Web Semantics: Science, Services and Agents on the World Wide Web 36 (2016) 1–22

Contents lists available at ScienceDirect

Web Semantics: Science, Services and Agents on the World Wide Web

journal homepage: www.elsevier.com/locate/websem

Review article

Semantic Web in data mining and knowledge discovery: A comprehensive survey

Petar Ristoski *, Heiko Paulheim

Data and Web Science Group, University of Mannheim, B6, 26, 68159 Mannheim, Germany

ARTICLE INFO

Article history:
Received 28 March 2015
Received in revised form
5 November 2015
Accepted 1 January 2016
Available online 8 January 2016

Keywords:
Linked Open Data
Semantic Web
Data mining
Knowledge discovery

ABSTRACT

Data Mining and Knowledge Discovery in Databases (KDD) is a research field concerned with deriving higher-level insights from data. The tasks performed in that field are knowledge intensive and can often benefit from using additional knowledge from various sources. Therefore, many approaches have been proposed in this area that combine Semantic Web data with the data mining and knowledge discovery process. This survey article gives a comprehensive overview of those approaches in different stages of the knowledge discovery process. As an example, we show how Linked Open Data can be used at various stages for building content-based recommender systems. The survey shows that, while there are numerous interesting research works performed, the full potential of the Semantic Web and Linked Open Data for data mining and KDD is still to be unlocked.

Contents

* Corresponding author.
E-mail addresses: petar.ristoski@informatik.uni-mannheim.de (P. Ristoski), heiko@informatik.uni-mannheim.de (H. Paulheim).

http://dx.doi.org/10.1016/j.websem.2016.01.001
1570-8268/

1. Introduction

Data mining is defined as "a non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data" [1], or "the analysis of (often large) observational datasets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner" [2]. As such, data mining and knowledge discovery are typically considered knowledge intensive tasks. Thus, knowledge plays a crucial role here. Knowledge can be (a) in the primary data itself, from where it is discovered using appropriate algorithms and tools, (b) in external data, which has to be included with the problem first (such as background statistics or master file data not yet linked to the primary data), or (c) in the data analyst's mind only.

The latter two cases are interesting opportunities to enhance the value of the knowledge discovery processes. Consider the following case: a dataset consists of countries in Europe and some economic and social indicators. There are, for sure, some interesting patterns that can be discovered in the data. However, an analyst dealing with such data on a regular basis will know that some of the countries are part of the European Union, while others are not. Thus, she may add an additional variable `EU_Member` to the dataset, which may lead to new insights (e.g., certain patterns holding for EU member states only).

In that example, knowledge has been added to the data from the analyst's mind, but it might equally well have been contained in some exterior source of knowledge, such as Linked Open Data.

Linked Open Data (LOD) is an open, interlinked collection of datasets in machine-interpretable form, covering multiple domains from life sciences to government data [3,4]. Thus, it should be possible to make use of that vault of knowledge in a given data mining, at various steps of the knowledge discovery process.

Many approaches have been proposed in the recent past for using LOD in data mining processes, for various purposes, such as the creation of additional variables, as in the example above. With this paper, we provide a structured survey of such approaches. Following the well-known data mining process model proposed by Fayyad et al. [1], we discuss how semantic data is exploited at the different stages of the data mining model. Furthermore, we analyze how different characteristics of Linked Open Data, such as the presence of interlinks between datasets and the usage of ontologies as schemas for the data, are exploited by the different approaches.

The rest of this paper is structured as follows. Section 2 sets the scope of this survey, and puts it in the context of other surveys in similar areas. Section 3 describes the knowledge discovery process according to Fayyad et al. In Section 4, we introduce a general model for data mining using Linked Open Data, followed by a description of approaches using Semantic Web data in the different stages of the knowledge discovery process in Sections 5 through 9. In Section 10, we give an example use-case of LOD-enabled KDD process in the domain of recommender systems. We conclude with a summary of our findings, and identify a number of promising directions for future research.

2. Scope of this survey

In the last decade, a vast amount of approaches have been proposed which combine methods from data mining and knowledge discovery with Semantic Web data. The goal of those approaches is to support different data mining tasks, or to improve the Semantic Web itself. All those approaches can be divided into three broader categories:

- Using Semantic Web based approaches, Semantic Web Technologies, and Linked Open Data to support the process of knowledge discovery.
- Using data mining techniques to mine the Semantic Web, also called *Semantic Web Mining*.
- Using machine learning techniques to create and improve Semantic Web data.

Stumme et al. [5] have provided an initial survey of all three categories, later focusing more on the second category. Dating back to 2006, this survey does not reflect recent research works and trends, such as the advent and growth of Linked Open Data. More recent surveys on the second category, i.e., Semantic Web Mining, have been published by Sridevi et al. [6], Quboa et al. [7], Sivakumar et al. [8], and Dou et al. [9].

Tresp et al. [10] give an overview of the challenges and opportunities for the third category, i.e., machine learning on the Semantic Web, and using machine learning approaches to support the Semantic Web. The work has been extended in [11].

In contrast to those surveys, the first category – i.e., the usage of Semantic Web and Linked Open Data to support and improve data mining and knowledge discovery – has not been subject of a recent survey. Thus, in this survey, we focus on that area.

The aim of this survey is to give a survey on the field as broad as possible, i.e., capturing as many different research directions as possible. As a consequence, a direct comparison of approaches is not always possible, since they may have been developed with slightly different goals, tailored towards particular use cases and/or datasets, etc. Nevertheless, we try to formulate at least coarse-grained comparisons and recommendations, wherever possible.

3. The knowledge discovery process

In their seminal paper from 1996, Fayyad et al. introduced a process model for knowledge discovery processes. The model comprises five steps, which lead from raw data to actionable knowledge and insights which are of immediate value to the user. The whole process is shown in Fig. 1. It comprises five steps:

1. *Selection* The first step is developing an understanding of the application domain, capturing relevant prior knowledge, and identifying the data mining goal from the end user's perspective. Based on that understanding, the target data used in the knowledge discovery process can be chosen, i.e., selecting proper data samples and a relevant subset of variables.
2. *Preprocessing* In this step, the selected data is processed in a way that allows for a subsequent analysis. Typical actions taken in this step include the handling of missing values, the identification (and potentially correction) of noise and errors in the data, the elimination of duplicates, as well as the matching, fusion, and conflict resolution for data taken from different sources.
3. *Transformation* The third step produces a projection of the data to a form that data mining algorithms can work on—in most cases, this means turning the data into a propositional form, where each instance is represented by a feature vector. To improve the performance of subsequent data mining

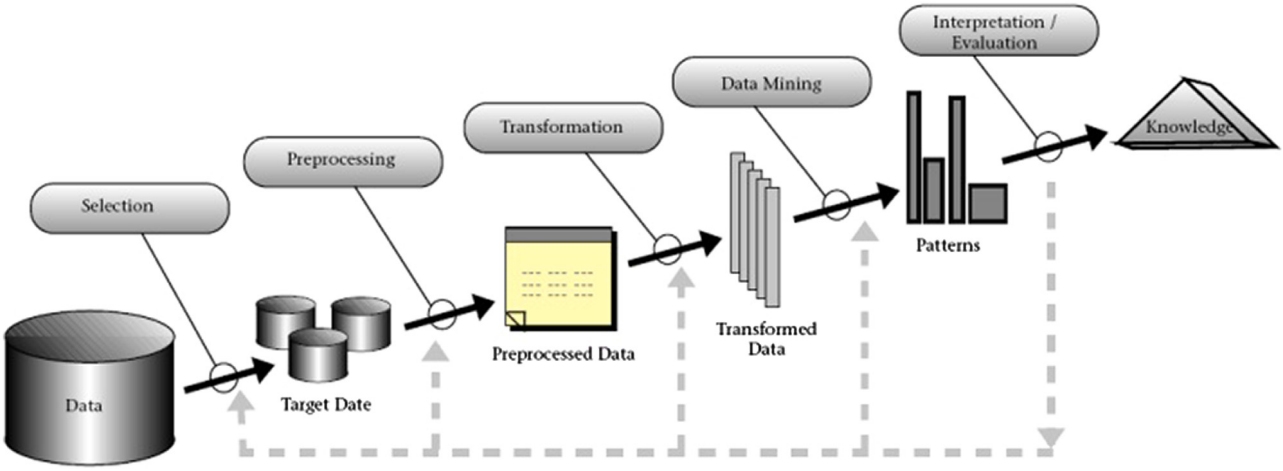

Fig. 1. An overview of the steps that compose the KDD process.

algorithms, dimensionality reduction methods can also be applied in this step to reduce the effective number of variables under consideration.

4. *Data mining* Once the data is present in a useful format, the initial goal of the process is matched to a particular method, such as classification, regression, or clustering. This step includes deciding which models and parameters might be appropriate (for example, models for categorical data are different than models for numerical data), and matching a particular data mining method with the overall criteria of the KDD process (for example, the end user might be more interested in an interpretable, but less accurate model than a very accurate, but hard to interpret model). Once the data mining method and algorithm are selected, the data mining takes place: searching for patterns of interest in a particular representational form or a set of such representations, such as rule sets or trees.
5. *Evaluation and interpretation* In the last step, the patterns and models derived by the data mining algorithm(s) are examined with respect to their validity. Furthermore, the user assesses the usefulness of the found knowledge for the given application. This step can also involve visualization of the extracted patterns and models, or visualization of the data using the extracted models.

The quality of the found patterns depends on the methods being employed in each of these steps, as well as their interdependencies. Thus, the process model foresees the possibility to go back to each previous step and revise decisions taken at that step, as depicted in Fig. 1. This means that the overall process is usually repeated after adjusting the parametrization or even exchanging the methods in any of these steps until the quality of the results is sufficient.

4. Data mining using linked open data

As a means to express knowledge about a domain in the Semantic Web, *ontologies* have been introduced in the early 1990s as "explicit formal specifications of the concepts and relations among them that can exist in a given domain" [12]. For the area of knowledge discovery and data mining, Nigro et al. [13] divide ontologies used in this area into three categories:

- *Domain ontologies:* Express background knowledge about the application domain, i.e., the domain of the data at hand on which KDD and data mining are performed.
- *Ontologies for data mining process:* Define knowledge about the data mining process, its steps and algorithms and their possible parameters.
- *Metadata ontologies:* Describe meta knowledge about the data at hand, such as provenance information, e.g., the processes used to construct certain datasets.

It has been already shown that ontologies for the data mining process and metadata ontologies can be used in each step of the KDD process. However, we want to put a stronger focus on the usage of Linked Open Data (LOD) in the process of knowledge discovery, which represents a publicly available interlinked collection of datasets from various topical domains [3,4].

Fig. 2 gives an overview of the Linked Open Data enabled knowledge discovery pipeline. Given a set of local data (such as a relational database), the first step is to link the data to the corresponding LOD concepts from the chosen LOD dataset (cf. Section 5).[1] Once the local data is linked to a LOD dataset, we can explore the existing links in the dataset pointing to the related entities in other LOD datasets. In the next step, various techniques for data consolidation, preprocessing and cleaning are applied, e.g., schema matching, data fusion, value normalization, treatment of missing values and outliers, etc. (cf. Section 6). Next, some transformations on the collected data need to be performed in order to represent the data in a way that it can be processed with any arbitrary data analysis algorithms (cf. Section 7). Since most algorithms demand a propositional form of the input data, this usually includes a transformation of the graph-based LOD data to a canonical propositional form. After the data transformation is done, a suitable data mining algorithm is selected and applied on the data (cf. Section 8). In the final step, the results of the data mining process are presented to the user. Here, ease the interpretation and evaluation of the results of the data mining process, Semantic Web and LOD can be used as well (cf. Section 9).

For the survey presented in the following section, we have compiled a list of approaches that fulfill the following criteria:

1. They are designed and suitable for improving the KDD process in at least one step.
2. They make use of one or more datasets on the Semantic Web.

Each of the approaches is assessed using a number of criteria:

1. Is the approach domain-independent or tailored to a specific domain?

[1] We should note that the data can be linked to the LOD datasets in different stages of the KDD process, for example, in some approaches only the results and the discovered patterns from the data mining process are linked to a given LOD dataset in order to ease the interpretation of them. For simplicity's sake we describe the process of linking as the first step, which is also depicted in Fig. 2.

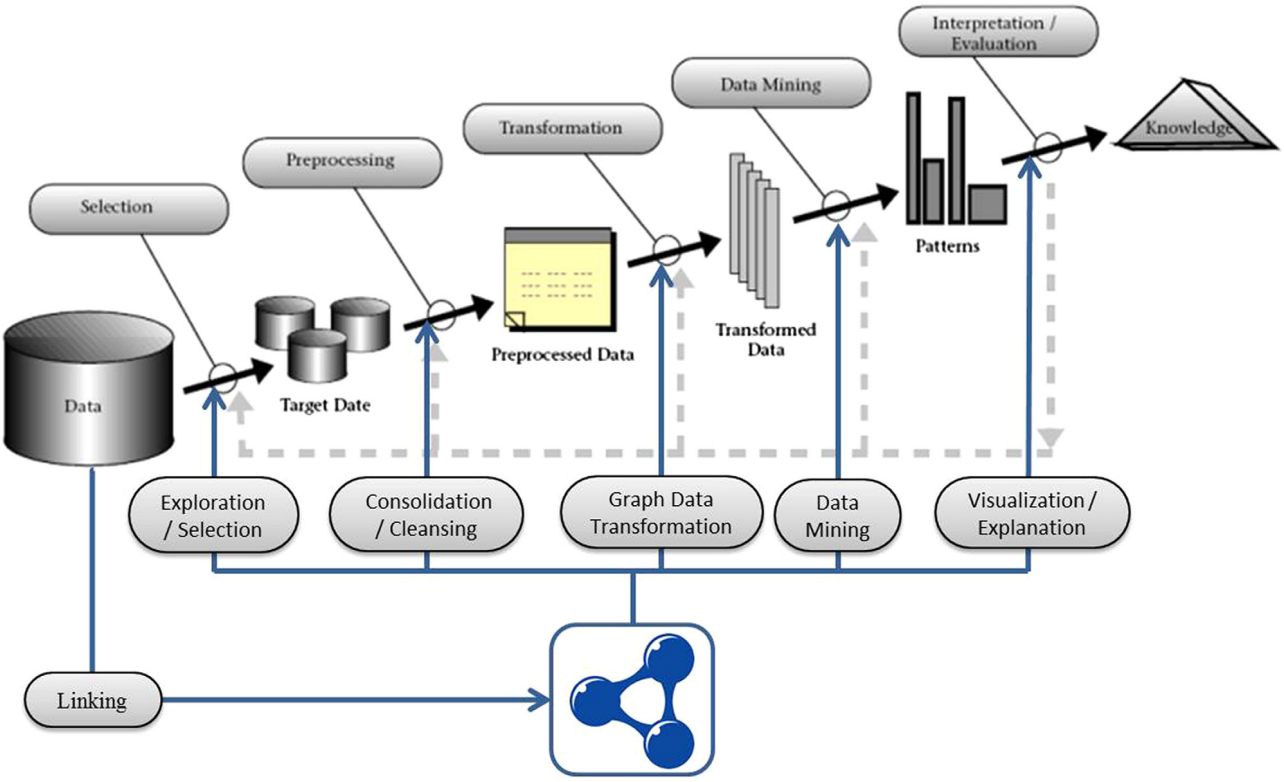

Fig. 2. An overview of the steps of the linked open data enabled KDD pipeline.

2. Is the approach tailored towards a specific data mining technique (e.g., rule induction)?
3. Does it use a complex ontology or only a weakly axiomatized one (such as a hierarchy)?
4. Is any reasoning involved?
5. Are links to other datasets (a core ingredient of Linked Open Data) used?
6. Are the semantics of the data (i.e., the ontology) exploited?

Furthermore, we analyze which Semantic Web datasets are used in the papers, to get a grasp of which are the most prominently used ones.

In the following sections, the survey introduces and discusses the individual approaches.[2] A small box at the end of each section gives a brief summary, a coarse-grained comparison, and some guidelines for data mining practitioners who want to use the approaches in actual projects.

5. Selection

To develop a good understanding of the application domain, and the data mining methods that are appropriate for the given data, a deeper understanding of the data is needed. First, the user needs to understand what is the domain of the data, what knowledge is captured in the data, and what is the possible additional knowledge that could be extracted from the data. Then, the user can identify the data mining goal more easily, and select a sample of the data that would be appropriate for reaching that goal.

However, the step of understanding the data is often not trivial. In many cases, the user needs to have domain specific knowledge in order to successfully understand the data. Furthermore, the data at hand is often represented in a rather complex structure that contains hidden relations.

To overcome this problem, several approaches propose using Semantic Web techniques for better representation and exploration of the data, by exploiting domain specific ontologies and Linked Open Data. This is the first step of the Semantic Web enhanced KDD pipeline, called *linking*. In this step, a *linkage*, or *mapping*, to existing ontologies, and LOD datasets is performed on the local data.

Once the linking is done, additional background knowledge for the local data can be automatically extracted. That allows to formally structure the domain concepts and information about the data, by setting formal types, and relations between concepts. Using background knowledge in many cases the users can easily understand the data domain, without the need for employing domain experts.

Furthermore, many tools for visualization and exploration of LOD data exist that would allow an easier and deeper understanding of the data. An overview of tools and approaches for visualization and exploration of LOD is given in the survey by Dadzie et al. [14]. The authors first set the requirements or what is expected of the tools for visualization or browsing the LOD: (i) the ability to generate an overview of the underlying data, (ii) support for filtering out less important data in order to focus on selected regions of interest (ROI), and (iii) support for visualizing the detail in ROIs. Furthermore, all these tools should allow the user to intuitively navigate through the data, explore entities and relations between them, explore anomalies within the data, perform advanced querying, and data extraction for reuse. They divided the analyzed browsers between those offering a text-based presentation, like Disco[3] and Sig.ma [15] and Piggy

[2] We should note that some of the approaches might be applicable in several steps of the LOD-enabled KDD pipeline. However, in almost all cases, there is one step which is particularly in the focus of that work, and we categorize those works under that step.

[3] http://www4.wiwiss.fu-berlin.de/bizer/ng4j/disco.

Bank [16], and those with visualization options, like Fenfire [17], IsaViz,[4] and RelFinder.[5] The analysis of the approaches shows that most of the text-based browsers provide functionalists to support the tech-users, while the visualization-based browsers are mostly focused on the non-tech users. Even though the authors conclude that there is only a limited number of SW browsers available, we can still make use of them to understand the data better and select the data that fits the data analyst's needs. The categorization of approaches in the survey by Dadzie et al. has been extended by Peña et al. [18], based on the datatypes that are visualized and the functionality needed by the analysts. The authors list some more recent approaches for advanced LOD visualization and exploration, like CODE [19], LDVizWiz [20], LODVisualization [21], and Payola [22].

The approaches for linking local data to LOD can be divided into three broader categories, based on the initial structural representation of the local data:

5.1. Using LOD to interpret relational databases

Relational databases are considered as one of the most popular storage solutions for various kinds of data, and are widely used. The data represented in relational databases is usually backed by a schema, which formally defines the entities and relations between them. In most of the cases, the schema is specific for each database, which does not allow for automatic data integration from multiple databases. For easier and automatic data integration and extension, a global shared schema definition should be used across databases.

To overcome this problem, many approaches for mapping relational databases to global ontologies and LOD datasets have been proposed. In recent surveys [23–25] the approaches have been categorized in several broader categories, based on three criteria: existence of an ontology, domain of the generated ontology, and application of database reverse engineering. Additionally, [25] provides a list of the existing tools and frameworks for mapping relational databases to LOD, from which the most popular and most used is the D2RQ tool [26]. D2RQ is a declarative language to describe mappings between application-specific relational database schemata and RDF-S/OWL ontologies. Using D2RQ, Semantic Web applications can query a non-RDF database using RDQL, publish the content of a non-RDF database on the Semantic Web using the RDF Net API,[6] do RDFS and OWL inferencing over the content of a non-RDF database using the Jena ontology API,[7] and access information in a non-RDF database using the Jena model API.[8] D2RQ is implemented as a Jena graph, the basic information representation object within the Jena framework. A D2RQ graph wraps one or more local relational databases into a virtual, read-only RDF graph. D2RQ rewrites RDQL queries and Jena API calls into application-datamodel-specific SQL queries. The result sets of these SQL queries are transformed into RDF triples which are passed up to the higher layers of the Jena framework.

5.2. Using LOD to interpret semi-structured data

In many cases, the data at hand is represented in a semi structured representation, meaning that the data can be easily understand by humans, but it cannot be automatically processed by machines, because it is not backed by a schema or any other formal representation. One of the most used semi-structure representations of data is the tabular representation, found in documents, spreadsheets, on the Web or databases. Such representation often follows a simple structure, and unlike relational databases, there is no explicit representation of a schema.

Evidence for the semantics of semi-structured data can be found, e.g., in its column headers, cell values, implicit relations between columns, as well as caption and surrounding text. However, general and domain-specific background knowledge is needed to interpret the meaning of the table.

Many approaches have been proposed for extracting the schema of the tables, and mapping it to existing ontologies and LOD. Mulwad et al. have made significant contribution for interpreting tabular data using LOD, coming from independent domains [27–32]. They have proposed several approaches that use background knowledge from the Linked Open Data cloud, like Wikitology [33], DBpedia [34], YAGO [35], Freebase [36] and WordNet [37], to infer the semantics of column headers, table cell values and relations between columns and represent the inferred meaning as graph of RDF triples. A table's meaning is thus captured by mapping columns to classes in an appropriate ontology, linking cell values to literal constants, implied measurements, or entities in the LOD cloud and identifying relations between columns. Their methods range from simple index lookup from a LOD source, to techniques grounded in graphical models and probabilistic reasoning to infer meaning associated with a table [32], which are applicable on different types of tables. i.e., relational tables, quasi-relational (Web) tables and spreadsheets tables.

Liu et al. [38] propose a learning-based semantic search algorithm to suggest appropriate Semantic Web terms and ontologies for the given data. The approach combines various measures for semantic similarity of documents to build a weighted feature-based semantic search model, which is then able to find the most suitable ontologies. The weights are learned from training data, using subgradient descent method and logistic regression.

Limaye et al. [39] propose a new probabilistic graphical model for simultaneously choosing entities for cells, types for columns and relations for column pairs, using YAGO as a background knowledge base. For building the graphical models, several types of features were used, i.e., cell text and entity label, column type and type label, column type and cell entity, relation and pair of column types, relation and entity pairs. The experiments showed that approaching the three sub-problems collectively and in a unified graphical inference framework leads to higher accuracy compared to making local decisions.

Venetis et al. [40] associate multiple class labels (or concepts) with columns in a table and identify relations between the "subject" column and the rest of the columns in the table. Both the concept identification for columns and relation identification are based on maximum likelihood hypothesis, i.e., the best class label (or relation) is the one that maximizes the probability of the values given the class label (or relation) for the column. The evidences for the relations and for the classes are retrieved from a previously extracted isA database, describing the classes of the entities, and relations database, which contains relations between the entities. The experiments show that the approach can obtain meaningful labels for tables that rarely exist in the tables themselves, and that considering the recovered semantics leads to high precision search with little loss of recall of tables in comparison to document based approaches.

Wang et al. [41] propose a multi-phase algorithm that using the universal probabilistic taxonomy called *Probase* [42] is capable of understanding the entitles, attributes and values in many tables on the Web. The approach begins by identifying a single "entity column" in a table and, based on its values and rest of the column

4 http://www.w3.org/2001/11/IsaViz/.

5 http://www.visualdataweb.org/relfinder.php.

6 http://wifo5-03.informatik.uni-mannheim.de/bizer/rdfapi/tutorial/netapi.html.

7 https://jena.apache.org/documentation/ontology/.

8 https://jena.apache.org/tutorials/rdf_api.html.

Table 1
Summary of approaches used in the selection step.

Approach	Domain		Ontology		LOD		Used datasets	
	Problem	Data mining	Complexity	Reasoning	Links	Semantics	LOD	Ontology
[27–32]	Persons, places, organizations	/	H	Yes	No	Yes	DBpedia, YAGO, Freebase, WordNet	Wikitology
[50]	Biology	/	L	No	No	No	/	/
[51]	Commerce	/	H	Yes	No	No	DBpedia	/
[53]	Medicine, publications	/	H	No	Yes	Yes	ClinicTrials.gov, BibBase[a]	/
[39]	Persons, places, organizations	/	H	No	No	Yes	DBpedia, YAGO, WordNet	/
[40]	Geography	/	H	No	No	No	YAGO, Freebase	/
[41]	Persons, places, organizations	/	H	No	Yes	Yes	Probase, DBpedia	/
[43–45]	Persons, places, organizations, music, movies	/	H	Yes	Yes	Yes	DBpedia, YAGO, MusicBrainz[b]	/
[46]	Books	/	H	No	No	Yes	YAGO	/

[a] http://data.bibbase.org/.
[b] http://linkedbrainz.org/.

headers, associates a concept from the Probase knowledge base with the table.

Zhang et al. [43,44] propose an incremental, bootstrapping approach that learns to label table columns using partial data in the column, and uses a generic feature model able to use various types of table context in learning. The work has been extended in [45], where the author shows that using sample selection techniques, it is possible to semantically annotate Web tables in a more efficient way.

Similarly, an approach for interpreting data from Web forms using LOD has been proposed [46]. The approach starts by extracting the attribute–value pairs of the form, which is done using probing methods. Then, the data extracted from the Web forms are represented as RDF triple, or complete RDF graph. To enrich the graph with semantics, it is aligned with a large reference ontology, like YAGO, using ontology alignment approaches.

A particular case are tables in Wikipedia, which follow a certain structure and, with links to other Wikipedia pages, can be more easily linked to existing LOD sources such as DBpedia. Therefore, several approaches for interpreting tables from Wikipedia with LOD have been proposed. Munoz et al. [47,48] propose methods for triplifying Wikipedia tables, called WikiTables, using existing LOD knowledge bases, like DBpedia and YAGO. Following the idea of the previous approaches, this approach starts by extracting entities from the tables, and then discovering existing relations between them. Similarly, a machine learning approach has been proposed by Bhagavatula et al. [49], where no LOD knowledge base is used, but only a metadata for the entities types and relations between them is added.

Similarly, approaches have been proposed for interpreting tabular data in spreadsheets [50,51], CSV [52], and XML [53].

5.3. Using LOD to interpret unstructured data

Text mining is the process of analyzing unstructured information, usually contained in a natural language text, in order to discover new patterns. Most common text mining tasks include text categorization, text clustering, sentiment analysis and others. In most cases, text documents contain named entities that can be identified in real world, and further information can be extracted about them. Several approaches and APIs have been proposed for extracting named entities from text documents and linking them to LOD. One of the most used APIs is DBpedia Spotlight [54,55], which allows for automatically annotating text documents with DBpedia URIs. This tool is used in several LOD enabled data mining approaches, e.g., [56–59]. Several APIs for extracting semantic richness from text exist, like Alchemy API,[9] OpenCalais API,[10] Textwise SemanticHacker API.[11] All these APIs are able to annotate named entities with concepts from several knowledge bases, like DBpedia, YAGO, and Freebase. These tools and APIs have been evaluated in the NERD framework, implemented by Rizzo et al. [60].

Furthermore, Linked Open Data is also heavily used for better understanding of social media, which unlike authored news and other textual Web content, social media data pose a number of new challenges for semantic technologies, due to their large-scale, noisy, irregular, and social nature. An overview of tools and approaches for semantic representation of social media streams is given in [61]. This survey discusses five key research questions: (i) What ontologies and Web of Data resources can be used to represent and reason about the semantics of social media streams? For example, FOAF[12] and GUMO ontology [62] for describing people and social network, SIOC[13] and DLPO ontology [63] for modeling and interlinking social media, MOAT [64] ontology for modeling tag semantics. (ii) How can semantic annotation methods capture the rich semantics implicit in social media? For example, keyphrase extraction [65,66], ontology-based entity recognition, event detection [67] and sentiment detection citegangemi2014frame, sentilo. (iii) How can we extract reliable information from these noisy, dynamic content streams? (iv) How can we model users' digital identity and social media activities? For example, discovering user demographics [68], deriving user interests [69] and capturing user behavior [70]. (v) What semantic-based information access methods can help address the complex information seeking behavior in social media? For example, semantic search in social media [71] and social media streams recommendation [72].

Once the user has developed a sufficient understanding of the domain, and the data mining task is defined, they need to select an appropriate data sample. If the data have already been mapped to appropriate domain specific ontologies or linked to external Linked Open Data, the users can more easily select a representative sample and/or meaningful subpopulation of the data for the given data mining task. For example, for a collection of texts, the user may decide to select those which mention a politician *after* the data has been linked to the semantic web, so that such a selection becomes possible.

Table 1 gives an overview of the discussed approaches in this section.[14] It can be observed that at the selection step, links between datasets play only a minor role, and reasoning is scarcely

[9] http://www.alchemyapi.com/api/.
[10] http://www.opencalais.com/documentation/opencalais-documentation.
[11] http://textwise.com/api.
[12] http://xmlns.com/foaf/spec/.
[13] http://sioc-project.org/.
[14] The tables used for summarizing approaches at the end of each section are structured as follows: The second column of the table states the problem domain on which the approach is applied. The third column states the data mining task/domain that was used in the approach. The next two columns capture the characteristics of

used. In most cases, general-purpose knowledge bases, such as DBpedia or YAGO, are used as sources of knowledge.

The selection of relevant semantic web datasets is usually done by *interlinking* a dataset at hand with data from Linked Open Data. There are strategies and tools for different kinds of data: relational databases are typically mapped to the semantic web using mapping rules and tools such as D2R. In those cases, mapping rules are typically written manually, which is easily possible because the schema of a relational database is usually explicitly defined.
Semi-structured data, such as Web tables, usually comes without explicit semantics, and in large quantities. Here, different heuristics and machine learning approaches are often applied to link them to LOD sources. For that case, it has been shown that combining approaches which perform schema and instance matching in a holistic way typically outperform approaches that handle both tasks in isolation.
For unstructured data, i.e., textual contents, the interlinking is typically done by linking named entities in the text to LOD sources with tools such as DBpedia Spotlight.
Once the interlinking is done, data visualization and summarization techniques can benefit from the additional knowledge contained in the interlinked datasets.

6. Preprocessing

Once the data is mapped to domain specific knowledge, the constraints expressed in the ontologies can be used to perform data validity checks and data cleaning. Ontologies can be used for detecting outliers and noise, as well as for handling missing values and data range and constraint violations, and guiding the users through custom preprocessing steps.

Ontologies are often used in many research approaches for the use of data cleaning and data preprocessing. Namely, there are two applications of ontologies in this stage: domain-independent ontologies used for data quality management, and domain ontologies. The first category of ontologies usually contains specifications for performing cleaning and preprocessing operations. In these approaches, the ontology is usually used to guide the user through the process of data cleaning and validation, by suggesting possible operations to be executed over the data. The second category of ontologies provides domain specific knowledge needed to validate and clean data, usually in an automatic manner.

6.1. Domain-independent approaches

One of the first approaches that uses a data quality ontology is proposed by Wang et al. [74]. They propose a framework called *OntoClean*[15] for ontology-based data cleaning. The core component of the framework is the data cleaning ontology component, which is used when identifying the cleaning problem and the relevant data. Within this component, the task ontology specifies the potential methods that may be suitable for meeting the user's goals, and the domain ontology includes all classes, instances, and axioms in a specific domain, which provides domain knowledge such as attribute constraints for checking invalid values during performing the cleaning tasks.

A similar approach is proposed by Perez et al. [75] with the *OntoDataClean* framework, which is able to guide the data cleaning process in a distributed environment. The framework uses a preprocessing ontology to store the information about the required transformations. First, the process of identifying and storing the required preprocessing steps has to be carried by a domain expert. Then, these transformations are needed to homogenize and integrate the records so they can be correctly analyzed or unified with other sources. Finally, the required information are stored in the preprocessing ontology, and the data transformations can be accomplished automatically. The approach has been tested on four databases in the domain of bio-medicine, showing that using the ontology the data can be correctly preprocessed and transformed according to the needs.

6.2. Domain-specific approaches

One of the first approaches to use a domain specific ontology is proposed by Philips et al. [76]. The approach uses ontologies to organize and represent knowledge about attributes and their constraints from relational databases. The approach is able to automatically, or semi-automatically with an assist of the user, identify the domains of the attributes, relations between the attributes, duplicate attributes and duplicate entries in the database.

Kedad et al. [77] propose a method for dealing with semantic heterogeneity during the process of data cleaning when integrating data from multiple sources, which is differences in terminologies. The proposed solution is based on linguistic knowledge provided by a domain is-a ontology. The main idea is to automatically generate correspondence assertions between instances of objects based on the is-a hierarchy, where the user can specify the level of accuracy expressed using the domain ontology. Once the user has specified the level of accuracy, two concepts will be considered the same if there is a subsumption relation between them, or both belong to the same class. Using this approach the number of results might be increased when querying the data, e.g., for the query "Do red cars have more accidents than others?" the system will not only look for *red cars*, but also for cars with color *ruby*, *vermilion*, and *seville*, which are subclasses of the red color.

Milano et al. introduce the OXC framework [78] that allows data cleaning on XML documents based on a uniform representation of domain knowledge through an ontology, which is gathered from domain analysis activities and from the DTDs of the documents. The framework comprises a methodology for data quality assessment and cleaning based on the reference ontology, and an architecture for XML data cleaning based on such methodology. Given a domain ontology, a mapping relation between the DTD and the ontology is defined, which is used to define quality dimensions (accuracy, completeness, consistency and currency), and perform data quality improvement by relying on the semantics encoded by the ontology.

Brüggemann et al. [79] propose a combination of domain specific ontologies and data quality management ontologies, by annotating domain ontologies with data quality management specific metadata. The authors have shown that such hybrid approach is suitable for consistency checking, duplicate detection, and metadata management. The approach has been extended in [80], where correction suggestions are being generated for each detected inconsistency. The approach uses the hierarchical structure of the

the ontologies used in the approach, i.e., the complexity level of the ontology, and if reasoning is applied on the ontology. Based on a prior categorization of ontologies presented in [73], we distinguish two degrees of ontology complexity: ontologies of low complexity that consist of class hierarchies and subclass relations (marked with *L*), and ontologies with high complexity that also contain relations other than the subclass relations, and further constraints, rules and so on (marked with *H*). The sixth column indicates if links (such as *owl:sameAs*) to other LOD sources were followed to extract additional information. The next column states whether explicit semantic information were used from a given LOD source. The final two columns list the used LOD sources and shared ontologies, respectively. If a LOD source is used, the respective ontology is used as well, without explicitly stating that in the table.

[15] Not to be confused with the ontology engineering method by Guarino and Welty.

ontology to offer the user semantically related context-aware correction suggestions. Moreover, the framework uses several measurements of semantic distances in ontologies to find the most suitable corrections for the identified inconsistencies. Based on those metrics the system can offer several suggestions for value corrections, i.e., value of next-sibling, first-child and parent. The approach has been applied on data from the cancer registry of Lower Saxony,[16] showing that it can successfully support domain experts.

Wang et al. [81] present a density-based outlier detecting method using domain ontology, named *ODSDDO* (Outlier Detecting for Short Documents using Domain Ontology). The algorithm is based on the *local outlier factor* algorithm, and uses domain ontology to calculate the semantic distance between short documents which improves the outlier detecting precision. To calculate the semantic similarity between two documents, first each word from each document is mapped to the corresponding concept in the ontology. Then, using the ontology concept tree, the similarity between each pair of concepts is calculated. The distance between two documents is then simply calculated as average of the sum of the maximum similarities between the pairs of concepts. The documents that have small or zero semantic similarity to other documents in the dataset are considered to be outliers.

Lukaszewski [82] propose an approach to admit and utilize noisy data by enabling to model different levels of knowledge granularity both in training and testing examples. The authors argue that erroneous or missing attribute values may be introduced by users of a system that are required to provide very specific values, but the level of their knowledge of the domain is too general to precisely describe the observation by the appropriate value of an attribute. Therefore, they propose knowledge representation that uses hierarchies of sets of attribute values, derived from subsumption hierarchies of concepts from an ontology, which decreases the level of attribute-noise in the data.

Fürber and Hepp [83–86] propose approaches for using Semantic Web technologies and Linked Open Data to reduce the effort for data quality management in relational databases. They show that using LOD reference data can help identifying missing values, illegal values, and functional dependency violations. In their first work [83], the authors describe how to identify and classify data quality problems in relational databases, through the use of SPARQL Inferencing Notation (SPIN).[17] SPIN is a Semantic Web vocabulary and processing framework that facilitates the representation of rules based on the syntax of the SPARQL protocol and RDF query language. To apply the approach on relational databases, the D2RQ tool [26] is used to extract data from relational databases into an RDF representation. The framework allows domain experts to define data requirements for their data based on forms as part of the data quality management process. The SPIN framework then automatically identifies requirement violations in data instances, i.e. syntactic errors, missing values, unique values violations, out or range values, and functional dependency violations. This approach is extended in [85] to assess the quality state of data in additional dimensions.

In a further work [84], instead of manually defining the data validation rules, the authors propose using Linked Open Data as trusted knowledge base that already contains information on the data dependencies. This approach has been shown to significantly reduce the effort for data quality management, when reference data is available in the LOD cloud. The approach was evaluated against a local knowledge base that contained manually created address data. Using GeoNames as a reference LOD dataset, the approach was able to identify invalid city entries, and invalid city–country relations.

A similar approach using SPIN, has been developed by Moss et al. [87] for assessing medical data. The system compromises a set of ontologies that support reasoning in a medical domain, such as human psychology, medical domain, and patient data. To perform the data cleaning, several rules for checking missing data points and value checking were used. The approach is evaluated on data from the Brain-IT network,[18] showing that it is able to identify invalid values in the data. Ontologies are often used in the healthcare domain for data quality management and data cleaning. Literature review of such papers is presented in [88].

In [89] we have developed an approach for filling missing values in a local table using LOD, which is implemented in a system named *Mannheim Search Joins Engine*.[19] The system relies on a large data corpus, crawled from over one million different websites. Besides two large quasi-relational datasets, the data corpus includes the *Billion Triples Challenge 2014 Dataset*[20] [90], and the *WebDataCommons Microdata Dataset*[21] [91]. For a given local table, the engine searches the data corpus for additional data for the attributes of the entities in the input table. To perform the search, the engine uses the existing information in the table, i.e. the entities' labels, the attributes' headers, and the attributes' data types. The discovered data is usually retrieved from multiple sources, therefore the new data is first consolidated using schema matching and data fusion methods. Then, the discovered data is used to fill the missing values in the local table. Also, the same approach can be used for validating the existing data in the given table i.e. outlier detection, noise detection and correction.

Table 2 gives an overview of the discussed approaches in this section. We can observe that, while ontologies are frequently used for data cleaning, well-known LOD datasets like DBpedia are scarcely exploited. Furthermore, many approaches have been tailored to and evaluated in the medical domain, likely because quite a few sophisticated ontologies exist in that domain.

Ontologies and Semantic Web data help with preprocessing the data, mostly for increasing the data quality. There are various data quality dimensions that can be addressed. Outliers and false values may be found by identifying data points and values that violate constraints defined in those ontologies. Subsumption hierarchies and semantic relations help unifying synonyms and detecting interrelations between attributes. Finally, missing values can be inferred and/or filled from LOD datasets.

7. Transformation

At this stage, the generation of better data for the data mining process is prepared. The transformation step includes dimensionality reduction, feature generation and feature selection, instance sampling, and attribute transformation, such as discretization of numerical data, aggregation, functional transformations, etc. In the context of Semantic Web enabled data mining, *feature generation* and *feature selection* are particularly relevant.

16 http://www.krebsregister-niedersachsen.de.
17 http://spinrdf.org/.
18 http://www.brain-it.eu/.
19 http://searchjoins.webdatacommons.org/.
20 http://km.aifb.kit.edu/projects/btc-2014/.
21 http://webdatacommons.org/structureddata/.

Table 2
Summary of approaches used in the preprocessing step.

Approach	Domain		Ontology		LOD		Used datasets	
	Problem	Data mining	Complexity	Reasoning	Links	Semantics	LOD	Ontology
[74]	Geography	/	H	No	No	No	/	OntoClean ontology
[75]	Biomedicine	/	H	No	No	No	/	OntoDataClean ontology
[77]	Medicine	/	H	No	No	No	/	Custom ontology
[77]	Medicine	/	H	No	No	No	/	Custom ontology
[78]	/	/	H	No	No	No	/	Custom ontology
[79,80]	Medicine	/	H	Yes	No	No	/	Custom ontology
[81]	Social media	Outlier detection	H	No	No	No	/	Custom ontology
[83–86]	Geography	/	H	Yes	No	Yes	DBpedia, GeoNames[a]	/
[89]	Geography, companies, movies, books, music, persons, drugs	/	H	No	Yes	Yes	BTC 2014, WebDataCommons Microdata Dataset	/
[87]	Medicine	/	H	No	No	No	/	Custom ontology

[a] http://sws.geonames.org/.

7.1. Feature generation

Linked Open Data has been recognized as a valuable source of background knowledge in many data mining tasks. Augmenting a dataset with features taken from Linked Open Data can, in many cases, improve the results of a data mining problem at hand, while externalizing the cost of creating and maintaining that background knowledge [92].

Most data mining algorithms work with a propositional *feature vector* representation of the data, i.e., each instance is represented as a vector of features $\langle f_1, f_2, \ldots, f_n \rangle$, where the features are either binary (i.e., $f_i \in \{true, false\}$), numerical (i.e., $f_i \in \mathbb{R}$), or nominal (i.e., $f_i \in S$, where S is a finite set of symbols) [93]. Linked Open Data, however, comes in the form of *graphs*, connecting resources with types and relations, backed by a schema or ontology.

Thus, for accessing Linked Open Data with existing data mining tools, transformations have to be performed, which create propositional features from the graphs in Linked Open Data, i.e., a process called *propositionalization* [94]. Usually, binary features (e.g., `true` if a type or relation exists, `false` otherwise) or numerical features (e.g., counting the number of relations of a certain type) are used. Furthermore, elementary numerical or nominal features (such as the population of a city or the production studio of a movie) can be added [95]. Other variants, e.g., computing the fraction of relations of a certain type, are possible, but rarely used.

In the recent past, a few approaches for propositionalizing Linked Open Data for data mining purposes have been proposed. Many of those approaches are supervised, i.e., they let the user formulate SPARQL queries, which means that they leave the propositionalization strategy up to the user, and a fully automatic feature generation is not possible. Usually, the resulting features are binary, or numerical aggregates using `SPARQL COUNT` constructs.

LiDDM [96] is an integrated system for data mining on the Semantic Web. The tool allows the users to declare SPARQL queries for retrieving features from LOD that can be used in different machine learning techniques, such as clustering and classification. Furthermore the tool offers operators for integrating data from multiple sources, data filtering and data segmentation, which are carried manually by the user. The usefulness of the tool has been presented through two use cases, using DBpedia, World FactBook[22] and LinkedMDB,[23] in the application of correlations analysis and rule learning.

A similar approach has been used in the RapidMiner[24] semweb plugin [97], which preprocesses RDF data in a way that it can be further processed by a data mining tool, RapidMiner in that case. Again, the user has to specify a SPARQL query to select the data of interest, which is then converted into feature vectors. The authors propose two methods for handling set-values data, by mapping them into an N-dimensional vector space. The first one is *FastMap*, which embeds points in an N-dimensional space based on a distance metric, much like Multidimensional Scaling (MDS). The second one is Correspondence Analysis (CA), which maps values to a new space based on their cooccurrence with values of other attributes. The approaches were evaluated on IMDB data,[25] showing that the mapping functions can improve the results over the baseline.

Cheng et al. [98] propose an approach for automated feature generation after the user has specified the type of features. To do so, the users have to specify the SPARQL query, which makes this approach supervised. The approach has been evaluated in the domain of recommender systems (movies domain) and text classification (tweets classification). The results show that using semantic features can improve the results of the learning models compared to using only standard features.

Mynarz et al. [99] have considered using user specified SPARQL queries in combination with SPARQL aggregates, including `COUNT`, `SUM`, `MIN`, `MAX`. Kauppinen et al. have developed the SPARQL package for R[26] [100,101], which allows importing LOD data in the very well known environment for statistical computing and graphics *R*. In their research they use the tool to perform statistical analysis and visualization of the linked Brazilian Amazon rainforest data. The same tool has been used in [102] for statistical analysis in piracy attack reports data. Moreover, they use the tool to import RDF data from multiple LOD sources in the environment of *R*, which allows them to easily analyze, interpret and visualize the discovered patterns in the data.

FeGeLOD [95] was the first fully automatic approach for enriching data with features that are derived from LOD. In that work, we have proposed six different feature generation strategies, allowing for both binary features and simple numerical aggregates. The first two strategies are only concerned with the instances themselves, i.e., retrieving the data properties of each entity, and the types of the entity. The four other strategies consider the relation of the instances to other resources in the graph, i.e. incoming and outgoing relations, and *qualified relations*, i.e., aggregates over the type of both the relation and the related entity. The work has been continued into the RapidMiner Linked Open Data extension[27] [103,104]. Currently, the RapidMiner LOD

22 http://wifo5-03.informatik.uni-mannheim.de/factbook/.
23 http://www.linkedmdb.org/.
24 http://www.rapidminer.com/.
25 http://www.imdb.com/.
26 http://linkedscience.org/tools/sparql-package-for-r/.
27 http://dws.informatik.uni-mannheim.de/en/research/rapidminer-lod-extension/.

extension supports the user in all steps of the LOD enabled knowledge discovery process. i.e. linking, combining data from multiple LOD sources, preprocessing and cleaning, transformation, data analysis, and interpretation of data mining findings.

FeGeLOD and the RapidMiner LOD extension have been used in different data mining applications, i.e., text classification [58,57,56,105], explaining statistics [106–108], linkage error detection [109], and recommender systems [110,111]. Besides using simple binary and numerical representation of the features, we have proposed using adapted versions of TF–IDF based measures. In [112] we have performed an initial comparison of different propositionalization strategies (i.e., binary, count, relative count and TF–IDF) for generating features from types and relations from Linked Open Data.

A problem similar to feature generation is addressed by *Kernel functions*, which compute the distance between two data instances. The similarity is calculated by counting common substructures in the graphs of the instances, e.g., walks, paths and threes. The graph kernels are used in kernel-based data mining and machine learning algorithms, most commonly support vector machines (SVMs), but can also be exploited for tasks such as clustering. In the past, many graph kernels have been proposed that are tailored towards specific application [113–115], or towards specific semantic representation [116–119]. But only a few approaches are general enough to be applied on any given RDF data, regardless of the data mining task. Lösch et al. [120] introduce two general RDF graph kernels, based on intersection graphs and intersection trees. First, they propose the use of walk and path kernels, which count the number of walks and paths in the intersected graphs. Then, they propose full subtree kernel, which counts the number of full subtrees of the intersection tree.

The intersection tree path kernel introduced by Lösch et al., has been modified and simplified by Vries et al. [121–124], which also allows for explicit calculation of the instances' feature vectors, instead of pairwise similarities. Computing the feature vectors significantly improves the computation time, and allows using any arbitrary machine learning methods. They have developed two types of kernels over RDF data, RDF walk count kernel and RDF WL sub tree kernel. The RDF walk count kernel counts the different walks in the sub-graphs (up to the provided graph depth) around the instances nodes. The RDF WL sub tree kernel counts the different full sub-trees in the sub-graphs (up to the provided graph depth) around the instances nodes, using the Weisfeiler–Lehman algorithm [125]. The approaches developed by Lösch et al. and by Vries et al. have been evaluated on two common relational learning tasks: entity classification and link prediction.

7.2. Feature selection

We have shown that there are several approaches that generate propositional feature vectors from Linked Open Data. Often, the resulting feature spaces can have a very high dimensionality, which leads to problems both with respect to the performance as well as the accuracy of learning algorithms. Thus, it is necessary to apply some feature selection approaches to reduce the feature space. Additionally, for datasets that already have a high dimensionality, background knowledge from LOD or linguistic resources such as WordNet may help reducing the feature space better than standard techniques which do not exploit such background knowledge.

Feature selection is a very important and well studied problem in the literature. The objective is to identify features that are correlated with or predictive of the class label. Generally, all feature selection methods can be divided into two broader categories: wrapper methods and filter methods (John et al. [126] and Blum et al. [127]).

In feature vectors generated from external knowledge we can often observe relations between the features. In many cases those relations are hierarchical relations, or we can say that the features subsume each other, and carry similar semantic information. Those hierarchical relations can be easily retrieved from the ontology or schema used for publishing the LOD, and can be used to perform better feature selection.

We have introduced an approach [128] that exploits hierarchies for feature selection in combination with standard metrics, such as *information gain* or *correlation*. The core idea of the approach is to identify features with similar relevance, and select the most valuable abstract features, i.e. features from as high as possible levels of the hierarchy, without losing predictive power, and thus, find and optimal trade-off between the predictive power and the generality of a feature in order to avoid over-fitting. To measure the similarity of relevance between two nodes, we use the standard correlation and information gain measure. The approach works in two steps, i.e., an initial selection and an additional pruning step.

Jeong et al. [129] propose the *TSEL* method using a semantic hierarchy of features based on WordNet relations. The presented algorithm tries to find the most representative and most effective features from the complete feature space. To do so, they select one representative feature from each path in the tree, where path is the set of nodes between each leaf node and the root, based on the *lift* measure, and use χ^2 to select the most effective features from the reduced feature space.

Wang et al. [130] propose a *bottom-up hill climbing* search algorithm to find an optimal subset of concepts for document representation. For each feature in the initial feature space, they use a kNN classifier to detect the k nearest neighbors of each instance in the training dataset, and then use the purity of those instances to assign scores to features.

Lu et al. [131] describe a *greedy top-down* search strategy for feature selection in a hierarchical feature space. The algorithm starts with defining all possible paths from each leaf node to the root node of the hierarchy. The nodes of each path are sorted in descending order based on the nodes' information gain ratio. Then, a greedy-based strategy is used to prune the sorted lists. Specifically, it iteratively removes the first element in the list and adds it to the list of selected features. Then, removes all ascendants and descendants of this element in the sorted list. Therefore, the selected features list can be interpreted as a mixture of concepts from different levels of the hierarchy.

When creating features from multiple LOD sources, often a single semantic feature can be found in multiple LOD source represented with different properties. For example, the area of a country in DBpedia is represented with *db:areaTotal*, and with *yago:hasArea* in YAGO. The problem of aligning properties, as well as instances and classes, in ontologies is addressed by *ontology matching* techniques [132]. Even though there exist a vast amount of work in the area of ontology matching, most of the approaches for generating features from Linked Open Data are not explicitly addressing this problem. The RapidMiner LOD extension offers an operator for matching properties extracted from multiple LOD sources, which are later fused into single feature. The operator is based on the probabilistic algorithm for ontology matching PARIS [133]. Unlike most other systems, PARIS is able to align both entities and relations. It does so by bootstrapping an alignment from the matching literals and propagating evidence based on relation functionalities. In [104] we have shown that, for example, the value for the population of a country can be found in 10 different sources within the LOD cloud, which using the RapidMiner LOD extension matching and fusion operator were merged into a single feature. Such a fusion can provide a feature that mitigates missing values and single errors for individual sources, leading to only one high-value feature.

Table 3
Summary of approaches used in the transformation step.

Approach	Domain		Ontology		LOD		Used datasets	
	Problem	Data mining	Complexity	Reasoning	Links	Semantics	LOD	Ontology
[96]	Government, economy, movies	Correlation, association mining	H	No	Yes	No	DBpedia, World FactBook,[a] LinkedMDB	/
[97]	Movies	Classification	H	No	Yes	No	DBpedia, LinkedMDB	/
[98]	Movies, social media	Recommender systems, classification	H	No	Yes	No	YAGO	/
[100,101]	Geography	Correlations	L	No	Yes	Yes	Linked Brazilian Amazon Rainforest, DBpedia	/
[95]	Biology, sociology, economy	Classification, regression	H	No	Yes	Yes	DBpedia	/
[104]	Economy, publications	Correlations	H	No	Yes	Yes	DBpedia, YAGO, LinkedGeoData,[b] Eurostat,[c] GeoNames, WHO,[d] Linked Energy Data,[e] OpenCyc,[f] World Factbook	/
[56]	News	Sentiment analysis	H	No	Yes	No	DBpedia	/
[109]	Music, movies, books	Linkage error detection	H	No	Yes	Yes	DBpedia, DBTropes,[g] Peel Sessions[h]	/
[121,122]	Publications, geology	Property value prediction, link prediction	H	No	No	No	Custom dataset, British Geological Survey[i]	SWRC[j]
[123]	Bio-medicine, publications	Classification	H	No	Yes	Yes	MUTAG, ENZYMES, Semantic Web Conference Corpus,[k] British Geological Survey	/
[129]	News	Text classification	H	No	No	No	WordNet	/
[130]	Biomedicine	Text classification	H	No	No	No	/	UMLS
[131]	Pharmacology	Classification	H	No	No	No	/	NDF-RT[l]
[134]	Commerce	Rule learning	H	No	No	No	/	Products ontology
[135,136]	Movies	Association rules	H	No	No	No	/	Custom ontology
[139]	Medicine	Association mining	H	Yes	No	No	/	UMLS[m]
[137,138]	Accident reports	Text mining, rule learning	H	No	No	No	/	Custom ontology

[a] http://wifo5-03.informatik.uni-mannheim.de/factbook/.
[b] http://linkedgeodata.org.
[c] http://eurostat.linked-statistics.org/ and http://wifo5-03.informatik.unimannheim.de/eurostat/.
[d] http://gho.aksw.org/.
[e] http://en.openei.org/lod/.
[f] http://sw.opencyc.org/.
[g] http://skipforward.opendfki.de/wiki/DBTropes.
[h] http://dbtune.org/bbc/peel/.
[i] http://www.bgs.ac.uk/opengeoscience/.
[j] http://ontoware.org/swrc/.
[k] http://data.semanticweb.org/.
[l] http://www.nlm.nih.gov/research/umls/sourcereleasedocs/current/NDFRT/.
[m] http://www.nlm.nih.gov/research/umls/.

In pattern mining and association rule mining, domain ontologies are often used to reduce the feature space in order to get more meaningful and interesting patterns. In the approach proposed by Bellandi et al. [134] several domain-specific and user-defined constraints are used, i.e., pruning constraints, used to filter uninteresting items, and abstraction constraints permitting the generalization of items towards ontology concepts. The data is first preprocessed according to the constraints extracted from the ontology, and then, the data mining step takes place. Applying the pruning constraints excludes the information that the user is not interested in, before applying the data mining approach.

Onto4AR is a constraint-based algorithm for association mining proposed by Antunes [135] and revised later in [136], where taxonomical and non-taxonomical constraints are defined over an item ontology. This approach is interesting in the way that the ontology offers a high level of expression for the constraints, which allows to perform the knowledge discovery at the optimal level of abstraction, without the need for user input. Garcia et al. developed a technique called *Knowledge Cohesion* [137,138] to extract more meaningful association rules. The proposed metric is based on semantic distance, which measures how close two items are semantically based within the ontology, where each type of relation is weighted differently.

7.3. Other

Zeman et al. [139] present the Ferda DataMiner tool, which is focused on the data transformation step. In this approach the ontologies are used for two purposes: construction of adequate attribute categorization, and identification and exploitation of semantically related attributes. The authors claim that ontologies can be

efficiently used for categorization of attributes as higher-level semantics could be assigned to individual values. For example, for blood pressure there are predefined values that divide the domain in a meaningful way: say, blood pressure above 140/90 mm Hg is considered as hypertension. For the second purpose, ontologies are used to discover the relatedness between the attributes, which can be exploited so as to meaningfully arrange the corresponding data attributes in the data transformation phase.

Table 3 gives an overview of the discussed approaches in this section. It can be observed that at this stage of the data mining process, many approaches also exploit links between LOD datasets to identify more features. On the other hand, the features are most often generated without regarding the schema of the data, which is, in most cases, rather used for post processing of the features, e.g., for feature selection. Likewise, reasoning is only scarcely used.

Most data mining algorithms and tools require a *propositional* representation, i.e., feature vectors for instances. Typical approaches for propositionalization are, e.g., adding all numerical datatype properties as numerical features, or adding all direct types as binary features. There are unsupervised and supervised methods, where for the latter, the user specifies a query for features to generate—those are useful if the user knows the LOD dataset at hand and/or has an idea which features could be valuable. While such classic propositionalization methods create human interpretable features and thus are also applicable for *descriptive* data mining, kernel methods often deliver better *predictive* results, but at the price of losing the interpretability of those results.

A crucial problem when creating explicit features from Linked Open Data is the scalability and the number of features generated. Since only few approaches focus on identifying high value features already at the generation step, combining feature generation with feature subset selection is clearly advised.

The schema information for the LOD sources, such as type hierarchies, can be exploited for feature space reduction. There are a few algorithms exploiting the schema, which often provide a better trade-off between feature space reduction and predictive performance than schema-agnostic approaches.

8. Data mining

After the data is selected, preprocessed and transformed in the most suitable representation, the next step is choosing the appropriate data mining task and data mining algorithm. Depending on the KDD goals, and the previous steps of the process, the users need to decide which type of data mining to use, i.e. classification, regression, clustering, summarization, or outlier detection. Understanding the domain will assist in determining what kind of information is needed from the KDD process, which makes it easier for the users to make a decision. There are two broader categories of goals in data mining: prediction and description. Prediction is often referred to as supervised data mining, which attempts to forecast the possible future or unknown values of data elements. On the other hand, descriptive data mining is referred as unsupervised data mining, which seeks to discover interpretable patterns in the data. After the strategy is selected, the most appropriate data mining algorithm should be selected. This step includes selecting methods to search patterns in the data, and deciding on specific models and parameters of the methods.

Once the data mining method and algorithm are selected, the data mining takes place.

To the best of our knowledge, there are rarely any approaches in the literature that incorporates data published as Linked Open Data into the data mining algorithms themselves. However, many approaches are using ontologies for the data mining process, not to only support the user in the stage of selecting the data mining methods, but to guide the users through the whole process of knowledge discovery.

8.1. Domain-independent approaches

While there is no universally established data mining ontology yet, there are several data mining ontologies currently under development, such as the Knowledge Discovery (KD) Ontology [140], the KDDONTO Ontology [141], the Data Mining Workflow (DMWF) Ontology[28] [142], the Data Mining Optimization (DMOP) Ontology[29] by Hilario [143,144], OntoDM[30] [145,146], and its sub ontology modules OntoDT,[31] OntoDM-core[32] [147] and OntoDM-KDD[33] [148].

An overview of existing intelligent assistants for data analysis that use ontologies is given in [149]. In this survey, all approaches are categorized by several criteria. First, which types of support the intelligent assistants offer to the data analyst. Second, it surveys the kinds of background knowledge that the IDAs rely on in order to provide the support. Finally, it performs thorough comparison of IDAs in light of the defined dimensions and the identification of limitations and missing features.

One of the earliest approaches, *CAMLET*, was proposed by Suyama et al. [150], which uses two light-weight ontologies of machine learning entities to support the automatic composition of inductive learning systems.

Among the first prototypes is the *Intelligent Discovery Assistant* proposed by Bernstein et al. [151], which provides users with systematic enumerations of valid sequences of data mining operators. The tool is able to determine the characteristics of the data and of the desired mining result, and uses an ontology to search for and enumerate the KDD processes that are valid for producing the desired result from the given data. Also, the tool assists the user in selecting the processes to execute, by ranking the processes according to what is important to the user. A light-weight ontology is used that contains only a hierarchy of data mining operators divided into three main classes: preprocessing operators, induction algorithms and post processing operators.

Many approaches are using Semantic Web technologies to assist the user in building complex data mining workflows. Žáková et al. [152,140] proposed an approach for semiautomatic workflow generation that requires only the user input and the user desired output to generate complete data mining workflows. To implement the approach, the authors have developed the knowledge discovery ontology, which gives a formal representation of a knowledge types and data mining algorithms. Second, a planning algorithm is implemented that assembles workflows based on the planning task descriptions extracted from the knowledge discovery ontology and the given user's input–output task requirements. In such semiautomatic environment the user is not required to be aware of the numerous properties of the wide range of relevant data mining algorithms. In their later work, the methodology is implemented in the Orange4WS environment for service-oriented data mining [153,154].

Diamantini et al. [155] introduce a semantic based, service-oriented framework for tools sharing and reuse, giving advanced support for the semantic enrichment through semantic annotation

28 http://www.e-lico.eu/dmwf.html.
29 http://www.e-lico.eu/DMOP.html.
30 http://www.ontodm.com/doku.php.
31 http://www.ontodm.com/doku.php?id=ontodt.
32 http://www.ontodm.com/doku.php?id=ontodm-core.
33 http://www.ontodm.com/doku.php?id=ontodm-kdd.

of KDD tools, deployment of the tools as web services and discovery and use of such services. To support the system an ontology named KDDONTO [141] is used, which represents a formal ontology describing the domain of KDD algorithms. The ontology provides information required by the KDD composer to assist them to choose the suitable algorithms for achieving their goal starting from the data at hand, and to correctly compose the algorithms for forming a valid process [156].

Kietz et al. [142,157] presented a data mining ontology for workflow planning, able to effectively organize hundreds of operators, which is the base for checking the correctness of KDD workflows and a Hierarchical Task Network planning component able to effectively enumerate useful KDD workflows. This includes the objects manipulated by the system, the metadata needed, the operators used, and a goal description. The workflow generator is tightly coupled with a meta-miner whose role is to rank the workflows and is based on the DMOP ontology. Furthermore, the authors introduced the eProPlan tool [158], which represents ontology-based environment for planning KDD workflows. Later on, the tool is used to semantically annotate all operators in the very well known data mining tool RapidMiner. This allows the users to easily, and faster build more efficient KDD workflows within RapidMiner [159]. Their evaluation showed that using Semantic Web technologies can speed up the workflow design time up to 80%. This is achieved by automatic suggestion for possible operations in all phases of the KDD process.

Furthermore, Hilario et al. [143] present the data mining optimization ontology, which provides a unified conceptual framework for analyzing data mining tasks, algorithms, models, datasets, workflows and performance metrics, as well as their relationships. One of the main goals of the ontology is to support meta-mining of complete data mining experiments in order to extract workflow patterns [144]. In addition, the authors have developed a knowledge base by defining instances of the DMOP ontology. The DMOP ontology is not based on any upper-level ontology and uses a large set of customized special-purpose relations.

Panov et al. [145,146] propose an ontology of data mining *OntoDM* that includes formal definitions of basic data mining entities, such as *datatype* and *dataset*, *data mining task* and *data mining algorithm*, which is based on the proposal for a general framework for data mining proposed by Džeroski [160]. The ontology is one of the first deep/heavy-weight ontology for data mining. To allow the representation of mining structured data, the authors developed a separate ontology module, named OntoDT, for representing the knowledge about datatypes. To represent core data mining entities, and to be general enough to represent the mining of structured data, the authors introduced the second ontology module called OntoDM-core [147]. The third, and final, module of the ontology is the OntoDM-KDD which is used for representing data mining investigations [148].

Gabriel et al. [161] propose the usage of semantic information about the attributes contained in a dataset for learning classification rules that are potentially better understandable. They use *semantic coherence*, i.e., the semantic proximity of attributes used in a rule, as a target criterion to increase the understandability of a rule. In their paper, they show that using WordNet as a source of knowledge, and adapting a standard separate-and-conquer rule learning algorithm [162], they can significantly increase the semantic coherence in a rule without a decrease in classification accuracy.

8.2. Domain-specific approaches

Santos et al. [163] describes a research of an ontological approach for leveraging the semantic content of ontologies to improve knowledge discovery in databases. The authors divide the KDD process into three main operations, and try to support each of them using ontologies. First, in the data understanding and data preparation phases, ontologies can facilitate the integration of heterogeneous data and guide the selection of relevant data to be mined, regarding domain objectives. Second, during the modeling phase, domain knowledge allows the specification of constraints for guiding data mining algorithms by narrowing the search space. Finally, in the interpretation and evaluation phase, domain knowledge helps experts to validate and rank the extracted patterns.

Ding et al. [164,165] introduce another ontology based framework for incorporating domain knowledge into data mining process. The framework is able to support the data mining process in several steps of the pipeline: data exploration, defining mining goals, data selection, data preprocessing and feature selection, data transformation, data mining algorithm parameter selection, and data mining results evaluation.

Češpivová et al. [166] have shown how ontologies and background knowledge can aid each step of the KDD process. They perform association mining using the LISp-Miner tool, over the STULONG medical dataset. To support the data mining they use UMLS ontologies[34] to map the data to semantic concepts. The mapping helped the authors to better understand the domain. They were able to identify and filter out redundant and unnecessary attributes, and group together semantically related attributes, by analyzing the relationships inside the ontology. Furthermore, they use ontologies to interpret and to give better explanation of the data mining results.

Table 4 gives an overview of the discussed approaches in this section. It shows that, while Linked Open Data based datasets play a minor role in this step, heavy-weight ontologies and reasoning are quite frequently used. Moreover, most of the ontologies are domain-independent, while domain-specific developments at this step are rather rare.

Ontologies are often used for supporting the user in creating a proper data mining process. They can be used to represent data sources, algorithms etc. in data mining processes, and assist the user in building reasonable data mining processes, e.g., by ensuring that a chosen algorithm is capable of handling the given data.
For example, the platform *RapidMiner* internally uses semantic descriptions of operators to assist the user in avoiding errors, e.g., when combining data preprocessing and machine learning operators. Here, reasoning does not only check the validity of a process, but also proposes solutions to fix an invalid process.
Approaches that use semantic information directly in an algorithm to influence the outcome of that algorithm are rather rare. There are some directions of using semantic background knowledge in data mining algorithms, e.g., for finding patterns that are easier to consume by an end user.

9. Interpretation

After the data mining step has been applied, we expect to discover some hidden patterns from the data. To be interpreted and understood, these patterns often require the use of some background knowledge, which is not always straightforward to find. In most real world contexts, providing the background knowledge is committed to the experts, whose work is to analyze the results of a data mining process, give them a meaning and refine them. The interpretation turns out to be an intensive and time-consuming process, where part of knowledge can remain unrevealed or unexplained.

[34] http://www.nlm.nih.gov/research/umls/.

Table 4
Summary of approaches used in the data mining step.

Approach	Domain		Ontology		LOD		Used datasets	
	Problem	Data mining	Complexity	Reasoning	Links	Semantics	LOD	Ontology
[151]	Commerce	Classification, regression, neural networks	L	No	No	No	/	Custom ontology
[152,140,153, 154]	Genomics, engineering,	Classification	H	Yes	No	No	/	KD
[155,141,156]	/	/	H	Yes	No	No	/	KDDONTO
[142,157–159]	Socio-economy	Clustering,	H	Yes	No	No	/	DMO, DMWF, DMOP
[143,144]	Medicine	Classification	H	Yes	No	No	/	DMO, DMOP
[145,146,148, 147]	Chemistry, pharmacology	Classification, regression	H	Yes	No	No	/	OntoDM, OntoDT, OntoDM-core, OntoDM-KDD
[161]	/	Classification rule learning	L	No	No	No	/	WordNet
[163]	/	/	L	No	No	No	/	Custom ontology
[164,165]	/	/	L	No	No	No	/	Custom ontology

Explain-a-LOD [106] is one of the first approaches in the literature for automatically generating hypothesis for explaining statistics by using LOD. The tool uses FeGeLOD (described in Section 7.1) for enhancing statistical datasets with background information from DBpedia, and uses correlation analysis and rule learning for producing hypothesis which are presented to the user. The tool has been later used to find and explain hypothesis for quality of living in cities across the world [107], and unemployment rates in France [108], among others. For example, in [107] the tool was able to automatically discover hypothesis like "Cities where many things take place have a high quality of living." and "European capitals of culture have a high quality of living.". While in [108] where data from DBpedia, Eurostat and LinkedGeoData has been used, the tool discovered hypothesis like "Regions in France that have high energy consumption have low unemployment rate.änd "French regions that are out of Europe, French African Islands, and French Islands in the Indian Ocean have high unemployment rate.".Furthermore, the approach is extended in [167], which allows automatic correlation analysis and visualizing statistical data on maps using Linked Open Data. The tool allows the users to import any statistical data from local spreadsheets or RDF datacubes, perform correlation analysis and automatically visualize the findings on a map.

Linked Open Data can not only add categorical information which allows for an easier exploration of results, but also additional visual clues. In [108,167], we have shown that polygon data for geographic entities published as LOD, like GADM[35] can be exploited for creating a map-based visualization of data mining results. Moreover, GADM offers shape data of geographical entities on different administrative levels, which can be accessed through DBpedia by following *owl:sameAs* links.

d'Aquin et al. [168] have proposed a method that exploits external information available as LOD to support the interpretation of data mining results, through automatically building a navigation–exploration structure in the results of a particular type of data mining, in this case sequential pattern mining, tool based on data dimensions chosen by the analyst. To do so, the authors first represent the data mining results into a way compatible with a LOD representation, and link them to existing LOD sources. Then, the analyst can easily explore the mined results with additional dimension. Furthermore, to organize the enriched results into a hierarchy, the authors use formal concept analysis to build a concept lattice. This can allow the analyst to drill down into the details of a sub-set of the patterns, and see how it relates to the original data.

A similar approach is used in [169] for interpreting sequential patterns in patient data. Linked Data is used to support the interpretation of patterns mined from patient care trajectories. Linked data exposed through the BioPortal system is used to create a navigation structure within the patterns obtained form sequential pattern mining. The approach provides a flexible way to explore data about trajectories of diagnoses and treatments according to different medical classifications.

Tiddi [170] proposes a three step approach for interpreting data mining results, i.e. clustering, association rules and sequence patterns. In the first step additional information for the patterns results are extracted from the LOD cloud. Using inductive logic programming, new hypotheses are generated from the pattern mining results and the knowledge extracted from LOD. In the last step the hypotheses are evaluated using ranking strategies, like Weighted Relative Accuracy, and Information Retrieval F-measure. The same approach has been used in [171] to explain why groups of books, obtained from a clustering process, have been borrowed by the same students. The analysis were done on the Huddersfield's books usage dataset,[36] using the British National Bibliography[37] and Library of Congress[38] as LOD datasets. The experiments lead to interesting hypothesis to explain the clusters, e.g., "books borrowed by students of Music Technologies are clustered together because they talk about music".

The work has been continued in [172,173], introducing *Dedalo*, framework that dynamically traverses Linked Data to find commonalities that form explanations for items of a cluster. Dedalo uses iterative approach for traversing LOD graphs, where the roots are the items of the clusters. The underlying assumption is that items that belong to one cluster share more common paths in the LOD graph, than the items outside the cluster. The authors were able to extract interesting and representative explanation for the clusters, however, the number of resulting atomic rules is rather large, and need to be aggregated in a post-processing step. The typical strategy to overcome those problems is providing the patterns to human experts, whose role consists in analyzing the results, discovering the interesting ones while explaining, pruning or refining the unclear ones. To cope with such a strenuous and time consuming process, the authors in their next work [174] have proposed an approach that is using Neural Network model to predict whether two rules, if combined, can lead to the creation of a new, improved rule (i.e., a new rule, with a better F-measure). The approach has been applied in the domain of education and publications.

Lavrač et al. have made a notable research work in the field of semantic subgroup discovery. The task of subgroup discovery is

35 http://gadm.geovocab.org/.

36 http://library.hud.ac.uk/data/usagedata/readme.html.

37 http://bnb.data.bl.uk/.

38 http://id.loc.gov/.

defined as follows: "Given a population of individuals and a property of those individuals that we are interested in, find population subgroups that are statistically most interesting, for example, are as large as possible and have the most unusual statistical (distributional) characteristics with respect to the property of interest" [175]. The authors define semantic subgroup discovery as part of semantic data mining, which is defined as: "Given a set of domain ontologies, and empirical data annotated by domain ontology terms, one can find a hypothesis (a predictive model or a set of descriptive patterns), expressed by domain ontology terms, explaining the given empirical data". The semantic subgroup discovery was first implemented in the SEGS system [176]. SEGS uses as background knowledge data from three publicly available, semantically annotated biological data repositories. Based on the background knowledge, it automatically formulates biological hypotheses: rules which define groups of differentially expressed genes. Finally, it estimates the relevance (or significance) of the automatically formulated hypotheses on experimental microarray data. The system was extended in the SegMine system, which allows exploratory analysis of microarray data, performed through semantic subgroup discovery by SEGS [177], followed by link discovery and visualization by Biomine [178], an integrated annotated bioinformatics information resource of interlinked data. The SEGS system was later extended to two general semantic subgroup discovery systems, SDM-SEGS and SDM-Aleph [179–181]. Finally, the authors introduced the Hedwig system [182], which overcomes some of the limitations of the previous systems. The findings of this series of work have been concluded in [183,184].

A similar problem is addressed in [185]. Instead of identifying subgroups, we aim at finding special characteristics of a given instance, given a contrast set. To that end, data about the instance at hand, as well as its contrast set, are retrieved from DBpedia. Attribute-wise outlier detection, which computes outlier scores for single attribute values [186], is exploited for identifying those attribute values of the instance that are significantly different from those of the other instances.

Many approaches are using ontologies for patterns post-mining, and interpretation of the results. The domain knowledge and metadata specification stored in the ontology are used in the interpretation stage to prune and filter out discovered patterns. Ontologies are commonly used to filter out redundant patterns, and too specific patterns without losing semantic information. One of the first approaches that use domain ontologies for that purpose is the work by Srikant [187], who introduced the concept of generalized association rules. Similarly, Zhou et al. [188] introduce the concept of raising. Raising is the operation of generalizing data mining rules in order to increase the support while keeping the confidence high enough. This is done with generalizing the entities by raising them to a specified level in the ontology. The authors use an ontology that consists of two taxonomies, one of which describes different customer classifications, while the other one contains a large hierarchy, based on Yahoo, which contains interests. In the experiments, the support values of rule sets were greatly increased, up to 40 times. GART is a very similar approach [189], which uses several taxonomies over attributes to iteratively generalize rules, and then, prunes redundant rules at each step. The experiments were performed using a sale database of a Brazilian supermarket. The experiments show reduction rates of the sets of association rules varying from 14,61% to 50,11%. Marinica et al. [190] present an interactive postprocessing framework, called ARIPSO (Association Rule Interactive post-Processing using Schemas and Ontologies). The framework assists the user throughout the analyzing task to prune and filter discovered rules. The system allows formalizing user knowledge and goals, which are latter used in applying iteratively a set of filters over extracted rules in order to extract interesting rules: minimum improvement constraint filter, item-relatedness filter, rule schema filters/pruning. The experiments were performed on the Nantes Habitat data,[39] dealing with customers satisfaction concerning accommodation, for which a corresponding ontology was developed by the authors. The results showed that the number of rules can be significantly reduced when using the schema, resulting in more descriptive rules.

Huang et al. [191] use LOD to interpret the results of text mining. The approach starts with extracting entities and semantic relations between them from text documents, resulting into semantic graphs. Then, a frequent sub-graph discovery algorithm is applied on the text graphs to find frequent patterns. To interpret the discovered subgraphs, an algorithm is proposed to traverse Linked Data graphs for relations that are used to annotate the vertices and the edges of the frequent sub-graphs. The approach is applied on a military dataset, where DBpedia is used as a LOD dataset.

Another approach that uses ontologies in rule mining is the 4ft-Miner tool [192]. The tool is used in four stages of the KDD process: data understanding, data mining, result interpretation and result dissemination. In the data understanding step, a data-to-ontology mapping was performed, which resulted in discovery of redundant attributes. In the data mining stage of the KDD process, the ontology was used to decompose the data mining task into more specific tasks, which can be run faster, resulting in more homogeneous results, and thus easily interpretable. In the interpretation stage, the data-to-ontology mappings are used to match some of the discovered associations to the corresponding semantic relations or their more complex chains from the ontology, which can be considered as potential explanation of the discovered associations. The approach was used to interpret associations in medical and social climate applications. In the medical domain, the STULONG dataset[40] is used, which contains cardiovascular risk data. As an ontology is used the UMLS ontology. Using the approach, the authors were able to discover hypothesis like "Patients who are not physically active within the job nor after the job (Antecedent) will more often have higher blood pressure (Succedent)" and "Increase of smoking leads to increase of cardiovascular diseases".

Table 5 gives an overview of the discussed approaches in this section. We observe that in this step, reasoning plays no crucial role. The datasets exploited are rather mixed, general purpose datasets such as DBpedia are often used, but also highly specific datasets can be exploited. Roughly half of the approaches also make use of the interlinks between those datasets.

> Semantic Web data can help in the interpretation of patterns found, in particular for descriptive tasks. Those typically encompass subgroups or clusters found, or rule models that are used to describe a dataset.
> The information used from LOD datasets and/or ontologies can help further analyzing those findings, e.g., by explicating typical features of instances in a subgroup or cluster, thus, they may explain the grouping chosen by a data mining algorithm. Furthermore, rules can be further refined and/or generalized, which improves their interpretability.

10. Example use case

Recommender systems have changed the way people find and buy products and services. As the Web has grown over time, and the number of products and services within, recommender

[39] http://www.nantes-habitat.fr/.
[40] http://euromise.vse.cz/stulong-en/.

Table 5
Summary of approaches used in the interpretation step.

Approach	Domain		Ontology		LOD		Used datasets	
	Problem	Data mining	Complexity	Reasoning	Links	Semantics	LOD	Ontology
[106]	Sociology, economy	Pattern mining	H	No	Yes	Yes	DBpedia	/
[107,108,167]	Statistics	Pattern mining	H	No	Yes	Yes	DBpedia, Eurostat, LinkedGeoData, GADM	/
[168,169]	Students, medicine	Pattern mining	H	No	Yes	Yes	Open University's course catalog,[a] ICD10, CCAM Bio Ontology[b]	/
[170,171]	Books, publications	Clustering, association rules	H	No	Yes	No	British National Bibliography,[c] Library of Congress[d]	/
[172–174]	Education, publication	Clustering	H	No	Yes	No	DBpedia, UIS,[e] British National Bibliography, Library of Congress	/
[176–181,184]	Biomedicine	Rule learning, subgroup discovery	H	No	No	No	/	GO,[f] KEGG,[g] Entrez
[185]	/	Outlier detection	H	No	No	No	DBpedia	/
[182]	Finance	Subgroup discovery	H	No	Yes	No	GeoNames	/
[188]	Commerce	Rule learning	H	No	No	No	/	Interest ontology (from Yahoo)
[189]	Commerce	Rule learning	H	No	No	No	/	Products taxonomy
[190]	Sociology	Rule learning	H	No	No	No	/	Custom ontology
[191]	Military	Subgroup discovery	H	No	Yes	Yes	DBpedia	/
[192]	Medicine, sociology	Association mining	H	Yes	No	No		UMLS, Social climate ontology

[a] http://data.open.ac.uk.
[b] http://sparql.bioontology.org/sparql/.
[c] http://bnb.data.bl.uk/.
[d] http://id.loc.gov/.
[e] http://uis.270a.info/.html.
[f] http://geneontology.org/.
[g] http://www.genome.jp/kegg/.

systems represent a powerful method for users to filter that large information and product space. With the introduction of the Linked Open Data recommender systems are emerging research area that extensively use Linked Open Data as background knowledge for extracting useful data mining features that could improve the recommendation results. It has been shown that Linked Open Data can improve recommender systems towards a better understanding and representation of user preferences, item features, and contextual signs they deal with. LOD has been used in content-based, collaborative, and hybrid techniques, in various recommendation tasks, i.e., rating prediction, Top-N recommendations, cross-domain recommendation and diversity in content-based recommendations.

Therefore, in this section, we show an example use-case of LOD-enabled knowledge discovery process in the domain of recommender systems. Through this example we will describe each step of the LOD-enabled KDD process, i.e., linking the local data to LOD dataset, combining data from multiple LOD datasets, transformation of the data, building recommender system, and interpretation of the results. In this example, we will use the dataset used in the Linked Open Data-enabled Recommender Systems Challenge at ESWC 2014 [193].

10.1. Linking local data to LOD

The first step of the LOD-enabled KDD pipeline is to link the data to the corresponding LOD concepts from the chosen LOD dataset (cf. Section 5). The initial dataset contained a list of books retrieved from the LibraryThing dataset,[41] together with user ratings for books. To be able to build LOD-enabled recommenders the datasets were linked to DBpedia. To do so, the label and the production year of the books are used to find the corresponding book entity in DBpedia, by using the following SPARQL query [194]:

```
SELECT DISTINCT ?movie ?label ?year WHERE {
?movie rdf:type dbpedia-owl:Film.
?movie rdfs:label ?label.
?movie dcterms:subject ?cat .
?cat rdfs:label ?year .
FILTER langMatches(lang(?label), "EN") .
FILTER regex(?year, "^[0-9]{4} film", "i")
}
ORDER BY ?label
```

This results in a dataset of books with the corresponding DBpedia URIs, and user ratings. We see here that, instead of a general purpose tool, we use a hand-crafted linkage rule which exploits a certain amount of non-formalized domain knowledge (e.g., there may be different films with the same title, but we are able to tell them apart by their production year). As explicated in Section 5, this is a common strategy for structured knowledge sources such as relational databases.

41 http://www.macle.nl/tud/LT.

10.2. Combining multiple LOD datasets

The second step is to explore the initial links to extract additional data from other LOD datasets that might be useful for the given task (cf. Section 6). Besides DBpedia there are multiple other datasets in the LOD cloud that contain information about books. To extract the corresponding entity URIs from the other datasets, we can follow the existing *owl:sameAs* links in DBpedia. For example, we can extract the URIs from the corresponding entities in YAGO and Freebase. Furthermore, we can find the corresponding URIs to other datasets for which an *owl:sameAs* does not exist, like dbTropes, using the book title and production year. In [110], we use the book ISBN and title to link the books to the RDF Book Mashup dataset,[42] which provides the average score assigned to a book on Amazon.

In the third step of the pipeline the data gathered from different sources need to be consolidated into one cleansed dataset. However, when combining data from different LOD sources, those usually use different schemas. For example, the book author in DBpedia is listed under the *dbpprop:author* property, while in YAGO the same information is under the *created* property. In order to use that data more effectively, such attributes can be merged into one by applying schema matching. For example, in [104] for that purpose we use the *PARIS* ontology matching approach. The resulting dataset will contain high quality and extensive information about the books.

As discussed in Section 6, this shows how Semantic Web data can help creating more valuable data, e.g., by fusing similar information from various sources to increase both the coverage and reduce the redundancy of attributes in the dataset.

10.3. Building LOD-based recommender system

In the fourth step the graph data needs to be transformed to propositional form so it could be used in a standard recommender system (cf. Section 7). For that purpose in [110] we use the RapidMiner LOD extension. In this approach we developed a hybrid multi-strategy content-based recommendation system. This approach builds on training individual base recommenders and using global popularity scores as generic recommenders. The results of the individual recommenders are combined using stacking regression and rank aggregation.

For building the content-based recommenders, we use the following feature sets for describing a book retrieved from DBpedia and the RDF Book Mashup dataset:

- All *direct types*, i.e., `rdf:type`, of a book[43]
- All *categories of a book*
- All *categories of a book including broader categories*[44]
- All *categories of a book's author(s)*
- All *categories of a book's author(s) and of all other books* by the book's authors
- All *genres of a book* and of all other books by the book's authors
- All *authors that influenced or were influenced* by the book's authors
- A bag of words created from the *abstract* of the book in DBpedia. That bag of words is preprocessed by tokenization, stemming, removing tokens with less than three characters, and removing all tokens less frequent than 3% or more frequent than 80%.

This feature creation strategy is a mix of automatic and manual feature generation. On the one hand, we automatically create all direct types, without caring about whether they are useful for the task at hand or not. Most of the other features, however, are guided by domain knowledge and assumptions, e.g., that the categories and genres of a book may be relevant for a book recommender system. As discussed in Section 7, fully automatic feature generation covers all the possible features, at the danger of creating a very high dimensional feature space with many irrelevant features. Thus, combinations of automatic and hand-crafted feature generation strategies, like in this example, are rather common in practice.

The content-based recommender system is based on the k-NN algorithm, where we use $k = 80$ and cosine similarity for the base recommenders. The rationale of using cosine similarity is that, unlike, e.g., Euclidean distance, only common features influence the similarity, but not common absence of features (e.g., two books *not* being American Thriller Novels).

10.4. Recommender results interpretation

The final step of the LOD-enabled KDD pipeline is the evaluation and interpretation of the developed data mining model (cf. Section 9). In the case of recommender systems, besides being able to efficiently produce accurate recommendations, the ability to effectively explain the recommendations to users is another important aspect of a recommender system. To this aim, Linked Open Data plays an important role since it eases the computation of a human-understandable explanation because it allows the user to explore the results space following different dimensions, i.e., explicitly listing, for each property, the values which are common between the movies in the user profile and the suggested ones [194].

Such an approach is particularly interesting if, unlike the use case above, the recommendation is based purely on statistical methods like collaborative filtering. For example, for a user who has already liked the book "The Lord of the Rings", a recommender system might recommend the book "The Hobbit". The system can then easily give an explanation to why the book was recommended to the user by displaying the most important shared relations for these two books, e.g., both books are "High fantasy", both books are written by the same author "J. R. R. Tolkien", and both books belong to the same category "British fantasy novels".

It is important to note that the interpretation is really an a posteriori step here, since the recommender system was purely based on statistical measures, i.e., finding the most similar books, without providing any explanations by itself.

11. Discussion

Given the amount of research works discussed in this paper, it is evident that, especially with the advent and growth of Linked Open Data, information from the Semantic Web can be used beneficially in the data mining and knowledge discovery process. Looking at the results from a larger distance, however, we can make a few interesting observations:

- DBpedia is used in the vast majority of the research papers discussed in this survey, with other LOD sources being used only scarcely, and the majority of the hundreds of LOD datasets not being used at all. There may be different reasons for that; ranging from DBpedia's relatively simple data model and its wide coverage to the availability of sophisticated tools such as DBpedia Lookup and DBpedia Spotlight.
  While this underlines the utility of such general purpose knowledge sources on the Semantic Web, it can also be problematic to tailor and evaluate approaches only to single datasets, since it limits the insights on the general applicability of the approaches.

42 http://wifo5-03.informatik.uni-mannheim.de/bizer/bookmashup/.

43 This includes types in the YAGO ontology, which can be quite specific (e.g., *American Thriller Novels*).

44 The reason for not including broader categories by default is that the category graph is not a cycle-free tree, with some subsumptions being rather questionable.

- Many papers use custom ontologies and datasets instead of reusing open datasets from the web of data. This is particularly often observed in the life sciences and medical domain, which, at the same time, is one of the largest most prominently represented domains within the Linked Open Data cloud. It is subject to future research to find out the reasons for this discrepancy, which may have different reasons, such as a limited awareness of open datasets, or an inferior fitness for use of those datasets.
- Links between datasets, which are one of the core ingredients to *Linked* Open Data, are used by relatively few approaches. This may also imply that many of the approaches stay below what is possible with Linked Open Data, leveraging only information from one dataset instead of using the full amount of knowledge captured in the Semantic Web. One reason may be that even in the presence of machine-interpretable schemas, developing schema-agnostic applications is a non-trivial task. Furthermore, building approaches that autonomously follow links and are ultimately capable of exploiting the whole Web of Linked Data as background knowledge would also lead to new scalability challenges.
- Expressive schemas/ontologies and reasoning on those, which has been a core selling point of the Semantic Web for years, are rarely combined with data mining and knowledge discovery. Again, it is subject to future research to find out whether this is due to a limited availability of suitable ontologies, limited awareness, or imperfect fitness to the problems found in practice.
- In most cases, knowledge from the Semantic Web is about the domain of the processed data, not the data mining domain. However, given endpoints such as *myExperiment.org*,[45] which provides lots of scientific workflows (including data mining workflows), would allow for solutions providing advice to data analysts building such workflows, such as the recently announced "Wisdom of Crowds Operator Recommendations" by RapidMiner,[46] based on open data.

These observations show that, although a remarkable amount of work in the area exists, data mining and knowledge discovery is still not tapping the full potential that is provided by the Semantic Web. Data mining workflows automatically leveraging information from different datasets by following links beyond single datasets such as DBpedia are still an interesting and promising area of research.

12. Conclusion and outlook

In this paper, we have given a survey on the usage of Semantic Web data, most prominently Linked Open Data, for data mining and knowledge discovery. Following Fayyad's classic workflow pipeline, we have shown examples for the usage of Semantic Web data at every stage of the pipeline, as well as approaches supporting the full pipeline.

Analyzing the findings from the survey, the first observation is that there are plenty of works of research in the area, and applications exist in many domains. A frequent application domain is biomedicine and life science, but the approaches are also transferred to quite a few other domains. Furthermore, some sophisticated applications and tool stacks exist, that go beyond mere research prototypes.

Furthermore, we see that there are still some uncharted territories in the research landscape of Semantic Web enabled data mining. This shows that, although impressive results can be achieved already today, the full potential of Semantic Web enabled data mining and KDD still remains to be unlocked.

[45] http://www.myexperiment.org.

[46] https://rapidminer.com/news-posts/rapidminer-makes-snap-move-predictive-analytics-data-mining-machine-learning-cloud/.

Acknowledgment

The work presented in this paper has been partly funded by the German Research Foundation (DFG) under grant number PA 2373/1-1 (Mine@LOD).

References

[1] U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, Advances in knowledge disc overy and data mining, in: American Association for Artificial Intelligence, Menlo Park, CA, USA, 1996, pp. 1–34. URL http://dl.acm.org/citation.cfm?id=257938.257942.
[2] D. Hand, H. Mannila, P. Smyth, Principles of Data Mining, MIT Press, 2001.
[3] C. Bizer, T. Heath, T. Berners-Lee, Linked data—the story so far, Int. J. Semant. Web Inform. Syst. 5 (3) (2009) 1–22.
[4] M. Schmachtenberg, C. Bizer, H. Paulheim, Adoption of the Linked Data Best Practices in Different Topical Domains, in: International Semantic Web Conference, 2014.
[5] G. Stumme, A. Hotho, B. Berendt, Semantic web mining—state of the art and future directions, J. Web Semant. 4 (2) (2006) 124–143. URL http://www.kde.cs.uni-kassel.de/hotho/pub/2006/JWS2006SemanticWebMining.pdf.
[6] K. Sridevi, D.R. UmaRani, A survey of semantic based solutions to web mining, Int. J. Emerging Trends Technol. Comput. Sci.
[7] Q.K. Quboa, M. Saraee, A state-of-the-art survey on semantic web mining, Intell. Inf. Manage. 5 (2013) 10.
[8] J. Sivakumar, et al., A review on semantic-based web mining and its applications.
[9] D. Dou, H. Wang, H. Liu, Semantic data mining: A survey of ontology-based approaches, in: Semantic Computing (ICSC), 2015 IEEE International Conference on, IEEE, 2015, pp. 244–251.
[10] V. Tresp, M. Bundschus, A. Rettinger, Y. Huang, Towards machine learning on the seman tic web, in: Uncertainty Reasoning for the Semantic Web i, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 282–314. http://dx.doi.org/10.1007/978-3-540-89765-1_17.
[11] A. Rettinger, U. Lsch, V. Tresp, C. dAmato, N. Fanizzi, Mining the semantic web, Data Min. Knowl. Discov. 24 (3) (2012) 613–662. http://dx.doi.org/10.1007/s10618-012-0253-2.
[12] T.R. Gruber, Toward principles for the design of ontologies used for knowledge sharing, Int. J. Hum.-Comput. Stud. 43 (5) (1995) 907–928.
[13] H.O. Nigro, S.G. Cisaro, D.H. Xodo, Data Mining With Ontologies: Implementations, Findings and Frameworks, in: Information Science Reference, Imprint of: IGI Publishing, Hershey, PA, 2007.
[14] A.-S. Dadzie, M. Rowe, Approaches to visualising linked data: A survey, Semant. Web 2 (2) (2011) 89–124.
[15] G. Tummarello, R. Cyganiak, M. Catasta, S. Danielczyk, R. Delbru, S. Decker, Sig. ma: Live views on the web of data, Web Semant.: Sci. Serv. Agents World Wide Web 8 (4) (2010) 355–364.
[16] D. Huynh, S. Mazzocchi, D. Karger, Piggy bank: Experience the semantic web inside your web browser, in: The Semantic Web–ISWC 2005, Springer, 2005, pp. 413–430.
[17] T. Hastrup, R. Cyganiak, U. Bojars, Browsing linked data with fenfire.
[18] O. Peña, U. Aguilera, D. López-de Ipiña, Linked open data visualization revisited: A survey, Semant. Web J.
[19] B. Mutlu, P. Hoefler, G. Tschinkel, E. Veas, V. Sabol, F. Stegmaier, M. Granitzer, Suggesting visualisations for published data, Proc. IVAPP (2014) 267–275.
[20] G.A. Atemezing, R. Troncy, Towards a linked-data based visualization wizard, in: Workshop on Consuming Linked Data, 2014.
[21] J.M. Brunetti, S. Auer, R. García, The linked data visualization model, in: International Semantic Web Conference, Posters & Demos, 2012.
[22] J. Klímek, J. Helmich, M. Neasky, Application of the linked data visualization model on real world data from the czech lod cloud, in: 6th International Workshop on the Linked Data on the Web, LDOW'14, 2014.
[23] J. Unbehauen, S. Hellmann, S. Auer, C. Stadler, Knowledge extraction from structured sources, in: S. Ceri, M. Brambilla (Eds.), Search Computing, in: Lecture Notes in Computer Science, vol. 7538, Springer, Berlin Heidelberg, 2012, pp. 34–52. http://dx.doi.org/10.1007/978-3-642-34213-4_3.
[24] S.S. Sahoo, W. Halb, S. Hellmann, K. Idehen, T.T. Jr., S. Auer, J. Sequeda, A. Ezzat, A survey of current approaches for mapping of relational databases to rdf (01 2009). URL http://www.w3.org/2005/Incubator/rdb2rdf/RDB2RDF_SurveyReport.pdf.
[25] D.-E. Spanos, P. Stavrou, N. Mitrou, Bringing relational databases into the semantic web: A survey, Semant. Web 3 (2) (2012) 169–209. http://dx.doi.org/10.3233/SW-2011-0055.
[26] C. Bizer, D2rq—treating non-rdf databases as virtual rdf graphs, in: Proceedings of the 3rd International Semantic Web Conference, 2004.
[27] V. Mulwad, T. Finin, Z. Syed, A. Joshi, Using linked data to interpret tables, in: Proc. 1st Int. Workshop on Consuming Linked Data, 2010.
[28] V. Mulwad, T. Finin, Z. Syed, A. Joshi, T2LD: interpreting and representing tables as linked data, in: Proceedings of the ISWC 2010 Posters & Demonstrations Track: Collected Abstracts, Shanghai, China, November 9, 2010, 2010. URL http://ceur-ws.org/Vol-658/paper489.pdf.

[29] Z. Syed, T. Finin, V. Mulwad, A. Joshi, Exploiting a web of semantic data for interpreting tables, in: Proceedings of the Second Web Science Conference, 2010.

[30] V. Mulwad, DC proposal: Graphical models and probabilistic reasoning for generating linked data from tables, in: l. Aroyo, et al. (Eds.), Proceedings of Tenth International Semantic Web Conference, Part II, in: LCNS, in: LCNS, 7032, Springer-Verlag, 2011, pp. 317–324. submitted at the Doctoral Consortium.

[31] V. Mulwad, T. Finin, A. Joshi, Semantic message passing for generating linked data from tables, in: Proceedings of the 12th International Semantic Web Conference, Springer, 2013.

[32] V. Mulwad, T. Finin, A. Joshi, Interpreting medical tables as linked data to generate meta-analysis reports, in: Proceedings of the 15th IEEE International Conference on Information Reuse and Integration, IEEE Computer Society, 2014.

[33] T. Finin, Z. Syed, Creating and exploiting a web of semantic data, in: ICAART, vol. 1, 2010, pp. 7–18.

[34] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, Z. Ives, Dbpedia: A nucleus for a web of open data, in: Proceedings of the 6th International The Semantic Web and 2Nd Asian Conference on Asian Semantic Web Conference, ISWC'07/ASWC'07, Springer-Verlag, Berlin, Heidelberg, 2007, pp. 722–735. URL http://dl.acm.org/citation.cfm?id=1785162.1785216.

[35] F.M. Suchanek, G. Kasneci, G. Weikum, Yago: A core of semantic knowledge, in: Proceedings of the 16th International Conference on World Wide Web, WWW'07, ACM, New York, NY, USA, 2007, pp. 697–706. http://dx.doi.org/10.1145/1242572.1242667. URL http://doi.acm.org/10.1145/1242572.1242667.

[36] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, J. Taylor, Freebase: A collaboratively created graph database for structuring human knowledge, in: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, SIGMOD'08, ACM, New York, NY, USA, 2008, pp. 1247–1250. http://dx.doi.org/10.1145/1376616.1376746. URL http://doi.acm.org/10.1145/1376616.1376746.

[37] G.A. Miller, Wordnet: a lexical database for english, Commun. ACM 38 (11) (1995) 39–41.

[38] H. Liu, Towards semantic data mining, in: In Proc. of the 9th International Semantic Web Conference, ISWC2010, 2010.

[39] G. Limaye, S. Sarawagi, S. Chakrabarti, Annotating and searching web tables using entities, types and relationships, Proc. VLDB Endow. 3 (1–2) (2010) 1338–1347. http://dx.doi.org/10.14778/1920841.1921005.

[40] P. Venetis, A. Halevy, J. Madhavan, M. Paşca, W. Shen, F. Wu, G. Miao, C. Wu, Recovering semantics of tables on the web, Proc. VLDB Endow. 4 (9) (2011) 528–538. http://dx.doi.org/10.14778/2002938.2002939.

[41] J. Wang, H. Wang, Z. Wang, K.Q. Zhu, Understanding tables on the web, in: Proceedings of the 31st International Conference on Conceptual Modeling, ER'12, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 141–155. http://dx.doi.org/10.1007/978-3-642-34002-4_11.

[42] W. Wu, H. Li, H. Wang, K.Q. Zhu, Probase: A probabilistic taxonomy for text understanding, in: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, SIGMOD'12, ACM, New York, NY, USA, 2012, pp. 481–492. http://dx.doi.org/10.1145/2213836.2213891. URL http://doi.acm.org/10.1145/2213836.2213891.

[43] Z. Zhang, A.L. Gentile, I. Augenstein, Linked data as background knowledge for information extraction on the web, SIGWEB Newsl. (Summer) (2014) 5:1–5:9 http://dx.doi.org/10.1145/2641730.2641735. URL http://doi.acm.org/10.1145/2641730.2641735.

[44] Z. Zhang, Start small, build complete: Effective and efficient semantic table interpretation using tableminer, Semat. Web J. Under transparent review.

[45] Z. Zhang, Learning with partial data for semantic table interpretation, in: K. Janowicz, S. Schlobach, P. Lambrix, E. Hyvnen (Eds.), Knowledge Engineering and Knowledge Management, in: Lecture Notes in Computer Science, vol. 8876, Springer International Publishing, 2014, pp. 607–618. http://dx.doi.org/10.1007/978-3-319-13704-9_45.

[46] M. Oita, A. Amarilli, P. Senellart, Cross-fertilizing deep web analysis and ontology enrichment, in: M. Brambilla, S. Ceri, T. Furche, G. Gottlob (Eds.), VLDS,, pp. 5–8.

[47] E. Muoz, A. Hogan, A. Mileo, Triplifying wikipedia's tables, in: LD4IE@ ISWC'13, 2013, pp. 1–1.

[48] E. Muñoz, A. Hogan, A. Mileo, Using linked data to mine rdf from wikipedia's tables, in: Proceedings of the 7th ACM International Conference on Web Search and Data Mining, WSDM'14, ACM, New York, NY, USA, 2014, pp. 533–542. http://dx.doi.org/10.1145/2556195.2556266. URL http://doi.acm.org/10.1145/2556195.2556266.

[49] C.S. Bhagavatula, T. Noraset, D. Downey, Methods for exploring and mining tables on wikipedia, in: Proceedings of the ACM SIGKDD Workshop on Interactive Data Exploration and Analytics, IDEA'13, ACM, New York, NY, USA, 2013, pp. 18–26. http://dx.doi.org/10.1145/2501511.2501516. URL http://doi.acm.org/10.1145/2501511.2501516.

[50] L. Han, T. Finin, C. Parr, J. Sachs, A. Joshi, Rdf123: From spreadsheets to rdf, in: Proceedings of the 7th International Conference on The Semantic Web, ISWC'08, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 451–466. http://dx.doi.org/10.1007/978-3-540-88564-1_29.

[51] A. Langegger, W. Wö, Xlwrap querying and integrating arbitrary spreadsheets with sparql, in: A. Bernstein, D. Karger, T. Heath, L. Feigenbaum, D. Maynard, E. Motta, K. Thirunarayan (Eds.), The Semantic Web - ISWC 2009, in: Lecture Notes in Computer Science, vol. 5823, Springer, Berlin, Heidelberg, 2009, pp. 359–374. http://dx.doi.org/10.1007/978-3-642-04930-9_23.

[52] L. Ding, D. DiFranzo, A. Graves, J. Michaelis, X. Li, D.L. McGuinness, J.A. Hendler, in: M. Rappa, P. Jones, J. Freire, S. Chakrabarti (Eds.), Twc Data-gov Corpus: Incrementally Generating Linked Government Data from data.gov, WWW, ACM, 2010, pp. 1383–1386. URL http://dblp.uni-trier.de/db/conf/www/www2010.html#DingDGMLMH10.

[53] O. Hassanzadeh, S.H. Yeganeh, R.J. Miller, Linking semistructured data on the web, in: WebDB, 2011.

[54] P.N. Mendes, M. Jakob, A. García-Silva, C. Bizer, Dbpedia spotlight: Shedding light on the web of documents, in: Proceedings of the 7th International Conference on Semantic Systems, I-Semantics'11, ACM, New York, NY, USA, 2011, pp. 1–8. http://dx.doi.org/10.1145/2063518.2063519. URL http://doi.acm.org/10.1145/2063518.2063519.

[55] J. Daiber, M. Jakob, C. Hokamp, P.N. Mendes, Improving efficiency and accuracy in multilingual entity extraction, in: Proceedings of the 9th International Conference on Semantic Systems, ACM, 2013, pp. 121–124.

[56] O. De Clercq, S. Hertling, V. Hoste, S.P. Ponzetto, H. Paulheim, Identifying disputed topics in the news, in: Linked Data for Knowledge Discovery, LD4KD, CEUR, 2014, pp. 37–48.

[57] A. Schulz, P. Ristoski, H. Paulheim, I see a car crash: Real-time detection of small scale incidents in microblogs, in: The Semantic Web: ESWC 2013 Satellite Events, Springer, 2013, pp. 22–33.

[58] D. Hienert, D. Wegener, H. Paulheim, Automatic classification and relationship extraction for multi-lingual and multi-granular events from wikipedia, in: Detection, Representation, and Exploitation of Events in the Semantic Web, DeRiVE 2012, 902, 2012, pp. 1–10.

[59] M. Schuhmacher, S.P. Ponzetto, Exploiting dbpedia for web search results clustering, in: Proceedings of the 2013 Workshop on Automated Knowledge Base Construction, ACM, 2013, pp. 91–96.

[60] G. Rizzo, R. Troncy, Nerd: a framework for unifying named entity recognition and disambiguation extraction tools, in: Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, 2012, pp. 73–76.

[61] K. Bontcheva, D. Rout, Making sense of social media streams through semantics: a survey, Semant. Web 1 (2012) 1–31.

[62] D. Heckmann, T. Schwartz, B. Brandherm, M. Schmitz, M. von, Wilamowitz-Moellendorff, Gumo—the general user model ontology, in: User Modeling 2005, Springer, 2005, pp. 428–432.

[63] S. Scerri, K. Cortis, I. Rivera, S. Handschuh, Knowledge discovery in distributed social web sharing activities, in: Making Sense of Microposts, # MSM2012, 2012, pp. 26–33.

[64] A. Passant, P. Laublet, Meaning of a tag: A collaborative approach to bridge the gap between tagging and linked data, in: LDOW, 2008.

[65] G. Solskinnsbakk, J.A. Gulla, Semantic annotation from social data, in: Proceedings of the Fourth International Workshop on Social Data on the Web Workshop, 2011.

[66] L. Qu, C. Müller, I. Gurevych, Using tag semantic network for keyphrase extraction in blogs, in: Proceedings of the 17th ACM Conference on Information and Knowledge Management, ACM, 2008, pp. 1381–1382.

[67] J. Eisenstein, D.H. Chau, A. Kittur, E.P. Xing, et al. Topicviz: Semantic navigation of document collections, arXiv preprint, arXiv:1110.6200.

[68] M. Pennacchiotti, A.-M. Popescu, A machine learning approach to twitter user classification, ICWSM 11 (2011) 281–288.

[69] F. Abel, Q. Gao, G.-J. Houben, K. Tao, Semantic enrichment of twitter posts for user profile construction on the social web, in: The Semanic Web: Research and Applications, Springer, 2011, pp. 375–389.

[70] J. Chan, C. Hayes, E. Daly, Decomposing discussion forums using common user roles.

[71] F. Abel, I. Celik, G.-J. Houben, P. Siehndel, Leveraging the semantics of tweets for adaptive faceted search on twitter, in: The Semantic Web–ISWC 2011, Springer, 2011, pp. 1–17.

[72] J. Chen, R. Nairn, L. Nelson, M. Bernstein, E. Chi, Short and tweet: experiments on recommending content from information streams, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, 2010, pp. 1185–1194.

[73] H. Paulheim, F. Probst, Ontology-enhanced user interfaces: A survey, Int. J. Semant. Web Inf. Syst. 6 (2) (2010) 36–59.

[74] X. Wang, H.J. Hamilton, Y. Bither, An Ontology-based Approach to Data Cleaning, 2005.

[75] D. Prez-Rey, A. Anguita, J. Crespo, Ontodataclean: Ontology-Based integration and preprocessing of distributed data, in: N. Maglaveras, I. Chouvarda, V. Koutkias, R.W. Brause (Eds.), ISBMDA, in: Lecture Notes in Computer Science, vol. 4345, Springer, 2006, pp. 262–272. URL http://dblp.uni-trier.de/db/conf/ismda/isbmda2006.html#Perez-ReyAC06.

[76] J. Phillips, B.G. Buchanan, Ontology-guided knowledge discovery in databases, in: Proceedings of the 1st International Conference on Knowledge Capture, ACM, 2001, pp. 123–130.

[77] Z. Kedad, E. Métais, Ontology-based data cleaning, in: Natural Language Processing and Information Systems, Springer, 2002, pp. 137–149.

[78] D. Milano, M. Scannapieco, T. Catarci, Using ontologies for xml data cleaning, in: Proceedings of the 2005 OTM Confederated International Conference on On the Move to Meaningful Internet Systems, OTM'05, Springer-Verlag, Berlin, Heidelberg, 2005, pp. 562–571. http://dx.doi.org/10.1007/11575863_75.
[79] S. Brüggemann, F. Grüning, Using domain knowledge provided by ontologies for improving data quality management, in: Proceedings of I-Know, 2008, pp. 251–258.
[80] S. Brüggemann, H.-J. Appelrath, Context-aware replacement operations for data cleaning, in: Proceedings of the 2011 ACM Symposium on Applied Computing, SAC'11, ACM, New York, NY, USA, 2011, pp. 1700–1704. http://dx.doi.org/10.1145/1982185.1982539.
URL http://doi.acm.org/10.1145/1982185.1982539.
[81] Y. Wang, S. Yang, Outlier detection from massive short documents using domain ontology, in: Intelligent Computing and Intelligent Systems, ICIS, in: 2010 IEEE International Conference on, vol. 3, IEEE, 2010, pp. 558–562.
[82] T. Lukaszewski, Extending bayesian classifier with ontological attributes.
[83] C. Fürber, M. Hepp, Using sparql and spin for data quality management on the semantic web, in: Business Information Systems, Springer, 2010, pp. 35–46.
[84] C. Fürber, M. Hepp, Using semantic web resources for data quality management, in: Proceedings of the 17th International Conference on Knowledge Engineering and Management by the Masses, EKAW'10, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 211–225.
URL http://dl.acm.org/citation.cfm?id=1948294.1948316.
[85] C. Fürber, M. Hepp, Swiqa-a semantic web information quality assessment framework, in: ECIS, vol. 15, 2011, p. 19.
[86] C. Fürber, M. Hepp, Using semantic web technologies for data quality management, in: Handbook of Data Quality, 2013, pp. 141–161.
[87] L. Moss, D. Corsar, I. Piper, A linked data approach to assessing medical data, in: Computer-Based Medical Systems, CBMS, in: 2012 25th International Symposium on, IEEE, 2012, pp. 1–4.
[88] S.-T. Liaw, A. Rahimi, P. Ray, J. Taggart, S. Dennis, S. de Lusignan, B. Jalaludin, A.E.T. Yeo, A. Talaei-Khoei, Towards an ontology for data quality in integrated chronic disease management: A realist review of the literature, I. J. Med. Inform. 82 (1) (2013) 10–24. URL http://dblp.uni-trier.de/db/journals/ijmi/ijmi82.html#LiawRRTDLJYT13.
[89] O. Lehmberg, D. Ritze, P. Ristoski, R. Meusel, H. Paulheim, C. Bizer, The mannheim search join engine, J. Web Semant.
[90] T. Käfer, A. Harth, Billion Triples Challenge data set, Downloaded from http://km.aifb.kit.edu/projects/btc-2014/, 2014.
[91] R. Meusel, P. Petrovski, C. Bizer, The WebDataCommons Microdata, RDFa and Microformat Dataset Series, in: Proc. of the 13th Int. Semantic Web Conference, ISWC14, 2014.
[92] H. Paulheim, Exploiting linked open data as background knowledge in data mining, in: Workshop on Data Mining on Linked Open Data, 2013.
[93] T. Pang-Ning, M. Steinbach, V. Kumar, Introduction to Data Mining, Pearson, 2006.
[94] S. Kramer, N. Lavra, P. Flach, Propositionalization approaches to relational data mining, in: S. DÅeroski, N. Lavra (Eds.), Relational Data Mining, Springer, Berlin, Heidelberg, 2001, pp. 262–291. http://dx.doi.org/10.1007/978-3-662-04599-2_11.
[95] H. Paulheim, J. Fürnkranz, Unsupervised Generation of Data Mining Features from Linked Open Data, in: International Conference on Web Intelligence, Mining, and Semantics (WIMS'12), 2012.
[96] V.N.P. Kappara, R. Ichise, O. Vyas, Liddm: A data mining system for linked data, in: Workshop on Linked Data on the Web, LDOW2011, 2011.
[97] M.A. Khan, G.A. Grimnes, A. Dengel, Two pre-processing operators for improved learning from semanticweb data, in: First RapidMiner Community Meeting And Conference, RCOMM 2010, 2010.
[98] W. Cheng, G. Kasneci, T. Graepel, D. Stern, R. Herbrich, Automated feature generation from structured knowledge, in: 20th ACM Conference on Information and Knowledge Management, CIKM 2011, 2011.
[99] J. Mynarz, V. Svátek, Towards a benchmark for lod-enhanced knowledge discovery from structured data, in: Proceedings of the Second International Workshop on Knowledge Discovery and Data Mining Meets Linked Open Data, CEUR-WS, 2013, pp. 41–48.
[100] T. Kauppinen, G.M. de Espindola, B. Gräler, Sharing and analyzing remote sensing observation data for linked science, in: Poster Proceedings of the Extended Semantic Web Conference, Citeseer, 2012.
[101] T. Kauppinen, G.M. de Espindola, J. Jones, A. Sánchez, B. Gräler, T. Bartoschek, Linked brazilian amazon rainforest data, Semant. Web 5 (2) (2014) 151–155.
[102] W. van Hage, M. van Erp, V. Malaisé, Linked open piracy: A story about e-science, linked data, and statistics, J. Data Semant. 1 (3) (2012) 187–201. http://dx.doi.org/10.1007/s13740-012-0009-6.
[103] H. Paulheim, P. Ristoski, E. Mitichkin, C. Bizer, Data mining with background knowledge from the web, in: RapidMiner World, 2014.
[104] P. Ristoski, C. Bizer, H. Paulheim, Mining the web of linked data with rapidminer, J. Web Semant.
[105] A. Schulz, C. Guckelsberger, F. Janssen, Semantic abstraction for generalization of tweet classification.
[106] H. Paulheim, Generating possible interpretations for statistics from linked open data, in: 9th Extended Semantic Web Conference, ESWC, 2012.
[107] H. Paulheim, Nobody wants to live in a cold city where no music has been recorded, in: The Semantic Web: ESWC 2012 Satellite Events, Springer, 2012, pp. 387–391.
[108] P. Ristoski, H. Paulheim, Analyzing statistics with background knowledge from linked open data, in: Workshop on Semantic Statistics, 2013.
[109] H. Paulheim, Identifying wrong links between datasets by multi-dimensional outlier detection, in: Workshop on Debugging Ontologies and Ontology Mappings, WoDOOM, 2014.
[110] P. Ristoski, E.L. Mencía, H. Paulheim, A hybrid multi-strategy recommender system using linked open data, in: Semantic Web Evaluation Challenge, Springer, 2014, pp. 150–156.
[111] M. Schmachtenberg, T. Strufe, H. Paulheim, Enhancing a location-based recommendation system by enrichment with structured data from the web, in: Web Intelligence, Mining and Semantics, 2014.
[112] P. Ristoski, H. Paulheim, A comparison of propositionalization strategies for creating features from linked open data, in: Linked Data for Knowledge Discovery, 2014.
[113] Y. Huang, V. Tresp, M. Nickel, A. Rettinger, H.-P. Kriegel, A scalable approach for statistical learning in semantic graphs, Semant. Web 5 (1) (2014) 5–22.
[114] Y. Huang, V. Tresp, M. Bundschus, A. Rettinger, H.-P. Kriegel, Multivariate prediction for learning on the semantic web, in: P. Frasconi, F. Lisi (Eds.), Inductive Logic Programming, in: Lecture Notes in Computer Science, vol. 6489, Springer, Berlin, Heidelberg, 2011, pp. 92–104. http://dx.doi.org/10.1007/978-3-642-21295-6_13.
[115] Y. Huang, M. Nickel, V. Tresp, H.-P. Kriegel, A scalable kernel approach to learning in semantic graphs with applications to linked data, in: 1st Workshop on Mining the Future Internet, 2010.
[116] N. Fanizzi, C. d'Amato, A declarative kernel for alc concept descriptions, in: Foundations of Intelligent Systems, Springer, 2006, pp. 322–331.
[117] N. Fanizzi, F. Esposito, Statistical learning for inductive query answering on owl ontologies, in: Proceedings of the 7th International Semantic Web Conference, ISWC, 2008.
[118] S. Bloehdorn, Y. Sure, Kernel methods for mining instance data in ontologies, in: Proceedings of the 6th International The Semantic Web and 2Nd Asian Conference on Asian Semantic Web Conference, ISWC'07/ASWC'07, Springer-Verlag, Berlin, Heidelberg, 2007, pp. 58–71. URL http://dl.acm.org/citation.cfm?id=1785162.1785168.
[119] V. Bicer, T. Tran, A. Gossen, Relational kernel machines for learning from graph-structured rdf data, in: The Semantic Web: Research and Applications, Springer, 2011, pp. 47–62.
[120] U. Lösch, S. Bloehdorn, A. Rettinger, Graph kernels for rdf data, in: The Semantic Web: Research and Applications, Springer, 2012, pp. 134–148.
[121] G.K.D. de Vries, S. de Rooij, A fast and simple graph kernel for rdf, in: Proceedings of the Second International Workshop on Knowledge Discovery and Data Mining Meets Linked Open Data, 2013.
[122] G.K.D. de Vries, A fast approximation of the weisfeiler-lehman graph kernel for rdf data, in: H. Blockeel, K. Kersting, S. Nijssen, F. Zelezn (Eds.), ECML/PKDD (1), in: Lecture Notes in Computer Science, vol. 8188, Springer, 2013, pp. 606–621. URL http://dblp.uni-trier.de/db/conf/pkdd/pkdd2013-1.html#Vries13.
[123] P. Bloem, A. Wibisono, G. de Vries, Simplifying rdf data for graph-based machine learning, in: 11th ESWC 2014, ESWC2014. 2014, URL http://data.semanticweb.org/conference/eswc/2014/paper/ws/KnowLOD/8.
[124] G.K.D. de Vries, S. de Rooij, Substructure counting graph kernels for machine learning from rdf data, Web Semantics: Science, Services and Agents on the World Wide Web.
[125] N. Shervashidze, P. Schweitzer, E.J. Van Leeuwen, K. Mehlhorn, K.M. Borgwardt, Weisfeiler-lehman graph kernels, J. Mach. Learn. Res. 12 (2011) 2539–2561.
[126] G.H. John, R. Kohavi, K. Pfleger, Irrelevant features and the subset selection problem, in: ICML'94, 1994, pp. 121–129.
[127] A.L. Blum, P. Langley, Selection of relevant features and examples in machine learning, Artif. Intell. 97 (1997) 245–271.
[128] P. Ristoski, H. Paulheim, Feature selection in hierarchical feature spaces, in: Discovery Science, 2014.
[129] Y. Jeong, S.-H. Myaeng, Feature selection using a semantic hierarchy for event recognition and type classification, in: International Joint Conference on Natural Language Processing, 2013.
[130] B.B. Wang, R.I.B. Mckay, H.A. Abbass, M. Barlow, A comparative study for domain ontology guided feature extraction, in: Australasian Computer Science Conference, 2003.
[131] S. Lu, Y. Ye, R. Tsui, H. Su, R. Rexit, S. Wesaratchakit, X. Liu, R. Hwa, Domain ontology-based feature reduction for high dimensional drug data and its application to 30-day heart failure readmission prediction, in: International Conference Conference on Collaborative Computing Collaboratecom, 2013, pp. 478–484.
[132] J. Euzenat, P. Shvaiko, Ontology Matching, Springer-Verlag, New York, Inc., Secaucus, NJ, USA, 2007.
[133] F.M. Suchanek, S. Abiteboul, P. Senellart, PARIS: Probabilistic alignment of relations, instances, and schema, PVLDB 5 (3) (2011) 157–168.
[134] A. Bellandi, B. Furletti, V. Grossi, A. Romei, Ontology-driven Association Rule Extraction: A Case Study, Contexts and Ontologies Representation and Reasoning, 2007, p. 10.

[135] C. Antunes, Onto4ar: a framework for mining association rules, in: Workshop on Constraint-Based Mining and Learning, CMILE-ECML/PKDD 2007, 2007, pp. 37–48.
[136] C. Antunes, An ontology-based framework for mining patterns in the presence of background knowledge, in: 1st International Conference on Advanced Intelligence, 2008, pp. 163–168.
[137] A.C.B. Garcia, A.S. Vivacqua, Does ontology help make sense of a complex world or does it create a biased interpretation?, in: Proc. Sensemaking Workshop in CHI, vol. 8, 2008.
[138] A.C. Bicharra Garcia, I. Ferraz, A.s. Vivacqua, From data to knowledge mining, Artif. Intell. Eng. Des. Anal. Manuf. 23 (4) (2009) 427–441. http://dx.doi.org/10.1017/S089006040900016X.
[139] M. Zeman, M. Ralbovskỳ, V. Svátek, J. Rauch, Ontology-driven data preparation for association mining, Online http://keg.vse.cz/onto-kdd-draft.pdf.
[140] M. Žáková, P. Kremen, F. Zelezny, N. Lavrac, Automating knowledge discovery workflow composition through ontology-based planning, IEEE Trans. Autom. Sci. Eng. 8 (2) (2010) 253–264.
[141] C. Diamantini, D. Potena, E. Storti, Kddonto: An ontology for discovery and composition of kdd algorithms, in: Third Generation Data Mining: Towards Service-Oriented Knowledge Discovery, SoKD'09, 2009, pp. 13–24.
[142] J. Kietz, F. Serban, A. Bernstein, S. Fischer, Towards cooperative planning of data mining workflows, in: Proceedings of the Third Generation Data Mining Workshop at the 2009 European Conference on Machine Learning, ECML 2009, 2009, pp. 1–12.
[143] M. Hilario, A. Kalousis, P. Nguyen, A. Woznica, A data mining ontology for algorithm selection and meta-mining, in: Proceedings of the ECML/PKDD09 Workshop on 3rd Generation Data Mining, SoKD-09, 2009, pp. 76–87.
[144] M. Hilario, P. Nguyen, H. Do, A. Woznica, A. Kalousis, Ontology-based meta-mining of knowledge discovery workflows, in: Meta-Learning in Computational Intelligence, Springer, 2011, pp. 273–315.
[145] P. Panov, S. Dzeroski, L.N. Soldatova, Ontodm: An ontology of data mining, in: Data Mining Workshops, 2008. ICDMW'08. IEEE International Conference on, IEEE, 2008, pp. 752–760.
[146] P. Panov, S. Džeroski, L.N. Soldatova, Representing entities in the ontodm data mining ontology, in: Inductive Databases and Constraint-Based Data Mining, Springer, 2010, pp. 27–58.
[147] P. Panov, L. Soldatova, S. DÅeroski, Ontology of core data mining entities, Data Min. Knowl. Discov. 28 (5–6) (2014) 1222–1265. http://dx.doi.org/10.1007/s10618-014-0363-0.
[148] P. Panov, L. Soldatova, S. Džeroski, Ontodm-kdd: Ontology for representing the knowledge discovery process, in: Discovery Science, Springer, 2013, pp. 126–140.
[149] F. Serban, J. Vanschoren, J.-U. Kietz, A. Bernstein, A survey of intelligent assistants for data analysis, ACM Comput. Surv. 45 (3) (2013) 31.
[150] A. Suyama, N. Negishi, T. Yamaguchi, Camlet: A platform for automatic composition of inductive learning systems using ontologies, in: PRICAI'98: Topics in Artificial Intelligence, Springer, 1998, pp. 205–215.
[151] A. Bernstein, F. Provost, S. Hill, Toward intelligent assistance for a data mining process: An ontology-based approach for cost-sensitive classification, IEEE Trans. Knowl. Data Eng. 17 (4) (2005) 503–518.
[152] M. Žáková, P. Kremen, F. Zelezny, N. Lavrac, Planning to learn with a knowledge discovery ontology, in: Second Planning to Learn Workshop (planlearn) at the ICML/COLT/UAI 2008, 2008, p. 29.
[153] M. Záková, V. Podpecan, F. Zeleznỳ, N. Lavrac, Advancing data mining workflow construction: A framework and cases using the orange toolkit, in: Proc. 2nd Intl. Wshop. Third Generation Data Mining: Towards Service-Oriented Knowledge Discovery, 2009, pp. 39–52.
[154] V. Podpečan, M. Zemenova, N. Lavrač, Orange4ws environment for service-oriented data mining, Comput. J. (2011) bxr077.
[155] C. Diamantini, D. Potena, Semantic annotation and services for kdd tools sharing and reuse, in: ICDM Workshops, 2008, pp. 761–770.
[156] C. Diamantini, D. Potena, E. Storti, Supporting users in kdd processes design: a semantic similarity matching approach, in: Planning to Learn Workshop (PlanLearn'10) at ECAI, 2010, pp. 27–34.
[157] J.-U. Kietz, F. Serban, A. Bernstein, S. Fischer, Data mining workflow templates for intelligent discovery assistance and auto-experimentation, in: Third-Generation Data Mining: Towards Service-oriented Knowledge Discovery, SoKD-10, 2010, pp. 1–12.
[158] J.-U. Kietz, F. Serban, A. Bernstein, eproplan: A tool to model automatic generation of data mining workflows, in: Proceedings of the 3rd Planning to Learn Workshop (WS9) at ECAI, vol. 2010, 2010.
[159] J.-U. Kietz, F. Serban, S. Fischer, A. Bernstein, semantics inside! but let's not tell the data miners: Intelligent support for data mining, in: The Semantic Web: Trends and Challenges, Springer, 2014, pp. 706–720.
[160] S. Džeroski, Towards a general framework for data mining, in: KDID'06, Springer-Verlag, Berlin, Heidelberg, 2007, URL http://dl.acm.org/citation.cfm?id=1777194.1777213.
[161] A. Gabriel, H. Paulheim, F. Janssen, Learning semantically coherent rules, in: Interactions between Data Mining and Natural Language Processing, 2014, pp. 49–63.
[162] J. Fürnkranz, Separate-and-conquer rule learning, Artif. Intell. Rev. 13 (1) (1999) 3–54.
[163] F.M. Pinto, M.F. Santos, Considering application domain ontologies for data mining, Trans. Inform. Sci. Appl. 6 (9) (2009) 1478–1492.
[164] D. Pan, J.-Y. Shen, M.-X. Zhou, Incorporating domain knowledge into data mining process: An ontology based framework, Wuhan Univ. J. Nat. Sci. 11 (1) (2006) 165–169.
[165] D. Pan, Y. Pan, Using ontology repository to support data mining, in: Intelligent Control and Automation, 2006. WCICA 2006, in: The Sixth World Congress on, vol. 2, IEEE, 2006, pp. 5947–5951.
[166] H. Cešpivová, J. Rauch, V. Svatek, M. Kejkula, M. Tomeckova, Roles of medical ontology in association mining crisp-dm cycle, in: ECML/PKDD04 Workshop on Knowledge Discovery and Ontologies, KDO04, vol. 220, Pisa, Citeseer, 2004.
[167] P. Ristoski, H. Paulheim, Visual analysis of statistical data on maps using linked open data, in: The 12th Extented Semantic Web Conference, ESWC2015. URL http://data.semanticweb.org/conference/eswc/2015/paper/demo/12.
[168] M. d'Aquin, N. Jay, Interpreting data mining results with linked data for learning analytics: Motivation, case study and directions, in: Proceedings of the Third International Conference on Learning Analytics and Knowledge, LAK'13, ACM, New York, NY, USA, 2013, pp. 155–164. http://dx.doi.org/10.1145/2460296.2460327. URL http://doi.acm.org/10.1145/2460296.2460327.
[169] N. Jay, M. d'Aquin, Linked data and online classifications to organise mined patterns in patient data, in: AMIA Annual Symposium Proceedings, vol. 2013, American Medical Informatics Association, 2013, p. 681.
[170] I. Tiddi, Explaining Data Patterns using Background Knowledge from Linked Data, 2013.
[171] I. Tiddi, M. d'Aquin, E. Motta, Explaining clusters with inductive logic programming and linked data, in: Proceedings of the ISWC 2013 Posters & Demonstrations Track, Sydney, Australia, October 23, 2013, 2013, pp. 257–260. URL http://ceur-ws.org/Vol-1035/iswc2013_poster_20.pdf.
[172] I. Tiddi, M. dAquin, E. Motta, Dedalo: Looking for clusters explanations in a labyrinth of linked data, in: V. Presutti, C. dAmato, F. Gandon, M. dAquin, S. Staab, A. Tordai (Eds.), The Semantic Web: Trends and Challenges, in: Lecture Notes in Computer Science, vol. 8465, Springer International Publishing, 2014, pp. 333–348. http://dx.doi.org/10.1007/978-3-319-07443-6_23.
[173] I. Tiddi, M. d'Aquin, E. Motta, Walking linked data: a graph traversal approach to explain clusters, in: Proceedings of the 5th International Workshop on Consuming Linked Data (COLD 2014) co-located with the 13th International Semantic Web Conference, ISWC 2014, Riva del Garda, Italy, October 20, 2014, 2014. URL http://ceur-ws.org/Vol-1264/cold2014_TiddiDM.pdf.
[174] I. Tiddi, M. dAquin, E. Motta, Using neural networks to aggregate linked data rules, in: K. Janowicz, S. Schlobach, P. Lambrix, E. Hyvnen (Eds.), Knowledge Engineering and Knowledge Management, in: Lecture Notes in Computer Science, vol. 8876, Springer International Publishing, 2014, pp. 547–562. http://dx.doi.org/10.1007/978-3-319-13704-9_41.
[175] W. Klsgen, Knowledge discovery in databases and data mining, in: Z. RaÅ, M. Michalewicz (Eds.), Foundations of Intelligent Systems, in: Lecture Notes in Computer Science, vol. 1079, Springer, Berlin, Heidelberg, 1996, pp. 623–632. http://dx.doi.org/10.1007/3-540-61286-6_186.
[176] I. Trajkovski, N. Lavrač, J. Tolar, Segs: Search for enriched gene sets in microarray data, J. Biomed. Inform. 41 (4) (2008) 588–601.
[177] V. Podpecan, N. Lavrac, I. Mozetic, P.K. Novak, I. Trajkovski, L. Langohr, K. Kulovesi, H. Toivonen, M. Petek, H. Motaln, K. Gruden, Segmine workflows for semantic microarray data analysis in orange4ws, BMC Bioinform. (2011) 416–416.
[178] L. Eronen, H. Toivonen, Biomine: predicting links between biological entities using network models of heterogeneous databases, BMC Bioinform. 13 (1) (2012) 119.
[179] P.K. Novak, A. Vavpetič, I. Trajkovski, N. Lavrač, N.: Towards semantic data mining with g-segs, in: Proceedings of the 11th International Multiconference Information Society, IS, 2009.
[180] N. Lavrač, A. Vavpetič, L. Soldatova, I. Trajkovski, P.K. Novak, Using ontologies in semantic data mining with segs and g-segs, in: Proceedings of the 14th International Conference on Discovery Science, DS'11, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 165–178.
URL http://dl.acm.org/citation.cfm?id=2050236.2050251.
[181] A. Vavpetič, N. Lavrač, Semantic subgroup discovery systems and workflows in the sdm-toolkit, Comput. J. (2013) 304–320.
[182] A. Vavpetič, P.K. Novak, M. Grčar, I. Mozeti, N. Lavrač, Semantic data mining of financial news articles, in: J. Frnkranz, E. Hllermeier, T. Higuchi (Eds.), Discovery Science, in: Lecture Notes in Computer Science, vol. 8140, Springer, Berlin, Heidelberg, 2013, pp. 294–307. http://dx.doi.org/10.1007/978-3-642-40897-7_20.
[183] N. Lavrač, P.K. Novak, Relational and Semantic Data Mining for Biomedical Research, 2012.
[184] A. Vavpetič, V. Podpečan, N. Lavrač, Semantic subgroup explanations, J. Intell. Inf. Syst. 42 (2) (2014) 233–254. http://dx.doi.org/10.1007/s10844-013-0292-1.
[185] B. Schäfer, P. Ristoski, H. Paulheim, What is special about bethlehem, pennsylvania? identifying unexpected facts about dbpedia entities, in: International Semantic Web Conferences, Posters and Demos, 2015.
[186] H. Paulheim, R. Meusel, A decomposition of the outlier detection problem into a set of supervised learning problems, Mach. Learn. (2–3) (2015) 509–531.

[187] R. Srikant, R. Agrawal, Mining generalized association rules, in: VLDB, vol. 95, 1995, pp. 407–419.
[188] X. Zhou, J. Geller, Raising, to enhance rule mining in web marketing with the use of an ontology, in: Data Mining with Ontologies: Implementations, Findings and Frameworks, 2007, pp. 18–36.
[189] M.A. Domingues, S.O. Rezende, Using taxonomies to facilitate the analysis of the association rules, arXiv preprint, arXiv:1112.1734.
[190] C. Marinica, F. Guillet, Knowledge-based interactive postmining of association rules using ontologies, IEEE Trans. Knowl. Data Eng. 22 (6) (2010) 784–797.
[191] Z. Huang, H. Chen, T. Yu, H. Sheng, Z. Luo, Y. Mao, Semantic text mining with linked data, in: INC, IMS and IDC, 2009. NCM'09. Fifth International Joint Conference on, 2009, pp. 338–343. http://dx.doi.org/10.1109/NCM.2009.131.
[192] V. Svátek, J. Rauch, M. Ralbovská, Ontology-enhanced association mining, in: M. Ackermann, B. Berendt, M. Grobelnik, A. Hotho, D. Mladenič, G. Semeraro, M. Spiliopoulou, G. Stumme, V. Svátek, M. van Someren (Eds.), Semantics, Web and Mining, in: Lecture Notes in Computer Science, vol. 4289, Springer, Berlin, Heidelberg, 2006, pp. 163–179. http://dx.doi.org/10.1007/11908678_11.
[193] T. Di Noia, I. Cantador, V.C. Ostuni, Linked open data-enabled recommender systems: Eswc 2014 challenge on book recommendation, in: Semantic Web Evaluation Challenge, Springer, 2014, pp. 129–143.
[194] T. Di Noia, R. Mirizzi, V.C. Ostuni, D. Romito, M. Zanker, Linked open data to support content-based recommender systems, in: Proceedings of the 8th International Conference on Semantic Systems, I-SEMANTICS '12, ACM, New York, NY, USA, 2012, pp. 1–8. http://dx.doi.org/10.1145/2362499.2362501. URL http://doi.acm.org/10.1145/2362499.2362501.