Valentin Robu, Harry Halpin, and Hana Shepherd. Emergence of consensus and shared vocabularies in collaborative tagging systems. *ACM Transactions on the Web (TWEB)*, 3(4):1–34, September 2009.

- [209] Lior Rokach. Ensemble methods in supervised learning. In Oded Maimon and Lior Rokach, editors, *Data Mining and Knowledge Discovery Handbook*, pages 959–979. Springer, 2010.
- [210] Jay F. Rosenberg. *Philosophieren: ein Handbuch für Anfänger*. Klostermann, Frankfurt am Main, 1986.
- [211] Arnon Rotem-Gal-Oz. Fallacies of distributed computing explained. <http://www.rgoarchitects.com/Files/fallacies.pdf>, 2006. Last retrieved August 23, 2012.
- [212] Bertrand Russell. On denoting. *Mind*, 14(56):479–493, 1905.
- [213] Sunita Sarawagi and Anuradha Bhamidipaty. Interactive deduplication using active learning. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '02)*, pages 269–278, New York, NY, USA, 2002. ACM.
- [214] Warren S. Sarle. Measurement theory: Frequently asked questions. <ftp://ftp.sas.com/pub/neural/measurement.html>, September 1997. Last retrieved August 24, 2012.
- [215] Manfred Schmidt-Schauß and Gert Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26, 1991.
- [216] Hans-Jochen Schneider. *Lexikon Informatik und Datenverarbeitung*. Oldenbourg, München, 4th edition, 1997.
- [217] Nigel Shadbolt, Tim Berners-Lee, and Wendy Hall. The Semantic Web revisited. *IEEE Intelligent Systems*, 21(3):96–101, May 2006.
- [218] Claude E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656, July, October 1948.
- [219] Colin Shearer, Julian Clinton, Pete Chapman, Randy Kerber, Rüdiger Wirth, Thomas Khabaza, and Thomas Reinartz. The CRISP-DM 1.0: Step-by-Step Data Mining Guide. <ftp://ftp.software.ibm.com/software/analytics/spss/support/Modeler/Documentation/14/UserManual/CRISP-DM.pdf>, 2000. Last retrieved August 24, 2012.
- [220] Amit P. Sheth and James A. Larson. Federated database systems for managing distributed, heterogeneous, and autonomous databases. *ACM Computing Surveys (CSUR) - Special Issue on Heterogeneous Databases*, 22(3):183–236, September 1990.

- [221] Amit P. Sheth, Cartic Ramakrishnan, and Christopher Thomas. Semantics for the Semantic Web: The implicit, the formal and the powerful. *International Journal on Semantic Web and Information Systems*, 1(1):1–18, January–March 2005.
- [222] Frank M. Shipman and Catherine C. Marshall. Formality considered harmful: Experiences, emerging themes, and directions on the use of formal representations in interactive systems. *Computer Supported Cooperative Work (CSCW)*, 8(4):333–352, December 1999.
- [223] Clay Shirky. The Semantic Web, syllogism, and worldview. http://www.shirky.com/writings/semantic_syllogism.html, November 2003. Last retrieved August 24, 2012.
- [224] Theodore Sider. *Four-Dimensionalism: An Ontology of Persistence and Time*. Oxford University Press, October 2001.
- [225] Jörn Sieglerschmidt. Knowledge organization and multilingual vocabularies. In *Managing the Global Diversity of Cultural Information*, Vienna, August 2007. Lecture at the annual meeting of the Comité International pour la Documentation (CIDOC).
- [226] Hung sik Kim and Dongwon Lee. Parallel linkage. In Mário J. Silva, Alberto H. F. Laender, Ricardo A. Baeza-Yates, Deborah L. McGuinness, Bjørn Olstad, Øystein Haug Olsen, and André O. Falcão, editors, *Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management (CIKM '07)*, pages 283–292. ACM, 2007.
- [227] Simeon J. Simoff, Michael H. Böhlen, and Arturas Mazeika. Visual data mining: An introduction and overview. In Simeon J. Simoff, Michael H. Böhlen, and Arturas Mazeika, editors, *Visual Data Mining*, pages 1–12. Springer-Verlag, Berlin, Heidelberg, 2008.
- [228] Philipp Simon. MySQL UDFs. http://www.codeguru.com/cpp/data/mfc_database/misc/article.php/c12615/MySQL-UDFs.htm, September 2006. Last retrieved August 14, 2012.
- [229] Parag Singla and Pedro Domingos. Multi-relational record linkage. In *Proceedings of the KDD-2004 Workshop on Multi-Relational Data Mining*, pages 31–48, August 2004.
- [230] Karl Sintenis, editor. *Plutarchi Vitae parallelae*. Teubner, Leipzig, Stuttgart, 3rd edition, 1960.

- [231] Steven Skiena. The Stony Brook Algorithm Repository. <http://www.cs.sunysb.edu/~algorithm/files/graph-data-structures.shtml>, July 2008. Last retrieved August 14, 2012.
- [232] Steven S. Skiena. *The Algorithm Design Manual*. Springer, London, 2nd edition, 2010.
- [233] Temple F. Smith and Michael S. Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147(1):195–197, March 1981.
- [234] Robert Spaemann. Ähnlichkeit. *Zeitschrift für philosophische Forschung*, 50(1/2):pp. 286–290, 1996.
- [235] Steffen Staab and York Sure-Vetter. The 10th International Semantic Web Conference: Home. <http://iswc2011.semanticweb.org/>, 2011. Last retrieved August 24, 2012.
- [236] Regine Stein, Axel Ermert, Jürgen Gottschewski, Monika Hagedorn-Saupe, Regine Heuchert, Hans-Jürgen Hansen, Angela Kailus, Carlos Saro, Regine Scheffel, Gisela Schulte-Dornberg, Jörn Sieglerschmidt, and Axel Vitzthum. museumdat – harvesting format for providing core data from museum holdings. Technical report, FG Dokumentation im Deutschen Museumsbund / Institut für Museumsforschung SMB-PK /Zuse-Institut, Berlin, September 2007.
- [237] Stanley S. Stevens. On the theory of scales of measurement. *Science*, 103:677–680, 1946.
- [238] Wolfgang G. Stock. *Information Retrieval: Informationen suchen und finden*. Oldenbourg, 2007.
- [239] Giorgos Stoilos, Giorgos B. Stamou, Vassilis Tzouvaras, Jeff Z. Pan, and Ian Horrocks. Fuzzy OWL: Uncertainty and the semantic web. In Bernardo Cuenca Grau, Ian Horrocks, Bijan Parsia, and Peter F. Patel-Schneider, editors, *Proceedings of the Workshop on OWL: Experiences and Directions (OWLED '05)*, volume 188 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2005.
- [240] TEI. TEI: Text Encoding Initiative. <http://www.tei-c.org/index.xml>, 2009. Last retrieved August 24, 2012.
- [241] Sheila Tejada, Craig A. Knoblock, and Steven Minton. Learning object identification rules for information integration. *Information Systems - Data Extraction, Cleaning and Reconciliation*, 26:607–633, December 2001.

- [242] The Apache Software Foundation. Apache Mahout: Scalable machine learning and data mining. <http://mahout.apache.org/>, 2011. Last retrieved August 24, 2012.
- [243] The Apache Software Foundation. Welcome to Apache Hadoop! <http://hadoop.apache.org/>, December 2011. Last retrieved August 24, 2012.
- [244] The Apache Software Foundation. Apache solr. <https://lucene.apache.org/solr/>, July 2013.
- [245] The Getty Research Institute. Art & Architecture Thesaurus Online. <http://www.getty.edu/research/tools/vocabularies/aat/>. Last accessed August 14, 2012.
- [246] Andreas Thor and Erhard Rahm. MOMA - a mapping-based object matching system. In *Proceedings of the Third Biennial Conference on Innovative Data Systems Research*, pages 247–258. www.crdrrdb.org, 2007.
- [247] University of Glamorgan. Semantic Technologies for Archaeological Resources, Hypermedia Research Unit, University of Glamorgan. <http://hypermedia.research.glam.ac.uk/kos/star/>, 2010. Last retrieved August 24, 2012.
- [248] W3C. W3C Semantic Web activity. <http://www.w3.org/2001/sw/>, 2012. Last retrieved August 7th, 2012.
- [249] W3C. World Wide Web Consortium (W3C). <http://www.w3.org/>, 2012. Last retrieved August 24, 2012.
- [250] W3C OWL Working Group. OWL 2 Web Ontology Language document overview. <http://www.w3.org/TR/owl-overview/>, October 2009. Last retrieved August 24, 2012.
- [251] Richard Y. Wang and Diane M. Strong. Beyond accuracy: what data quality means to data consumers. *Journal of Management Information Systems*, 12(4):5–33, March 1996.
- [252] Melanie Weis and Felix Naumann. DogmatiX tracks down duplicates in XML. In *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*, SIGMOD '05, pages 431–442, New York, NY, USA, 2005. ACM.

- [253] WEKA. Attribute-Relation File Format (ARFF). <http://weka.wikispaces.com/ARFF>, August 2009. Last retrieved August 24, 2012.
- [254] Ben Wellner, Andrew McCallum, Fuchun Peng, and Michael Hay. An integrated, conditional model of information extraction and coreference with application to citation matching. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, UAI '04*, pages 593–601, Arlington, Virginia, United States, 2004. AUAI Press.
- [255] Marc Wick and the GeoNames Community. GeoNames. <http://www.geonames.org/>, 2012. Last retrieved August 14, 2012.
- [256] Dominic Widdows and Kathleen Ferraro. Semantic Vectors: a scalable open source package and online technology management application. In Dougherty et al. [91].
- [257] Karl M. Wigg. Knowledge vs Information. <http://www.km-forum.org/t000008.htm>, February–March 1996. Last retrieved August 31, 2012.
- [258] William E. Winkler. String comparator metrics and enhanced decision rules in the Fellegi-Sunter model of record linkage. In *Proceedings of the Section on Survey Research*, pages 354–359, 1990.
- [259] William E. Winkler. Methods for record linkage and Bayesian Networks. Technical report, Series RRS2002/05, U.S. Bureau of the Census, 2002.
- [260] WissKI. WissKI. Scientific Communication Infrastructure. A project funded by the German Research Foundation. <http://www.wiss-ki.eu/>. Last retrieved August 14, 2012.
- [261] Ian H. Witten and Eibe Frank. *Data Mining. Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2nd ed. edition, July 2005.
- [262] Ludwig Wittgenstein. *Tractatus logico-philosophicus*. *Logisch-philosophische Abhandlung*. Suhrkamp-Taschenbuch Wissenschaft, Teil 501. Suhrkamp, 4th edition, 1963.
- [263] Su Yan, Dongwon Lee, Min-Yen Kan, and C. Lee Giles. Adaptive sorted neighborhood methods for efficient record linkage. In *Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '07*, pages 185–194, 2007.

- [264] Morris Zelditch Jr. Intelligible comparisons. *Comparative Methods in Sociology: Essays on Trends and Applications*, pages 267–307, 1974.
- [265] Duo Zhang, Qiaozhu Mei, and ChengXiang Zhai. Cross-lingual latent topic extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1128–1137, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.
- [266] Huimin Zhao. Semantic matching across heterogeneous data sources. *Communications of the ACM - The patent holder's dilemma: buy, sell, or troll?*, 50(1):45–50, January 2007.

List of Figures

1	The Perseus reading environment displaying a text passage by Strabo that deals with the expansion of Pergamum (Screenshot taken from The Perseus Digital Library).	24
2	The Arachne single record view displaying topographic information about the city wall of Pergamum (Screenshot taken from Arachne).	25
3	The Arachne context browser displaying contextual information about the city wall of Pergamum (Screenshot taken from Arachne).	25
4	Documentation of a trenching at the city wall of Pergamon (Image taken from iDAI.field).	26
5	Image of a trenching at the city wall of Pergamon (Copyright by DAI Istanbul, Pergamon excavations).	27
6	The semantic triangle.	43
7	A classification of data quality problems in and among data sources.	46
8	Architecture classification of integrated information systems.	51
9	Five-level-architecture of a federated information system.	51
10	The Semantic Web “layercake” diagram.	58
11	Complexity and relation to Rules, LP and DL of different OWL dialects.	63
12	The Linked Data Cloud.	70
13	The General Structure of the CIDOC CRM.	81
14	The (reduced) scenario as linked information objects.	83
15	Information extraction and knowledge discovery workflow.	91
16	Determining the similarity of documents in a vector space model.	94
17	Classification of an example day by a decision tree. The learned concept is whether tennis should be played on days with specific features.	103
18	An example for Canopy clustering.	109

19	A matrix for calculating the Levenshtein distance.	121
20	A probability tree showing the example experiment.	126
21	Approaches to coreference resolution.	131
22	Architecture of an entity matching framework.	140
23	Probabilistic approaches in the taxonomy of relational ER models. .	147
24	The relation of precision and recall to efficiency.	155
25	Frequency plots of locations that are attributed to objects in Arachne and Perseus.	183
26	Frequency plots of collection and depository locations that are at- tributed to objects in Arachne and Perseus.	186
27	Frequency plot of findspots that are attributed to objects in Arachne and Perseus.	188
28	Histograms for the distribution of measured heights up to 3000 millimeters for material objects described by Arachne and Perseus. .	193
29	The frequency of tokens in the short descriptions of the archaeolog- ical objects in Perseus.	196
30	A rudimentary user interface for verifying the decision of the ma- chine learning model.	208
31	By using the links to the original information sources, entities can be evaluated in their “original” digital context.	209
32	A browser relying on CIDOC CRM data.	217

List of Tables

1	Entailment rule <i>rdfs11</i> that covers the subclass relationship in RDFS. .	64
2	An example for a term-document co-occurrence matrix.	96
3	A contingency table to evaluate the success of a classifier for entity resolution.	153
4	Estimated number of extractable entities for Arachne and Perseus. .	172
5	Extraction ratio for Arachne and Perseus.	177
6	Extraction ratios for sample pairs of entity descriptions of Arachne and Perseus.	177
7	Quality of the training sample (632 pairs).	178

List of Algorithms

1	Abbreviated pseudocode for the ID3 algorithm.	104
2	Abbreviated pseudocode for a canopy clusterer.	108

Listings

1	An SQL command that finds records that a links by their depository.	174
2	The extracted information expressed about bibliographic references serialized as Turtle.	183
3	The extracted information expressed about collection and depository locations serialized as Turtle.	186
4	The extracted information expressed about findspots serialized as Turtle.	189
5	The extracted information expressed about accession numbers serialized as Turtle.	190
6	The extracted information about the height of a material object serialized as Turtle.	193
7	The extracted information from the short description of a material object serialized as Turtle.	196
8	An excerpt of the dumped clustering information as CSV file. . . .	198
9	An excerpt of the log file that records comparison results for the short descriptions in Arachne and Perseus.	200
10	An excerpt of the log file that records results of the application of LSA.	201
11	Determined similarities of pairs in the sample represented as an ARFF file.	202
12	The decision tree model after performing the training.	205