Towards Linked Open Data Enabled Data Mining

Strategies for Feature Generation, Propositionalization, Selection, and Consolidation

Petar Ristoski(✉)

University of Mannheim, Mannheim, Germany
petar.ristoski@informatik.uni-mannheim.de

Abstract. Background knowledge from Linked Open Data sources can be used to improve the results of a data mining problem at hand: predictive models can become more accurate, and descriptive models can reveal more interesting findings. However, collecting and integrating background knowledge is a tedious manual work. In this paper we propose a set of desiderata, and identify the challenges for developing a framework for unsupervised generation of data mining features from Linked Data.

Keywords: Linked Open Data · Data mining · Feature generation

1 Introduction

Knowledge discovery is defined as "a non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data" [9]. 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, (b) in external data, which has to be included with the problem first, 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. 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. However, collecting and integrating large amounts of background knowledge can be a labor intensive task. Moreover, in most cases, only a small fraction of that background knowledge will be actually used in the data mining model itself, but it is hard to pinpoint the relevant parts in advance. Furthermore, variables involved in unexpected findings are easily overseen, since assumptions

F. Gandon et al. (Eds.): ESWC 2015, LNCS 9088, pp. 772–782, 2015.
DOI: 10.1007/978-3-319-18818-8_50

about interrelations in the application domain lead the user when selecting additional attributes, i.e., she will be subject to a selection bias. To overcome these shortcomings, Linked Open Data represents a valuable source of background knowledge.

Linked Open Data (LOD) is an open, interlinked collection of datasets in machine-interpretable form, built on W3C standards as RDF[1], and SPARQL[2]. Currently the LOD cloud consist of about 1,000 datasets covering various domains [1,25], making it a valuable source for background knowledge in data mining.

Figure 1 gives an overview of a general LOD-enabled knowledge discovery process. 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. After the links are set, outgoing links to external LOD datasets can be explored. In the next step, various techniques for data consolidation and cleansing are applied. Next, 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. After the data transformation is done, a suitable data mining algorithm is applied on the data. In the final step, the results of the data mining process are presented to the user.

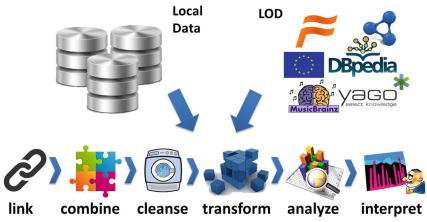

Fig. 1. LOD-enabled knowledge discovery process

In this proposal we focus on the second, third and fourth step of the LOD-enabled knowledge discovery pipeline. Moreover, we propose a framework for automated unsupervised generation of data mining features from LOD. Such a framework should be able to find useful and relevant data mining features, which can be used in any arbitrary predictive or descriptive data mining model, aiming to increase the model's performances.

2 Problem Statement and Contributions

To develop a scalable framework for unsupervised generation of data mining features from LOD, we will need to address the following working domains:

Feature Generation. 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 $\langle f_1, f_2, ..., f_n \rangle$, 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). Linked Open Data, however, comes in the form of *graphs*, connecting resources with types and relations, backed by a schema or ontology.

[1] W3C. RDF. http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/, 2004.
[2] W3C. SPARQL Query Language for RDF. http://www.w3.org/TR/rdf-sparql-query/, 2008.

Thus, for accessing LOD with existing data mining tools, transformations have to be performed, which create propositional features from the graphs in LOD, i.e., a process called *propositionalization* [14].

Defining an appropriate set of features for a data mining problem at hand is still much of an art. However, it is also a step of key importance for the successful use of data mining. Therefore, we define requirements the feature generation framework needs to fulfill:

(i) Given a data mining task, and an input data mining dataset, with the corresponding $1:1$ or $1:m$ (common in text mining) mappings of the local instances to LOD entities, the framework should be able to generate features from any given LOD source that are highly relevant for the given data mining task, where the task is predictive or descriptive.

(ii) Beside setting basic parameters, the feature generation should be performed without user interaction, i.e., unsupervised and automated.

(iii) The generated feature set should be optimal, i.e., the goal is to generate minimal feature set that maximizes the learning model's performances, and minimizes the cost of the feature generation process itself. For creating such optimal feature set, two paradigms exist: *minimal representation*, and *maximal between-class separability*. The minimal representation implies that the instances in the input dataset should be represented with as simple feature set as possible that fully describes all target concepts, e.g., Occam's razor approach. To provide a good generalization such approaches should appropriately address the *bias-variance dilemma*, i.e., the generated features should be general enough to keep the variance low, but relevant enough to keep the bias low. The second paradigm, mainly applicable in classifiers design, relates to generating feature set that guarantees *maximal between-class separability* for a given data set, and thus help building better learning models.

(iv) Although the input dataset contains only links to one LOD dataset, the framework should be able to find useful features from multiple LOD sources by exploring links (such as *owl:sameAs*).

(v) When designing such a framework, scalability should be taken in mind, as the size of many LOD datasets is rather large, e.g., DBpedia 2014[3] contains about 3 billion triples.

(vi) The framework should comply with various standards for publishing and consuming LOD, e.g., the data can be served via SPARQL endpoint, RDF dumps, or URI dereferencing.

Propositionalization Strategies. When generating data mining features from graph-based data, different propositionalization strategies can be used. For example, the standard binary or numerical representation can be used, or more sophisticated representation strategies that use some graph characteristics might be introduced. Our hypothesis is that the strategy of creating features may have an influence on the data mining result. For example, proximity-based algorithms like k-NN will behave differently depending on the strategy used to create numerical features, as the strategy has a direct influence on most distance functions.

[3] http://dbpedia.org/About.

Feature Selection. Although the optimal feature generation approach would not require the use of feature selection step afterwards, in some cases the feature selection step might be desirable. For example, the complexity of the feature generation approach can be reduced by allowing it to generate features with more flexible constraints, which will be later processed by the feature selection algorithm. In feature vectors generated from LOD we can often observe relations between the features, which in most of the cases are explicitly expressed in the LOD schema, or can be inferred using appropriate reasoning approaches. If those relations are not properly explored during the feature generation step, it can be done in the feature selection step to reduce the feature space, which will allow us to remove correlated, contradictory, and repetitive features.

Feature Consolidation. 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 the property *db:areaTotal*, while in YAGO[4] using the property *yago:hasArea*. The problem of aligning properties, as well as instances and classes, in ontologies is addressed by *ontology matching* techniques [7]. Using such techniques, we can find correspondences between features in multiple LOD sources, which then can be fused into a single feature using data fusion techniques [2]. Such a fusion can provide a feature that would mitigate missing values and single errors for individual sources, leading to only one high-value feature.

The initial contributions of the proposal can be summarized as follows: (i) A framework for automated unsupervised generation of data mining features from LOD, from single or (ii) multiple LOD sources. (iii) Novel propositionalization strategies for generating features from LOD, and analysis on their effect on the performances of the data mining models. (iv) Novel feature selection and consolidation methodologies that can be applied on features generated from LOD.

3 State of the Art

Feature Generation. In the recent past, a few approaches for generating data mining features from Linked Open Data have been proposed. Many of those approaches are supervised, i.e., they let the user formulate SPARQL queries, and a fully automatic feature generation is not possible. LiDDM [12] is an integrated system for data mining on the semantic web, allowing the users to declare SPARQL queries for retrieving features from LOD that can be used in different machine learning techniques. Similar approach has been used in the RapidMiner[5] semweb plugin [13], which preprocesses RDF data in a way that it can be further processed directly in RapidMiner. Cheng et al. [4] proposes an approach for automated feature generation after the user has specified the type of features. To do so, similar like the previous approaches, the users have to specify the SPARQL query, which makes this approach supervised rather than unsupervised.

[4] www.mpi-inf.mpg.de/yago-naga/yago.
[5] http://www.rapidminer.com/.

Mynarz et al. [17] have considered using user specified SPARQL queries in combination with SPARQL aggregates.

FeGeLOD [18] is the first fully automatic unsupervised approach for enriching data with features that are derived from LOD. In this work six different unsupervised feature generation strategies are proposed, by exploring specific or generic relations.

A similar problem is handled by *Kernel functions*, which compute the distance between two data instances, by counting common substructures in the graphs of the instances, i.e. walks, paths and threes. In the past, many graph kernels have been proposed that are tailored towards specific application [10], or towards specific semantic representation [8]. Only several approaches are general enough to be applied on any given RDF data, regardless the data mining task. Lösch et al. [15] introduce two general RDF graph kernels, based on intersection graphs and intersection trees. Later, the intersection tree path kernel was simplified by Vries et al. [6]. In another work, Vries et al. [5] introduce an approximation of the state-of-the-art Weisfeiler-Lehman graph kernel algorithm aimed at improving the computation time of the kernel when applied to RDF.

Furthermore, Tiddi et al. [27] introduced the *Dedalo* framework that traverses LOD to find commonalities that form explanations for items of a cluster. Given a supervised data mining task, such an approach could be easily adapted and used as feature generation approach.

Propositionalization Strategies. Even though several approaches have been proposed for creating propositional features from LOD, usually the resulting features are binary, or numerical aggregates using SPARQL COUNT constructs. Furthermore, none of them provide evaluation of the model performances when using different propositionalization strategies.

Feature Selection. Feature selection is a very important and well studied problem in the literature [3]. The objective is to identify features that are correlated with or predictive of the class label. Standard feature selection methods tend to select the features that have the highest relevance score without exploiting the semantic relations between the features in the feature space. Therefore, such methods are not appropriate to be applied on feature sets generated from LOD.

While there are a lot of state-of-the-art approaches for feature selection in a standard feature space [3], only few approaches for feature selection in a feature space extracted from structured knowledge bases are proposed in the literature. Jeong et al. [11] propose the *TSEL* method using a semantic hierarchy of features based on WordNet relations. The algorithm tries to find the most representative and most effective features from the complete feature space, based on the *lift* measure, and χ^2. Wang et al. [28] propose an k-NN based *bottom-up hill climbing* search algorithm to find an optimal subset of concepts for document representation. Lu et al. [16] describe a *greedy top-down* search strategy, based on the nodes' information gain ratio, trying to select a mixture of concepts from different levels of the hierarchy.

Feature Consolidation. To the best of our knowledge, there is no proposed approach in the literature for generating and consolidating data mining features from multiple LOD sources.

4 Research Methodology and Approach

Feature Generation. So far, we have implemented and extended the approaches initially presented in the FeGeLOD system [18]. For a given input dataset containing the entities and the corresponding LOD entity URIs, the following strategies for feature generation may be used: (i) Generating feature for each direct data property of an entity in the dataset. (ii) Features for specific relations of an entity, e.g. *dcterms:subject* in DBpedia. This approach allows to further explore the relation to a user specified length, e.g., one can follow the *skos:broader* relation for an already extracted *dcterms:subject* from DBpedia. (iii) Features for each incoming or outgoing relation of an entity. (iv) Feature for each incoming or outgoing relation of an entity including the value of the relation. (v) Feature for each incoming or outgoing relation of an entity, including the related types, i.e., they are concerned with qualified relations

Furthermore, we implemented approaches for generating features based on graph sub-structures using graph kernels: the *Weisfeiler-Lehman Kernel* [5], and the *Intersection Tree Path Kernel* [6,15].

These approaches are rather trivial and simplistic. As shown in the evaluation, using these approaches we are able to generate useful feature vectors that improve the performances of the learning model in unsupervised environment. However, the generated feature vectors are rather large and contain many irrelevant features.

Propositionalization Strategies. In this phase we have only considered some of the trivial propositionalization strategies: (i) *Binary*, indicating the presence of a given feature. (ii) *Count*, specifying the exact number of appearances of the feature. (iii) *Relative Count*, specifying the relative number of appearances of the feature. (iv) *TF-IDF*, calculated using the standard TF-IDF equation.

More sophisticated propositionalization strategies might be developed. For example, the target variable from the local dataset can be used for developing supervised weighting approaches, as used in some text mining application. Furthermore, we can use the graph properties for calculating feature weights, e.g., the fan-in and fan-out values of the graph nodes can give a better representation of the popularity of the resources included in the features, which might be a good indicator of the feature's relevance for the data mining task. More sophisticated popularity scores can be calculated using some of the standard graph ranking algorithms, e.g., PageRank and HITS.

Feature Selection. We have introduced an approach [24] that exploits hierarchies for feature selection in combination with standard metrics, such as *information gain* and *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. To measure the similarity of relevance between two nodes, we use the standard correlation and information gain measure. The approach is implemented in two steps, i.e., initial selection and pruning. In the first step, we try to identify, and filter out the ranges of nodes with similar relevance in each branch of the hierarchy. In the second step we try to select only the most valuable features from the previously reduced set.

Feature Consolidation. To identify features that represent the same information retrieved from multiple LOD sources, we have implemented an approach that relies on the probabilistic algorithm for ontology matching PARIS [26]. The approach outputs all discovered properties correspondences, which then can be resolved using different conflict resolution strategies [2], e.g., majority voting, average, etc. New fusion strategies can be developed based on the provenance information, e.g., if building a learning model in the movies domain, information retrieved from movies specific LOD sources (like LinkedMDB[6]) should be more accurate and extensive than cross-domain LOD sources (like DBpedia).

5 Evaluation Plan

To evaluate the feature generation framework, the feature selection and consolidation, and the propositionalization strategies, we need to collect significant number of datasets that cover different application domains, and can be used in different data mining tasks and different data mining algorithms. We consider two types of dataset for evaluation. First, datasets that already contain initial data mining features and a target variable. Such datasets could be easily collected from some of the popular machine learning repositories, like the UCI ML Repository[7]. The initial features of such datasets could be used for building models using state-of-the-art methods, which will serve as baselines for evaluating the performances of the learning models built on the enriched datasets with LOD features. An example for such a dataset is the *Auto MPG* dataset[8], which captures different characteristics of cars (such as cyclinders, horsepower, etc.), and the target is to predict the fuel consumption.

The second category of datasets are so called "empty datasets", which contain only the instances and one or more target variables. An example for such a dataset is the Mercer quality of living dataset[9], which contains a list of cities and their quality of living as numerical value (the target variable).

To evaluate the performances of a given data mining model, performance function p is used. In different data mining tasks different performance functions are used, e.g., accuracy is used for classification; root mean squared error for

[6] http://www.linkedmdb.org/.
[7] http://archive.ics.uci.edu/ml/index.html.
[8] http://archive.ics.uci.edu/ml/datasets/Auto+MPG.
[9] http://across.co.nz/qualityofliving.htm.

regression; support and confidence for association rules; purity and entropy for clustering, etc. Then, the evaluation for each of the given data mining tasks can be easily performed just by using the corresponding performance function on the model built on the enriched dataset.

For supervised data mining tasks where gold standard is available, the evaluation can be performed using some of the standard evaluation techniques, e.g. cross-validation. However, in unsupervised data mining tasks, like rule learning or clustering, in many cases the validity of the discovered patterns and hypothesis cannot be trivially and uniformly decided. Therefore, a user study may need to be conducted, where humans can decide the validity of the discovered hypothesis. For example, the ratings could be acquired using services like Amazon Mechanical Turk[10] or CrowdFlower[11].

As the feature generation complexity may rise very fast, as well as the number of generated features, a second evaluation metric should be introduced. Such a metric should be able to measure the trade-off between the feature generation complexity, the learning model training runtime on the enriched dataset, and the model performances.

To evaluate the performances of the feature selection approaches we introduce the *feature space compression* measure, which is defined as: $c(V') := 1 - \frac{|V'|}{|V|}$, where V is the original feature space, V' is the filtered feature space, and $V' \subseteq V$. Since there is a trade-off between the feature set and the performances, an overall target function is, e.g., the harmonic mean of p and c[12].

To evaluate the feature consolidation approaches we can collect some existing datasets that are commonly used for evaluation in the ontology matching community, or generate new ones. Once the gold standard is defined, standard evaluation metrics may be used, e.g., precision, recall and F-measure. To evaluate the model performances on the reduced feature space, again we use the model performance function p.

6 Intermediate Results

In this section we present some initial results of the approaches described in this proposal. The approaches are implemented in the RapidMiner Linked Open Data extension[13] [19,20], which represents an integral part of this thesis. The RapidMiner LOD extension supports the user in all steps of the LOD-enabled knowledge discovery process. The extension is publicly available, and has been successfully used in several applications.

Feature Generation. The initial feature generation strategies from the FeGeLOD framework have been evaluated in several prior publications. In [20,22]

[10] https://www.mturk.com.

[11] http://www.crowdflower.com/.

[12] Note that the value for p might need to be normalized first, depending on the used metric.

[13] http://dws.informatik.uni-mannheim.de/en/research/rapidminer-lod-extension.

we have shown that features generated from LOD can help finding useful explanations for interpreting statistical data. In [21] several LOD sources were used to generate features that can be used in books recommender systems. More extensive evaluation of the strategies was performed in [19,23]. The evaluation on the Cities and the Auto MPG datasets is extended and presented here.

We use the Cities dataset for classification (the target variable was discretized into high, medium, and low) using three classification methods. The Auto MPG dataset is used for the task of regression, also using three regression methods. The instances of both datasets were first linked to the corresponding resource in DBpedia, and then the following feature sets were generated: direct types (rdf:type), categories (dcterms:subject), incoming relations (rel in), outgoing relations (rel out), combination of both, outgoing relations including values (rel-vals out), incoming relations including values (rel-vals in), numerical values, and dataset generated using the Weisfeiler-Lehman graph kernel algorithm (WLK).

Table 1 depicts the size and the results for each feature set, except for the incoming relations values set, which is rather large to be evaluated. We can notice that the features generated from LOD lead to RMSE five times smaller than the original data. From the results we can notice that the results differ for different feature sets, and different algorithms, but in almost all cases the features generated using the kernel feature generation strategy lead to the best results. However, the complexity for generating the kernel functions is by three orders of magnitude higher than any other strategy. Additionally, the number of features generated with the kernel strategy is 20 to 40 times higher than any other strategy, which also greatly affects the runtime for building the learning models. Therefore, a near optimal trade-off between the feature generation complexity, the size of the dataset and the learning model performances should be found.

Table 1. Classification accuracy results for the Cities dataset, and RMSE results for the Auto MPG dataset.

Dataset	Cities				Auto MPG			
Set/Method	#Att.	NB	KNN	C4.5	#Att.	LR	M5	KNN
original	0	/	/	/	8	3.35	2.85	4.02
types	721	55.71	56.17	59.05	264	3.84	2.83	3.57
categories	999	59.52	44.35	58.96	308	4.47	2.9	3.62
rel in	1,304	60.41	58.46	60.35	227	3.84	2.9	3.61
rel out	1,081	47.62	**60.0**	56.71	370	3.79	3.1	3.6
rel in & out	2,385	59.44	58.57	56.47	597	3.92	3.0	3.57
rel-vals out	3,091	53.68	49.98	61.82	1,497	**2.87**	1.83	1.50
numerics	774	46.29	34.48	49.98	185	4.32	3.47	2.98
WLK	48,373	**64.55**	52.36	**71.26**	26,687	3.05	**1.69**	**0.74**

Propositionalization Strategies. In [23] we performed an evaluation on different propositionalization strategies on three different data-mining tasks, i.e., classification, regression and outlier detection, using three different data mining algorithms for each task. The evaluation was performed for binary, numerical, relative count and TF-IDF vector representation, on five different feature sets. The evaluation showed that the propositionalization strategy have major impact on the data mining results, however we were not able to come with a general recommendation for a strategy, as it depends on the given data mining task, the given dataset, and the data mining algorithm to be used.

Feature Selection. In [24] we have performed initial evaluation of the feature selection approach in hierarchical feature spaces, on both synthetic and real

world dataset, using three algorithms for classification. Using the approach, we were able to achieve feature space compression up to 95 %, without decreasing the model's performances, or in some cases increasing it. The evaluation has shown that the approach outperforms standard feature selection techniques as well as recent approaches which explore hierarchies.

Feature Consolidation. In [20] 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 matching and fusion approach were merged into a single feature without missing values.

7 Conclusion

In this work we have identified the challenges, and set the initial bases for developing a scalable framework for automatic and unsupervised feature generation from LOD that can be used in any arbitrary data mining algorithms. We believe that such a framework will be of a great value in the data preparation step of the knowledge discovery process, by reducing the time needed for data transformation and manipulation, with as little as possible user interaction.

Acknowledgements. This thesis is supervised by prof. Dr. Heiko Paulheim. 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. Bizer, C., Heath, T., Berners-Lee, T.: Linked data - the story so far. IJSWIS **5**, 1–22 (2009)
2. Bleiholder, J., Naumann, F.: Data fusion. ACM Comput. Surv. **41**(1), 1–41 (2008)
3. Blum, A.L., Langley, P.: Selection of relevant features and examples in machine learning. Artif. intell. **97**, 245–271 (1997)
4. Cheng, W., Kasneci, G., Graepel, T., Stern, D., Herbrich, R.: Automated feature generation from structured knowledge. In: CIKM (2011)
5. de Vries, G.K.D.: A fast approximation of the Weisfeiler-Lehman graph kernel for RDF data. In: Blockeel, H., Kersting, K., Nijssen, S., Železný, F. (eds.) ECML PKDD 2013, Part I. LNCS, vol. 8188, pp. 606–621. Springer, Heidelberg (2013)
6. de Vries, G.K.D., de Rooij, S.: A fast and simple graph kernel for RDF. In: DMLOD (2013)
7. Euzenat, J., Shvaiko, P.: Ontology Matching. Springer, New York (2007)
8. Fanizzi, N., d'Amato, C.: A declarative kernel for $\mathcal{ALC}$ concept descriptions. In: Esposito, F., Raś, Z.W., Malerba, D., Semeraro, G. (eds.) ISMIS 2006. LNCS (LNAI), vol. 4203, pp. 322–331. Springer, Heidelberg (2006)
9. Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P.: Advances in Knowledge Discovery and Data Mining. AAAI Press, Cambridge (1996)
10. Huang, Y., Tresp, V., Nickel, M., Kriegel, H.-P.: A scalable approach for statistical learning in semantic graphs. Semant. Web **5**, 5–22 (2014)

11. Jeong, Y., Myaeng, S.-H.: Feature selection using a semantic hierarchy for event recognition and type classification. In: International Joint Conference on Natural Language Processing (2013)
12. Kappara, V.N.P., Ichise, R., Vyas, O.P.: Liddm: a data mining system for linked data. In: LDOW (2011)
13. Khan, M.A., Grimnes, G.A., Dengel, A.: Two pre-processing operators for improved learning from semanticweb data. In: RCOMM (2010)
14. Kramer, S., Lavrač, N., Flach, P.: Propositionalization approaches to relational data mining. In: Džeroski, S., Lavrač, N. (eds.) Relational Data Mining, pp. 262–291. Springer, New York (2001)
15. Lösch, U., Bloehdorn, S., Rettinger, A.: Graph kernels for RDF data. In: Simperl, E., Cimiano, P., Polleres, A., Corcho, O., Presutti, V. (eds.) ESWC 2012. LNCS, vol. 7295, pp. 134–148. Springer, Heidelberg (2012)
16. Lu, S., Ye, Y., Tsui, R.: Domain ontology-based feature reduction for high dimensional drug data and its application to 30-day heart failure readmission prediction. In: Collaboratecom, pp. 478–484 (2013)
17. Mynarz, J., Svátek, V.: Towards a benchmark for LOD-enhanced knowledge discovery from structured data. In: The Second International Workshop on Knowledge Discovery and Data Mining Meets Linked Open Data (2013)
18. Paulheim, H., Fürnkranz, J.: Unsupervised generation of data mining features from linked open data. In: WCWIMS (2012)
19. Paulheim, H., Ristoski, P., Mitichkin, E., Bizer, C.: Data mining with background knowledge from the web. In: RapidMiner World (2014)
20. Ristoski, P., Bizer, C., Paulheim, H.: Mining the web of linked data with rapidminer. In: Semantic Web Challenge at ISWC (2014)
21. Ristoski, P., Loza Mencía, E., Paulheim, H.: A hybrid multi-strategy recommender system using linked open data. In: Presutti, V., et al. (eds.) SemWebEval 2014. CCIS, vol. 475, pp. 150–156. Springer, Heidelberg (2014)
22. Ristoski, P., Paulheim, H.: Analyzing statistics with background knowledge from linked open data. In: Workshop on Semantic Statistics (2013)
23. Ristoski, P., Paulheim, H.: A comparison of propositionalization strategies for creating features from linked open data. In: LD4KD (2014)
24. Ristoski, P., Paulheim, H.: Feature selection in hierarchical feature spaces. In: Džeroski, S., Panov, P., Kocev, D., Todorovski, L. (eds.) DS 2014. LNCS, vol. 8777, pp. 288–300. Springer, Heidelberg (2014)
25. Schmachtenberg, M., Bizer, C., Paulheim, H.: Adoption of the linked data best practices in different topical domains. In: Mika, P., Tudorache, T., Bernstein, A., Welty, C., Knoblock, C., Vrandečić, D., Groth, P., Noy, N., Janowicz, K., Goble, C. (eds.) ISWC 2014, Part I. LNCS, vol. 8796, pp. 245–260. Springer, Heidelberg (2014)
26. Suchanek, F.M., Abiteboul, S., Senellart, P.: PARIS: probabilistic alignment of relations, instances, and schema. PVLDB **5**(3), 157–168 (2011)
27. Tiddi, I., d'Aquin, M., Motta, E.: Dedalo: looking for clusters explanations in a labyrinth of linked data. In: Presutti, V., d'Amato, C., Gandon, F., d'Aquin, M., Staab, S., Tordai, A. (eds.) ESWC 2014. LNCS, vol. 8465, pp. 333–348. Springer, Heidelberg (2014)
28. Wang, B.B., Mckay, R.I.B., Abbass, H.A., Barlow, M.: A comparative study for domain ontology guided feature extraction. In: ACSC (2003)