2. Knowledge Discovery for Ontology Construction

Marko Grobelnik, Dunja Mladenić
Jozef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia

2.1. Introduction

We can observe that the focus of modern information systems is moving from "data-processing" towards "concept-processing", meaning that the basic unit of processing is less and less an atomic piece of data and is becoming more a semantic concept which caries an interpretation and exists in a context with other concepts. As mentioned in the previous chapter, an ontology is a structure capturing semantic knowledge about a certain domain by describing relevant concepts and relations between them.

Knowledge Discovery (KD) is a research area developing techniques that enable computers to discover novel and interesting information from raw data. Usually the initial output from KD is further refined via an iterative process with a human in the loop in order to get knowledge out of the data. With the development of methods for semi-automatic processing of complex data it is becoming possible to extract hidden and useful pieces of knowledge which can be further used for different purpose including semi-automatic ontology construction. As ontologies are taking a significant role in the Semantic Web, we address the problem of semi-automatic ontology construction supported by Knowledge Discovery. This chapter presents several approaches from Knowledge Discovery that we envision as useful for the Semantic Web and in particular for semi-automatic ontology construction. In that light, we propose to decompose the semi-automatic ontology construction process into several phases. Several scenarios of the ontology learning phase are identified based on different assumptions regarding the provided input data. We outline some ideas how the defined scenarios can be addressed by different Knowledge Discovery approaches.

The rest of this Chapter is structured as follows. Section 2.2 provides a brief description of Knowledge Discovery. Section 2.3 gives a definition of the term ontology. Section 2.4 describes the problem of semi-automatic ontology construction. Section 2.5 describes the proposed methodology for semi-automatic ontology construction where the whole process is decomposed into several phases. Section 2.6 describes several Knowledge Discovery methods in the context of the semi-automatic ontology construction phases defined in Section 2.5. Section 2.7 gives a brief overview of the existing work in the area of semi-automatic ontology construction. Section 2.8 concludes the Chapter with discussion.

2.2 Knowledge Discovery

The main goal of Knowledge Discovery is to find useful pieces of knowledge within the data with little or no human involvement. There are several definitions of Knowledge

Discovery and here we cite just one of them: Knowledge Discovery is a process which aims at the extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) information from data in large databases (Fayad et al., 1996).

In Knowledge Discovery there has been recently an increased interest for learning and discovery in unstructured and semi-structured domains such as text (Text Mining), web (Web Mining), graphs/networks (Link Analysis), learning models in relational/first-order form (Relational Data Mining), analyzing data streams (Stream Mining), etc. In these we see a great potential for addressing the task of semi-automatic ontology construction.

Knowledge Discovery can be seen as a research area closely connected to the following research areas: *Computational Learning Theory* with a focus on mainly theoretical questions about learnability, computability, design and analysis of learning algorithms; *Machine Learning* (Mitchell, 1997), where the main questions are how to perform automated learning on different kinds of data and especially with different representation languages for representing learned concepts; *Data-Mining (*Fayyad et al., 1996; Witten and Frank, 1999; Hand et al., 2001), being rather applied area with the main questions on how to use learning techniques on large-scale real-life data; *Statistics* and statistical learning (Hastie et al., 2001) contributing techniques for data analysis (Duda et al., 2000) in general.

2.3. Ontology Definition

Ontologies are used for organizing knowledge in a structured way in many areas – from philosophy to Knowledge Management and the Semantic Web. We usually refer to an ontology as a graph/network structure consisting from:

1. a set of concepts (vertices in a graph)
2. a set of relationships connecting concepts (directed edges in a graph)
3. a set of instances assigned to a particular concepts (data records assigned to concepts or relation)

More formally, an ontology is defined (Ehrig et al., 2005) as a structure $O = (C, T, R, A, I, V, \leq_C, \leq_T, \sigma_R, \sigma_A, \iota_C, \iota_T, \iota_R, \iota_A)$. It consists of disjoint sets of concepts (C), types (T), relations (R), attributes (A), instances (I) and values (V). The partial orders $\leq_C$ (on C) and $\leq_T$ (on T) define a concept hierarchy and a type hierarchy respectively. The function $\sigma_R: R \rightarrow C^2$ provides relation signatures (i.e. for each relation, the function specifies which concepts may be linked by this relation), while $\sigma_A: A \rightarrow C \times T$ provides attribute signatures (for each attribute, the function specifies to which concept the attribute belongs and what is its datatype). Finally, there are partial instantiation functions $\iota_C: C \rightarrow 2^I$ (the assignment of instances to concepts), $\iota_T: T \rightarrow 2^V$ (the assignment of values to types), $\iota_R: R \rightarrow 2^{I \times I}$ (which instances are related by a particular relation), and $\iota_A: A \rightarrow 2^{I \times V}$ (what is the value of each attribute for each instance). Another formalization of ontologies, based on similar principles, has been described by (Bloehdorn et al., 2005). Notice that this theoretical framework can be used to define evaluation of ontologies as a function that maps the ontology O to a real number (Brank et al, 2005).

2.4. Methodology for Semi-automatic Ontology Construction

Knowledge Discovery technologies can be used to support different phases and scenarios of semi-automatic ontology construction. We believe that today a completely automatic construction of good quality ontologies is in general not possible for theoretical, as well as practical reasons (e.g. the soft nature of the knowledge being conceptualized). As in Knowledge Discovery in general, human interventions are necessary but costly in terms of resources. Therefore the technology should help in efficient utilization of human interventions, providing suggestions, highlighting potentially interesting information and enabling refinements of the constructed ontology.

There are several definitions of the ontology engineering and construction methodology, mainly based on a knowledge management perspective. For instance, the DILIGENT ontology engineering methodology described in Chapter 9 defines five main steps of ontology engineering: building, local adaptation, analysis, revision, and local update. Here, we define a methodology for *semi-automatic ontology construction* analogous to the CRISP-DM methodology (Chapman et al., 2000) defined for the Knowledge Discovery process. CRISP-DM involves six interrelated phases: business understanding, data understanding, data preparation, modeling, evaluation, and deployment. From the perspective of Knowledge Discovery, semi-automatic ontology construction can be defined as consisting of the following interrelated phases:

1. *domain understanding* (what is the area we are dealing with?),
2. *data understanding* (what is the available data and its relation to semi-automatic ontology construction?),
3. *task definition* (based on the available data and its properties, define task(s) to be addressed),
4. *ontology learning* (semi-automated process addressing the task(s) defined in the phase 3),
5. *ontology evaluation* (estimate quality of the solutions to the addressed task(s)), and
6. *refinement with human in the loop* (perform any transformation needed to improve the ontology and return to any of the previous steps, as desired).

The first three phases require intensive involvement of the user and are prerequisites for the next three phases. While phases 4 and 5 can be automated to some extent, the last phase heavily relays on the user. Section 2.5 describes the fourth phase and some scenarios related to addressing the ontology learning problem by Knowledge Discovery methods. Using Knowledge Discovery in the fifth phase for semi-automatic ontology evaluation is not in the scope of this Chapter, an overview can be found in (Brank et al, 2005).

2.5. Ontology Learning Scenarios

From a Knowledge Discovery perspective, we see an ontology as just another class of models (somewhat more complex compared to typical Machine Learning models) which needs to be expressed in some kind of hypothesis language. Depending on the different assumptions regarding the provided input data, ontology learning can be addressed via different tasks: learning just the ontology concepts, learning just the ontology relationships between the existing concepts, learning both the concepts and relations at

the same time, populating an existing ontology/structure, dealing with dynamic data streams, simultaneous construction of ontologies giving different views on the same data, etc.. More formally, we define the ontology learning tasks in terms of mappings between ontology components, where some of the components are given and some are missing and we want to induce the missing ones. Some typical **scenarios in ontology learning** are the following:

1. Inducing concepts/clustering of instances (given instances)
2. Inducing relations (given concepts and the associated instances)
3. Ontology population (given an ontology and relevant, but not associated instances)
4. Ontology generation (given instances and any other background information)
5. Ontology updating/extending (given an ontology and background information, such as, new instances or the ontology usage patterns)

Knowledge discovery methods can be used in all of the above typical scenarios of ontology learning. When performing the learning using Knowledge Discovery, we need to select a language for representation of a membership function. Examples of different representation languages as used by machine learning algorithms are: Linear functions (eg., used by Support-Vector-Machines), Propositional logic (eg., used in decision trees and decision rules), First order logic (eg., used in Inductive Logic programming). The representation language selected informs the expressive power of the descriptions and complexity of computation.

2.6. Using Knowledge Discovery for Ontology Learning

Knowledge Discovery techniques are in general aiming at discovering knowledge and that is often achieved by finding some structure in the data. This means that we can use these techniques to map unstructured data-sources, such as a collection of text documents, into an ontological structure. Several techniques that we find relevant for ontology learning have been developed in Knowledge Discovery, some of them in combination with related fields such as Information Retrieval (van Rijsbergen, 1979) and Language Technologies (Manning and Schutze, 2001). Actually, Knowledge Discovery techniques are well integrated in many aspects of Language Technologies combining human background knowledge about the language with automatic approaches for modeling the “soft” nature of ill structured data formulated in natural language. More on the usage of Language Technologies in knowledge management can be found in (Cunningham and Bontcheva, 2005).

It is also important to point out that scalability is one of the central issues in Knowledge Discovery, where one needs to be able to deal with real-life dataset volumes of the order of terabytes. Ontology construction is ultimately concerned with real-life data and on the Web today we talk about tens of billions of Web pages indexed by major search engines. Because of the exponential growth of data available in electronic form, especially on the Web, approaches where a large amount of human intervention is necessary, become inapplicable. Here we see a great potential for Knowledge Discovery with its focus on scalability.

The following subsections briefly describe some of the Knowledge Discovery techniques that can be used for addressing the ontology learning scenarios described in Section 2.5.

2.6.1 Unsupervised learning

In the broader context, the Knowledge Discovery approach to ontology learning deals with some kind of data objects which need to have some kind of properties – these may be text documents, images, data records or some combination of them. From the perspective of using Knowledge Discovery methods for inducing concepts given the instances (ontology learning scenario 1 in Section 2.5), the important part is comparing ontological instances to each other. As document databases are the most common data type conceptualized in the form of ontologies, we can use methods developed in Information Retrieval and Text Mining research, for estimating similarity between documents as well as similarity between objects used within the documents (e.g., named entities, words, etc.) – these similarity measures can be used together with unsupervised learning algorithms, such as clustering algorithms, in an approach to forming an approximation of ontologies from document collections.

An approach to semi-automatic topic ontology construction from a collection of documents (ontology learning scenario 4 in Section 2.5) is proposed in (Fortuna et al., 2005a). Ontology construction is seen as a process where the user is constructing the ontology and taking all the decisions while the computer provides suggestions for the topics (ontology concepts), and assists by automatically assigning documents to the topics, naming the topics, etc. The system is designed to take a set of documents and provide suggestions of possible ontology concepts (topics) and relations (sub-topic-of) based on the text of documents. The user can use the suggestions for concepts and their names, further split or refine the concepts, move a concept to another place in the ontology, explore instances of the concepts (in this case documents), etc. The system supports also extreme case where the user can ignore suggestions and manually construct the ontology. All this functionality is available through an interactive GUI-based environment providing ontology visualization and the ability to save the final ontology as RDF. There are two main methodological contributions introduced in this approach: (i) suggesting concepts as subsets of documents and (ii) suggesting naming of the concepts. Suggesting concepts based on the document collection is based on representing documents as word-vectors and applying *Document clustering* or *Latent Semantic Indexing(LSI)*. As ontology learning scenario 4 (described in Section 2.5) is one of the most important and demanding, in the remaining of this subsection we briefly describe both methods (clustering and LSI) for suggesting concepts. Turning to the second approach, naming of the concepts is based on proposing labels comprised of the most common keywords (describing a subset of documents belonging to the topic), and alternatively on providing the most discriminative keywords (enabling classification of documents into the topic relative to the neighboring topics). Methods for document classification are briefly described in Section 2.6.2.

Document clustering (Steinbach et al., 2000) is based on a general data clustering algorithm adopted for textual data by representing each document as a word-vector, which for each word contains some weight proportional to the number of occurrences of the word (usually TFIDF weight as given in equation (2.1)).

$$d^{(i)} = TF(W_i, d)IDF(W_i), where IDF(W_i) = \log \frac{D}{DF(W_i)} \quad (2.1)$$

where D is the number of documents; document frequency DF(W) is the number of documents the word W occurred in at least once; and TF(W,d) is the number of times word W occurred in document d. The exact formula used in different approaches may vary somewhat but the basic idea remains the same – namely, that the weighting is a measure of how frequently the given word occurs in the document at hand and of how common (or otherwise) the word is in an entire document collection.

The similarity of two documents is commonly measured by the cosine-similarity between the word-vector representations of the documents (see equation (2.2)). The clustering algorithm groups documents based on their similarity, putting similar documents in the same group. Cosine-similarity is commonly used also by some supervised learning algorithms for document categorization, which can be useful in populating topic ontologies (ontology learning scenario 3 in Section 2.5). Given a new document, cosine-similarity is used to find the most similar documents (e.g., using *k*-Nearest Neighbor algorithm (Mitchell, 1997)). Cosine-similarity between all the documents and the new document is used to find the *k* most similar documents whose categories (topics) are then used to assign categories to a new document. For documents d_i and dj, the similarity is calculated as given in equation (2.2). Note that the cosine similarity between two identical documents is 1 and between two documents that share no words is zero.

$$\cos(d_i, d_j) = \frac{\sum_k d_{ik} d_{jk}}{\sqrt{\sum_l d_{il}^2 \sum_m d_{jm}^2}} \quad (2.2)$$

Latent Semantic Indexing is a linear dimensionality reduction technique based on a technique from linear algebra called Singular Value Decomposition. It uses a word-vector representation of text documents for extracting words with similar meanings (Deerwester, 2001). It relies on the fact that two words related to the same topic more often co-occur together than words describing different topics. This can also be viewed as extraction of hidden semantic concepts or topics from text documents. The result of applying Latent Semantic Indexing on a document collection are fuzzy clusters of words each describing topics.

More precisely, in the process of extracting the hidden concepts first a term-document matrix *A* is constructed from a given set of text documents. This is a matrix having word-vectors of documents as columns. This matrix is decomposed using singular value decomposition so that $A = USV^T$, where matrices *U* and *V* are orthogonal and *S* is a diagonal matrix with ordered singular values on the diagonal. Columns of the matrix *U* form an orthogonal basis of a subspace of the original space where vectors with higher singular values carry more information (by truncating singular values to only the *k* biggest values, we get the best approximation of matrix *A* with rank *k*). Because of this, vectors that form this basis can also be viewed as concepts or topics. Geometrically each basis vector splits the original space into two halves. By taking just the words with the

highest positive or the highest negative weight in this basis vector, we get a set of words which best describe a concept generated by this vector. Note that each vector can generate two concepts; one is generated by positive weights and one by negative weights.

2.6.2 Semi-supervised, supervised and active learning

Often it is too hard or too costly to integrate available background domain knowledge into fully automatic techniques. *Active Learning* and *Semi-supervised Learning* make use of small pieces of human knowledge for better guidance towards the desired model (e.g., an ontology). The effect is that we are able to reduce the amount of human effort by an order of magnitude while preserving the quality of results (Blum and Chawla, 2001). The main task of both methods is to attach labels to unlabeled data (such as content categories to documents) by maximizing the quality of the label assignment and by minimizing the effort (human or computational).

A typical example scenario for using semi-supervised and active learning methods would be assigning content categories to uncategorized documents from a large document collection (e.g., from the Web or from a news source) as described in (Novak, 2004a). Typically, it is too costly to label each document manually – but there is some limited amount of human resource available. The task of active learning is to use the (limited) available user effort in the most efficient way, to assign high quality labels (e.g., in the form of content categories) to documents; semi-supervised learning, on the other hand, is applied when there are some initially labeled instances (e.g., documents with assigned topic categories) but no additional human resources are available. Finally, *supervised learning* is used when there is enough labeled data provided in advance and no additional human resources are available. All the three methods can be useful in populating ontologies (ontology learning scenario 3 in Section 2.5) using document categorization as well as in more sophisticated tasks such as inducing relations (ontology learning scenario 2 in Section 2.5), ontology generation and extension (ontology learning scenario 4 and 5 in Section 2.5).

Supervised learning for text *document categorization* can be applied when a set of predefined topic categories, such as “arts, education, science”, are provided as well as a set of documents labeled with those categories. The task is to classify new (previously unseen) documents by assigning each document one or more content categories (e.g., ontology concepts or relations). This is usually performed by representing documents as word-vectors and using documents that have already been assigned to the categories, to generate a model for assigning content categories to new documents (Jackson and Moulinier, 2002; Sebastiani, 2002). In the word-vector representation of a document, a vector of word frequencies is formed taking all the words occurring in all the documents (usually several thousands of words) and often applying some feature subset selection approach (Mladenic and Grobelnik, 2003). The representation of a particular document contains many zeros, as most of the words from the collection do not occur in a particular document. The categories can be organized into a topic ontology, for example, the MeSH ontology for medical subject headings or the Yahoo! hierarchy of Web documents that

can be seen as a topic ontology[1]. Different Knowledge Discovery methods have been applied and evaluated on different document categorization problems. For instance, on the taxonomy of US patents, on Web documents organized in the Yahoo! Web directory (McCallum et al., 1998; Mladenic, 1998; Mladenic and Grobelnik 2004), on the DMoz Web directory (Grobelnik and Mladenic 2005), on categorization of Reuters news articles (Kholer and Sahami, 1997, Mladenic et all, 2004). Documents can also be related in ways other than common words (for instance, hyperlinks connecting Web documents) and these connections can be also used in document categorization (eg., (Craven and Slattery, 2001)).

2.6.3 Stream mining and Web mining

Ontology updating is important not only because the ontology construction process is demanding and frequently requires further extension, but also because of the dynamic nature of the world (part of which is reflected in an ontology). The underlying data and the corresponding semantic structures change in time, the ontology gets used, etc. As a consequence, we would like to be able to adapt the ontologies accordingly. We refer to these kind of structures as "dynamic ontologies" (ontology learning scenario 5 in Section 2.5). For most ontology updating scenarios, extensive human involvement in building models from the data is not economic, tending to be too costly, too inaccurate and too slow.

A sub-field of Knowledge Discovery called *Stream Mining* addresses the issue of rapidly changing data. The idea is to be able to deal with the stream of incoming data quickly enough to be able to simultaneously update the corresponding models (e.g., ontologies), as the amount of data is too large to be stored: new evidence from the incoming data is incorporated into the model without storing the data. The underlying methods are based on the machine learning methods of *on-line learning*, where the model is built from the initially available data and updated regularly as more data becomes available.

Web Mining, another sub-field of Knowledge Discovery, addresses Web data including three interleaved threads of research: Web *content* mining, Web *structure* mining and Web *usage* mining. As ontologies are used in different applications and by different users, we can make an analogy between usage of ontologies and usage of Web pages. For instance, in Web usage mining (Chakrabarti 2002), by analyzing frequencies of visits to particular Web pages and/or sequences of pages visited one after the other, one can consider restructuring the corresponding Web site or modeling the users behavior (eg., in Internet shops, a certain sequence of visiting Web pages may be more likely to lead to a purchase than the other sequence). Using similar methods, we can analyze the usage patters of an ontology to identify parts of the ontology that are hardly used and reconsider their formulation, placement or existence. The appropriateness of Web usage mining methods for ontology updating still needs to be confirmed by further research.

2.6.4 Focused crawling

An important step in ontology construction can be collecting the relevant data from the Web and using it for populating (ontology learning scenario 3 in Section 2.5) or updating

[1] The notion of a topic ontology is explored in detail in Chapter 7.

the ontology (ontology learning scenario 5 in Section 2.5). Collecting data relevant for the existing ontology can be also used in some other phases of the semi-automatic ontology construction process, such as ontology evaluation or ontology refinement (phases 5 and 6, Section 2.4), for instance, via associating new instances to the existing ontology in a process called ontology grounding (Jakulin and Mladenic 2005). In the case of topic ontologies (see Chapter 7), where the concepts correspond to topics and documents are linked to these topics through an appropriate relation such as hasSubject (Grobelnik and Mladenic 2005a), one can use the Web to collect documents on a predefined topic. In Knowledge Discovery, the approaches dealing with collecting documents based on the Web data are referred in the literature under the name *Focused Crawling* (Chakrabarti, 2002; Novak, 2004b). The main idea of these approaches is to use the initial "seed" information given by the user to find similar documents by exploiting (1) background knowledge (ontologies, existing document taxonomies, etc), (2) web topology (following hyper-links from the relevant pages), and (3) document repositories (through search engines). The general assumption for most of the focused crawling methods is that pages with more closely-related content are more inter-connected. In the cases where this assumption is not true (or we cannot reasonably assume it), we can still use the methods for selecting the documents through search engine querying (Ghani et al., 2005). In general, we could say that focused crawling serves as a generic technique for collecting data to be used in the next stages of data processing, such as constructing (ontology learning scenario 4 in Section 2.5) and populating ontologies (ontology learning scenario 3 in Section 2.5).

2.6.5 Data visualization

Visualization of data in general and also visualization of document collections is a method for obtaining early measures of data quality, content, and distribution (Fayyad et al., 2001). For instance, by applying document visualization it is possible to get an overview of the content of a Web site or some other document collection. This can be useful especially for the first phases of semi-automatic ontology construction aiming at domain and data understanding (see Section 2.4). Visualization can be also used for visualizing an existing ontology or some parts thereof, which is potentially relevant for all the ontology learning scenarios defined in Section 2.5.

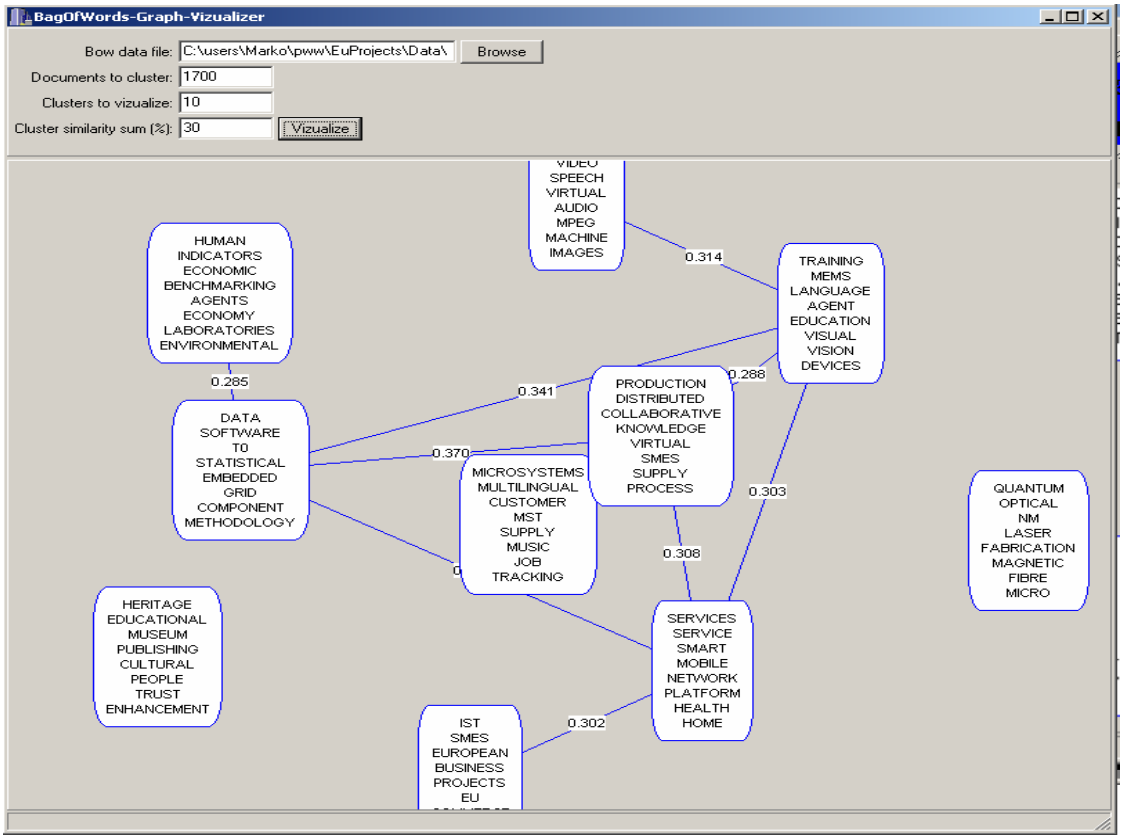

Figure 2.1. An example output of a system for graph-based visualization of document collection. The documents are 1700 descriptions of European research projects in information technology (5FP IST).

One general approach to document collection visualization is based on clustering of the documents (Grobelnik and Mladenic, 2002) by first representing the documents as word-vectors and performing k-means clustering on them (see Section 2.6.1). The obtained clusters are then represented as nodes in a graph, where each node in the graph is described by the set of most characteristic words in the corresponding cluster. Similar nodes, as measure by their cosine-similarity (equation (2.2)), are connected by a link. When such a graph is drawn, it provides a visual representation of the document set (see Figure 1 for an example output of the system). An alternative approach that provides different kinds of document corpus visualization is proposed in (Fortuna et all., 2005b). It is based on Latent Semantic Indexing, which is used to extract hidden semantic concepts from text documents and multidimensional scaling which is used to map the high dimensional space onto two dimensions. Document visualization can be also a part of more sophisticated tasks, such as generating a semantic graph of a document or supporting browsing through a news collection. For illustration, we provide two examples of document visualization that are based on Knowledge Discovery methods (see Figure 2.2 and Figure 2.3). Figure 2.2 shows an example of visualizing a single document via its semantic graph (Leskovec et all., 2004). Figure 2.3 shows an example of visualizing news stories via visualizing relationships between the named entities that appear in the news stories (Grobelnik and Mladenic, 2004).

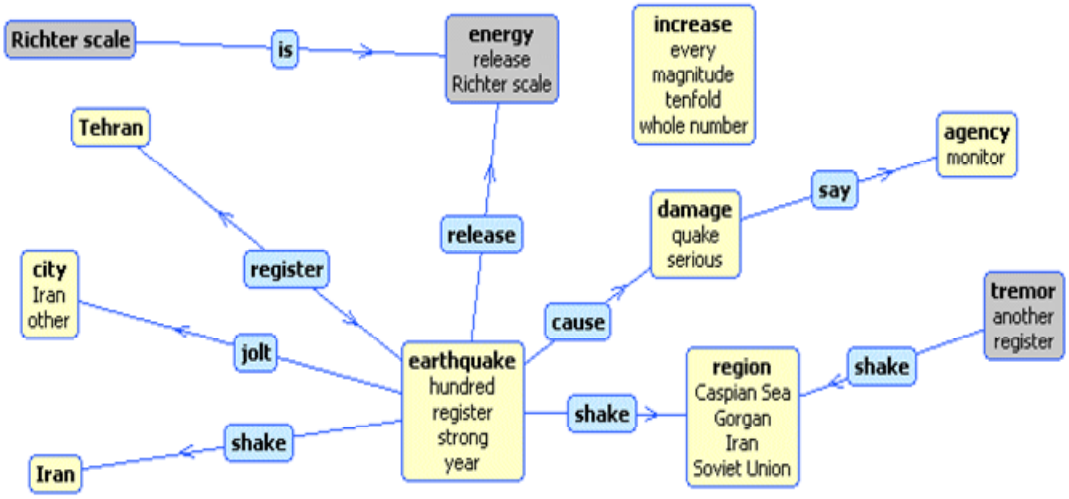

Figure 2.2. Visual representation of an automatically generated summary of a news story about earthquake. The summarization is based on deep parsing used for obtaining semantic graph of the document, followed by machine learning used for deciding which parts of the graph are to be included in the document summary.

2.7. Related Work on Ontology Construction

Different approaches have been used for building ontologies, most of them to date using mainly manual methods. An approach to building ontologies was set up in the CYC project (Lenat and Guha, 1990), where the main step involved manual extraction of common sense knowledge from different sources. There have been some methodologies for building ontologies developed, again assuming a manual approach. For instance, the methodology proposed in (Uschold and King, 1995) involves the following stages: *identifying the purpose of the ontology* (why to build it, how will it be used, the range of the users), *building the ontology*, *evaluation and documentation*. Building of the ontology is further divided into three steps. The first is ontology capture, where key concepts and relationships are identified, a precise textual definition of them is written, terms to be used to refer to the concepts and relations are identified, the involved actors agree on the definitions and terms. The second step involves coding of the ontology to represent the defined conceptualization in some formal language (committing to some meta-ontology, choosing a representation language and coding). The third step involves possible integration with existing ontologies. An overview of methodologies for building ontologies is provided in (Fernández, 1999), where several methodologies, including the above described one, are presented and analyzed against the IEEE Standard for Developing Software Life Cycle Processes, thus viewing ontologies as parts of some software product. As there are some specifics to semi-automatic ontology construction compared to the manual approaches to ontology construction, the methodology that we have defined (see Section 2.4) has six phases. If we related them to the stages in the

methodology defined in (Uschold and King, 1995), we can see that the first two phases referring to domain and data understanding roughly correspond to *identifying the purpose of the ontology*, the next two phases (tasks definition and ontology learning) correspond to the stage of *building the ontology*, and the last two phases on ontology evaluation and refinement correspond to the *evaluation and documentation* stage.

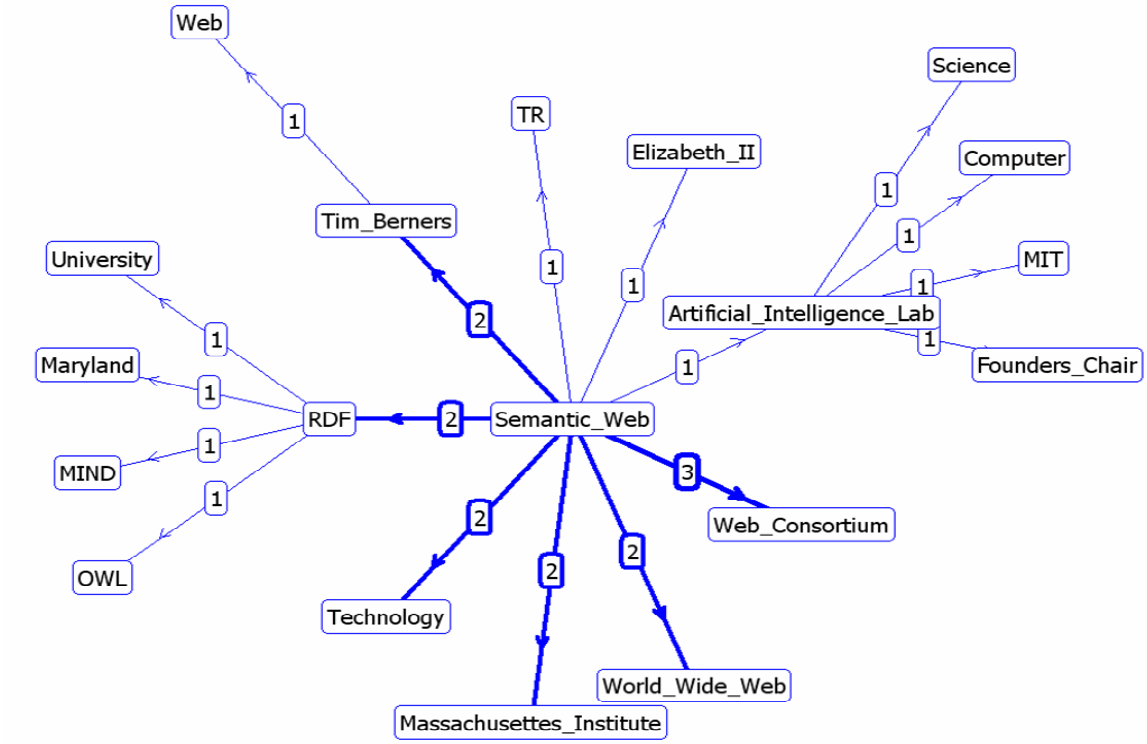

Figure 2.3. Visual representation of relationships (edges in the graph) between the named entities (vertices in the graph) appearing in a collection of news stories. Each edge shows intensity of co-mentioning of the two named entities. The graph is an example focused on the named entity "Semantic Web" that was extracted from the 11.000 ACM Technology news stories from 2000-2004.

Several workshops at the main Artificial Intelligence and Knowledge Discovery conferences (ECAI, IJCAI, KDD, ECML/PKDD) have been organized addressing the topic of ontology learning. Most of the work presented there addresses one of the following problems/tasks:

- **Extending the existing ontology.** Given an existing ontology with concepts and relations (commonly used is the English lexical ontology WordNet), the goal is to extend that ontology using some text, e.g.Web documents are used in (Agirre et al., 2000). This can fit under the ontology learning scenario 5 in Section 2.5.
- **Learning relations for an existing ontology.** Given a collection of text documents and ontology with concepts, learn relations between the concepts. The approaches include learning taxonomic, eg., isa, (Cimiano et al., 2004) and non-taxonomic, eg., "hasPart" relations (Maedche and Staab, 2001) and extracting semantic relations from text based on collocations (Heyer et al., 2001). This fits under the ontology learning scenario 2 in Section 2.5.

- **Ontology construction based on clustering.** Given a **collection of text documents**, split each document into sentences, parse the text and apply clustering for semi-automatic construction of an ontology (Bisson et al., 2000; Reinberger et al., 2004). Each cluster is labeled by the most characteristic words from its sentences or using some more sophisticated approach (Popescull and Ungar, 2000). Documents can be also used as a whole, without splitting them into sentences, and guiding the user through a semi-automatic process of ontology construction (Fortuna et all., 2005a). The system provides suggestions for ontology concepts, automatically assigns documents to the concepts, proposed naming of the concepts, etc. In (Hotho et al., 2003) the clustering is further refined by using WordNet to improve the results by mapping the found sentence clusters upon the concepts of a general ontology. The found concepts can be further used as semantic labels (XML tags) for annotating documents. This fits under the ontology learning scenario 4 in Section 2.5.
- **Ontology construction based on semantic graphs.** Given a **collection of text documents**, parse the documents; perform co-reference resolution, anaphora resolution, extraction of subject-predicate-object triples and construct semantic graphs. These are further used for learning summaries of the documents (Leskovec et al., 2004). An example summary obtained using this approach is given in Figure 2.2. This can fit under the ontology learning scenario 4 in Section 2.5.
- **Ontology construction from a collection of news stories based on named entities.** Given a collection of news stories, represent it as a collection of graphs, where the nodes are named entities extracted from the text and relationships between them are based on the context and collocation of the named entities. These are further used for visualization of news stories in an interactive browsing environment (Grobelnik and Mladenic, 2004). An example output of the proposed approach is given in Figure 2.3. This can fit under the ontology learning scenario 4 in Section 2.5.

More information on ontology learning from text can be found in a collection of papers (Buitelaar et al., 2005) addressing three perspectives: *methodologies* that have been proposed to automatically extract information from texts, *evaluation methods* defining procedures and metrics for a quantitative evaluation of the ontology learning task, and *application scenarios* that make ontology learning a challenging area in the context of real applications.

2.8. Discussion and Conclusion

We have presented several techniques from Knowledge Discovery that are useful for semi-automatic ontology construction. In that light, we propose to decompose the semi-automatic ontology construction process into several phases ranging from *domain and data understanding* through *task definition* via *ontology learning* to *ontology evaluation* and *refinement*. A large part of this chapter is dedicated to ontology learning. Several scenarios are identified in the ontology learning phase depending on different assumptions regarding the provided input data and the expected output: inducing concepts, inducing relations, ontology population, ontology construction and ontology updating/extension. Different groups of Knowledge Discovery techniques are briefly

described including unsupervised learning, semi-supervised, supervised and active learning, on-line learning and web mining, focused crawling, data visualization. In addition to providing brief description of these techniques, we also relate them to different ontology learning scenarios that we identified.

Some of the described Knowledge Discovery techniques have already been applied in the context of semi-automatic ontology construction, while others still need to be adapted and tested in that context. A challenge for future research is setting up evaluation frameworks for assessing contribution of these techniques to specific tasks and phases of the ontology construction process. In that light, we briefly describe some existing approaches to ontology construction and point to the original papers that provide more information on the approaches, usually including some evaluation of their contribution and performance on the specific tasks. We also related existing work on learning ontologies to different ontology learning scenarios that we have identified. Our hope is that this chapter in addition to contributing by proposing a methodology for semi-automatic ontology construction and description of some relevant Knowledge Discovery techniques also shows potential for future research and triggers some new ideas related to the usage of Knowledge Discovery techniques for ontology construction.

Acknowledgements

This work was supported by the Slovenian Research Agency and the IST Programme of the European Community under SEKT Semantically Enabled Knowledge Technologies (IST-1-506826-IP) and PASCAL Network of Excellence (IST-2002-506778). This publication only reflects the authors' views.

References

Agirre, E., Ansa, O., Hovy, E., Martínez, D. (2000). Enriching very large ontologies using the WWW. In Proceedings of the First Workshop on Ontology Learning OL-2000. The 14th European Conference on Artificial Intelligence ECAI-2000.

Bisson, G, Nédellec, C., Cañamero, D. (2000). Designing clustering methods for ontology building: The Mo'K workbench. In Proceedings of the First Workshop on Ontology Learning OL-2000. The 14th European Conference on Artificial Intelligence ECAI-2000.

Bloehdorn, S., Haase, P., Sure, Y, Voelker, J., Bevk, M., Bontcheva, K., Roberts, I. (2005). Report on the integration of ML, HLT and OM. SEKT Deliverable D.6.6.1, July 2005.

Blum, A., Chawla, S. (2001). Learning from Labelled and Unlabelled Data Using Graph Mincuts, Proceedings of the 18th International Conf. on Machine Learning, pg. 19-26.

Buitelaar, P., Cimiano, P., Magnini, B. (2005). Ontology Learning from Text: Methods, Applications and Evaluation, Frontiers in Artificial Intelligence and Applications, IOS Press.

Mladenic, D., Brank, J., Grobelnik, M., Milic-Frayling, N. (2002). Feature Selection using Linear Classifier Weights: Interaction with Classification Models, SIGIR-2002.

Brank, J., Grobelnik, M., Mladenic, D. (2005). A survey of ontology evaluation techniques. Proceedings of the 8th International multi-conference Information Society IS-2005, Ljubljana: Institut "Jožef Stefan", 2005.

Chakrabarti. S. (2002). Mining the Web: Analysis of Hypertext and Semi Structured Data, Morgan Kaufmann.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., Wirth, R. (2000). CRISP-DM 1.0: Step-by-step data mining guide.

Cimiano, P., Pivk, A., Schmidt-Thieme, L., Staab, S. (2004) Learning Taxonomic Relations from Heterogeneous Evidence. In Proceedings of ECAI 2004 Workshop on Ontology Learning and Population.

Craven, M., Slattery, S., (2001). Relational learning with statistical predicate invention: Better models for hypertext. Machine Learning, 43(1/2):97-119.

Cunningham, H., Bontcheva, K. (2005) Knowledge Management and Human Language: Crossing the Chasm. Journal of Knowledge Management.

Deerwester, S., Dumais, S., Furnas, G., Landuer, T., Harshman, R., (2001). Indexing by Latent Semantic Analysis.

Duda, R. O., Hart, P. E. and Stork, D. G. (2000). Pattern Classification 2nd edition, Wiley-Interscience.

Ehrig, M., Haase, P., Hefke, M., Stojanovic, N. (2005). Similarity for ontologies — a comprehensive framework. Proc. 13th European Conference on Information Systems, May 2005.

Fayyad, U., Grinstein, G. G. and Wierse, A. (eds.), (2001). Information Visualization in Data Mining and Knowledge Discovery, Morgan Kaufmann.

Fayyad, U., Piatetski-Shapiro, G., Smith, P., and Uthurusamy R. (eds.) (1996) Advances in Knowledge Discovery and Data Mining. MIT Press, Cambridge, MA, 1996.

Fernández, L.M., (1999). Overview Of Methodologies For Building Ontologies. In Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods (KRR5).

Fortuna, B., Mladenic, D., Grobelnik, M., (2005a). Semi-automatic construction of topic ontology. Proceedings of the ECML/PKDD Workshop on Knowledge Discovery for Ontologies.

Fortuna, B., Mladenic, D., Grobelnik, M., (2005b). Visualization of text document corpus. Informatica journal, 2005, vol. 29, no. 4, pp. 497-502.

Ghani, R., Jones, R., Mladenic, D., (2005). Building minority language corpora by learning to generate web search queries. Knowledge and information systems, 2005, vol. 7, pp. 56-83.

Grobelnik, M., and Mladenic, D., (2002). Efficient visualization of large text corpora. Proceedings of the seventh TELRI seminar. Dubrovnik, Croatia.

Grobelnik, M., Mladenic, D. (2004). Visualization of news articles. Informatica journal, 2004, vol. 28, no. 4.

Grobelnik, M., Mladenic, D. (2005). Simple classification into large topic ontology of Web documents, Journal of Computing and Information Technology – CIT 13, 2005, 4, pp. 279 - 285.

Grobelnik, M., Mladenic, D. (2005a). Automated Knowledge Discovery in Advanced Knowledge Management. Journal of Knowledge Management.

Hand, D.J., Mannila, H., Smyth, P. (2001) Principles of Data Mining (Adaptive Computation and Machine Learning), MIT Press.

Hastie, T., Tibshirani, R. and Friedman, J. H. (2001). The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer Series in Statistics, Springer Verlag.

Heyer, G., Läuter, M., Quasthoff, U., Wittig, T., Wolff, C. (2001) Learning Relations using Collocations. In Proceedings of IJCAI-2001 Workshop on Ontology Learning.

Hotho, A., Staab, S., Stumme, G. (2003) Explaining text clustering results using semantic structures. In Proceedings of ECML/PKDD 2003, LNAI 2838, pages 217-228, Springer Verlag.

Jackson, P., Moulinier, I., (2002). Natural Language Processing for Online Applications: Text Retrieval, Extraction, and Categorization, John Benjamins Publishing Co.

Jakulin, A., Mladenic, D., (2005). Ontology Grounding, Proceedings of the 8th International multi-conference Information Society IS-2005, Ljubljana: Institut "Jožef Stefan", 2005.

Koller, D., Sahami, M., (1997). Hierarchically classifying documents using very few words, Proceedings of the 14th International Conference on Machine Learning ICML-97, pp. 170-178, Morgan Kaufmann, San Francisco, CA.

Leskovec, J., Grobelnik, M., Milic-Frayling, N. (2004). Learning Sub-structures of Document Semantic Graphs for Document Summarization. In Workshop on Link Analysis and Group Detection (LinkKDD2004). The Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

Maedche, A., Staab, S. (2001). Discovering conceptual relations from text. In Proc. of ECAI'2000, pages
321-325.

McCallum A., Rosenfeld R., Mitchell T., Ng A., (1998). Improving Text Classification by Shrinkage in a Hierarchy of Classes, Proceedings of the 15th International Conference on Machine Learning ICML-98, Morgan Kaufmann, San Francisco, CA.

Manning, C.D., Schutze, H. (2001).Foundations of Statistical Natural Language Processing, The MIT Press, Cambridge, MA.

Mitchell, T.M. (1997). Machine Learning. The McGraw-Hill Companies, Inc.

Mladenic, D. (1998). Turning Yahoo into an Automatic Web-Page Classifier. Proc. 13th European Conference on Artificial Intelligence (ECAI'98, John Wiley & Sons), 473–474.

Mladenic, D., Grobelnik, M. (2003). Feature selection on hierarchy of web documents. Journal of Decision support systems, 35, 45-87.

Mladenic, D., Grobelnik, M. (2004). Mapping documents onto web page ontology. In: Web mining : from web to semantic web (Berendt, B., Hotho, A., Mladenic, D., Someren, M.W. Van, Spiliopoulou, M., Stumme, G., eds.), Lecture notes in artificial inteligence, Lecture notes in computer science, vol. 3209, Berlin; Heidelberg; New York: Springer, 2004, 77-96.

Novak, B., (2004a). Use of unlabeled data in supervised machine learning. Proceedings of the 7th International multi-conference Information Society IS-2004, Ljubljana: Institut "Jožef Stefan", 2004.

Novak, B., (2004b). A survey of focused web crawling algorithms. Proceedings of the 7th International multi-conference Information Society IS-2004, Ljubljana: Institut "Jožef Stefan", 2004.

Popescul, A., Ungar, L.H. (2000). Automatic labeling of document clusters. Department of Computer and Information Science, University of Pennsylvania, unpublished paper available from
http://www.cis.upenn.edu/~popescul/Publications/popescul00labeling.pdf

Reinberger, M-L., Spyns, P. (2004) Discovering Knowledge in Texts for the learning of DOGMA-inspired ontologies. In Proceedings of ECAI 2004 Workshop on Ontology Learning and Population.

Sebastiani, F., Machine Learning for Automated Text Categorization, ACM Computing Surveys, 2002.

Steinbach, M., Karypis, G. and Kumar, V. (2000). A comparison of document clustering techniques. Proc. KDD Workshop on Text Mining. (eds. Grobelnik, M., Mladenić, D. and Milic-Frayling, N.), Boston, MA, USA, 109–110.

Uschold, M., King, M. (1995). Towards a methodology for building ontologies. In Workshop on Basic Ontological Issues in Knowledge Sharing. International Joint Conference on Arti.cial Intelligence, 1995. Also available as AIAI-TR-183 from AIAI, the University of Edinburgh.

van Rijsbergen, C. J. (1979), Information Retrieval, 2nd Edition, London:Butterworths.

Witten, I.H., Frank, E., (1999) Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann.