Theme Identification in RDF Graphs*

Hanane Ouksili

PRiSM, Univ. Versailles St Quentin, UMR CNRS 8144, Versailles France
hanane.ouksili@prism.uvsq.fr

Abstract. An increasing number of RDF datasets is published on the Web. A user willing to use these datasets will first have to explore them in order to determine which information is relevant for his specific needs. To facilitate this exploration, we present an approach allowing to provide a thematic view of a given RDF dataset, making it easier to target the relevant resources and properties. Our approach combines a density-based graph clustering algorithm with semantic criteria in order to identify clusters, each one corresponding to a theme. Prior to clustering, the initial RDF graph is simplified, and user preferences are mapped into a set of transformations applied to the graph. Once the clusters are identified, labels are extracted to express their semantics. In this paper, we describe the main features of our approach to generate a set of themes from an RDF dataset.

Keywords: Theme idendification, RDF(S) data, Clustering.

1 Introduction

An increasing number of RDF datasets is published on the Web, making a huge amount of data available for users and applications. In this context, a key issue for the users is to locate the relevant information for their specific needs. A typical way of exploring RDF datasets is the following: the users first select a URI, called a seed of interest, which they are willing to use as a starting point for their queries; then they explore all the URIs reachable from this seed by submitting queries to obtain information about the existing properties.

To facilitate this interaction, a thematic view of an RDF dataset can be given in order to guide the exploration process. We argue that once the data is presented as a set of themes, it is easier to target the relevant resources and properties by exploring the interesting topics only. In this paper, we present our approach for theme identification which combines a density-based graph clustering algorithm with semantic clustering criteria in order to identify clusters, each one corresponding to a theme.

The paper is organized as follows. Section 2 gives an overview of our proposal. Section 3 details the preprocessing step. Section 4 presents the clustering algorithm and we discuss methods of describing themes in Section 5. Our prototype and an example scenario are described in Section 6, related works are provided in Section 7, and finally, Section 8 concludes the paper.

* This work was supported by Electricity of France (EDF R&D).

2 General Principle of Theme Identification

Given an RDF dataset, our goal is to identify a set of themes and to extract the labels or tags which best capture their semantics. Providing this thematic view raises several questions:

- Which information could be used to define a theme?
- As different users may not have the same perception of the data, how to capture their preferences and use them for building the themes?
- Finally, once the themes have been identified, how to label them so as to make their semantic as clear as possible to the user?

Our approach relies on the idea that a theme corresponds to a highly connected area on the RDF graph. The more a set of resources is connected, the more likely it is that they belong to the same theme or are related to the same topic. We will therefore use the structure of the RDF graph itself in order to build the themes. We apply a graph clustering algorithm which identifies these highly connected areas and their neighborhood in order to form clusters, each one corresponding to a theme.

The structure of the graph alone is not sufficient to provide meaningful themes. Indeed, different users may have distinct perceptions of what a theme is. If we consider a dataset providing information about universities and scientists, one possible view is that themes correspond to research areas such as Mathematics or Physics, another one is that themes correspond to research teams located in the same geographical area. These preferences will be used for identifying the themes, in addition to the structure of the graph.

User preferences are captured by specifying the characteristics of all resources which should be assigned to the same cluster (for example, resources having the same value or linked by a given property). Each preference will be mapped into one or several transformations applied to the graph. For example, if the user expresses that two resources related by the *owl:sameAs* property should be assigned to the same cluster, the transformation will consist in merging the corresponding nodes in the graph.

An overview of our approach is given in Figure 1. It comprises three main steps, (i) preprocessing, where transformations are applied on the RDF graph, (ii) graph clustering, where themes are identified, and (iii) label extraction which provides a summary of the content of each cluster. In this paper, we mainly address the first two steps.

3 Preprocessing

The initial RDF graph will be transformed prior to the execution of the clustering algorithm. Some transformations are systematic regardless of the context, others consist in integrating user preferences in the graph. This section describes both of them.

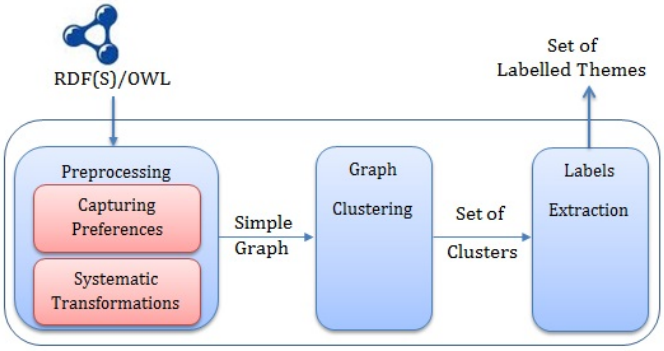

Fig. 1. Overview of our Approach

3.1 Systematic Transformations

Some of the information in the initial graph is not useful in order to group the nodes into meaningful clusters.

In the initial RDF graph, edges are oriented and labeled with the name of a predicate. The clustering algorithm used in our approach will try to identify highly connected areas, regardless of the orientation of the edges; no matter what the orientation of an edge is, what we are interested in is that some semantic relation exists between the resources. For example, consider *dbo:influenced* property that is asymmetric in nature; if we have the triplet $\langle r_i$ *dbo:influenced* $r_j \rangle$. We are not interested in which researcher between r_i and r_j that influenced the other; the most important is there is a semantic relationship between the two researchers. We can therefore simplify the graph by removing the orientation of the edges. Similarly, the clustering algorithm will not use the label of the edges, and they are also removed from the graph.

An RDF graph contains several types of nodes which can be either resources or literals. A literal is related to one resource and is a characteristic of this resource. Obviously, a resource and the related literals should be grouped into the same cluster. We could therefore apply the clustering algorithm on a simplified version of the graph which doesn't contain the literals.

The output of the preprocessing stage is a graph where the labels, orientation of the edges and literal nodes have been removed. Figure 2 shows an example of simplified graph (2(b)) corresponding to an RDF dataset (2(a)).

3.2 Capturing User Preferences

As stated earlier, the structure of the graph alone is not sufficient for the identification of meaningful clusters. Sometimes the density of the graph doesn't fully capture semantic closeness: for example, two resources might not be located in a very connected area of the graph, but if there is an edge in the graph relating them, and if this edge expresses a strong semantic link (e.g. *owl:sameAs*), the

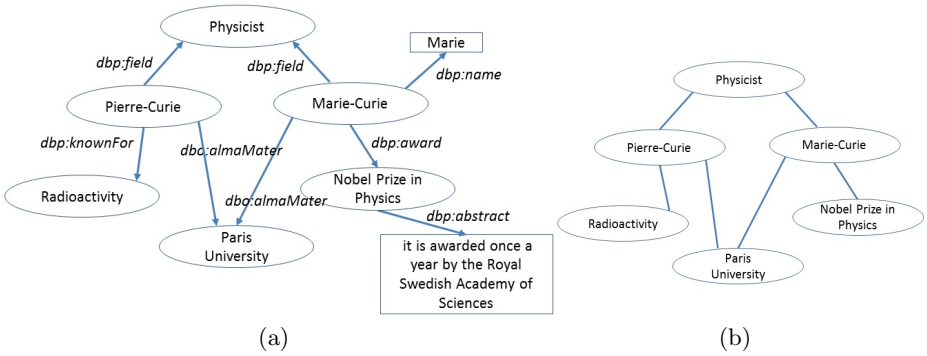

Fig. 2. Transformation of the Initial RDF Graph

two resources should be assigned to the same cluster. Furthermore, for the same dataset, different users might have different points of view and be interested in distinct properties. To capture this need, the clustering should take into account these properties as semantic criteria.

In our approach, user preferences are captured by mapping each of them into one or several graph transformation primitives. We consider that there are mainly two kinds of preferences a user might want to express. The first one is that two resources related by a given property should belong to the same theme. The second one is that a set of resources having the same value for a given property should belong to the same theme.

Grouping Two Resources According to a Property. Some properties express a strong semantic link that should be used as a clustering criteria. For example, resources linked by the *owl:sameAs* property should obviously be assigned to the same cluster, and this is true for any user in any context. Besides, some users may wish to give a property more importance than other users. For example, if we consider a dataset containing information about scientists in different research domains, a given user might consider that the resources *Student* and *Scientist* related by the *dbo:doctoralAdvisor* property should be grouped in the same cluster. This kind of preference is taken into account by merging the two resources.

Grouping a Set of Resources According to a Property. Resources that should be assigned to the same theme are not always linked by a property; the semantic closeness between them can be expressed by the values of some shared property. In other words, a set of resources having the same value for a property p should be assigned to the same cluster. For instance, the user could state that scientists having the same value for the *dbo:field* property should be in the same cluster, thus ensuring that scientists of the same research domain are grouped together. This kind of preference is taken into account by creating in the graph a highly connected area containing the specified resources. If we consider the set R of

resources r_i having the same value for a property p, then an edge (r_i, r_j) will be added for each pair (r_i, r_j) of resources such that r_i and r_j are in R, unless the edge already exists in the graph.

4 Clustering Algorithm

The clustering algorithm at the core of our approach has to fulfill a set of requirements, the first of which is exploiting the density of the graph to enable the identification of clusters corresponding to highly connected areas of the graph. The second requirement is that the algorithm should not require the number of clusters as a parameter, as this information cannot be known prior to clustering in our context. Finally, resulting clusters provided by the algorithm should not necessarily be disjoint, as it is possible that two distinct resources in our initial graph belong to two different themes.

We have chosen the algorithm proposed by [1] and initially used in the domain of bioinformatics. It is a density-based algorithm producing possibly overlapping clusters.

The algorithm operates in three steps. First (i), it computes the weights of each node in the graph using the concept of k-core. Consider that the degree of a node is the number of his adjacent nodes. A k-core is a graph in which the minimal node degree is k. The weight of a node S_i is computed based on the highest possible k-core value in the subgraph composed of S_i and its adjacent nodes; once the weights have been computed, (ii) the nodes are explored in a descending order of their weights; each node S_i will initiate a cluster, and for each adjacent node S_j such as the difference between the weights of S_i and S_j is below a threshold t is assigned to the same cluster as S_i; finally, (iii) once all the nodes have been explored, the algorithm enriches the clustering by checking all the adjacent nodes for a given cluster; if for a node S_i in a cluster C_i, the subgraph composed of S_i and its adjacent nodes is highly connected, then all the adjacent nodes of S_i will also be added to C_i. This will enable nodes to be part of more than one clusters.

5 Labels Extraction

Goal of this step is to provide the user a view of the cluster content by extracting a set of relevant labels that describe the theme. The set of labels is extracted from the names of RDF resources is composed by the top-k keywords having the high weight in the cluster C_i. The weight w_{ij} of the keyword j (noted $keyword_j$) in the cluster C_i is computed according to the degree of the node j (noted $node_j$). We note that $keyword_j$ appears in the name of $node_j$. We give an example in Figure 4, where selected theme represents a set of researchers workings in the field of physics. The top-1 labels extracted using our approach is *"Physics"* as we can see in the name of the sub window of the figure. This label reflects the semantic content of the cluster. We can add more labels by increasing the value of k.

This approach can be extended to use more characteristics to calculate the weight of keyword by combining the degree of the node with the frequency of the keyword in the cluster. Castano et al. [2] use the most frequent keyword combining with the most frequent type of entities in the cluster. Another alternative would be to use an adaptation of the tf-idf function to determinate the weight w_ij. In this way, the relevant of the $keyword_j$ is proportional to its frequency of the keyword in the cluster C_i and its scarcity in other clusters C_k with $k \neq i$.

6 Our System

We have implemented a tool to support our approach for theme identification. The system requires two types of parameters: (i) clustering parameters, used to specify thresholds for assigning a node to a cluster, and (ii) semantic parameters, used to capture user preferences.

To illustrate the way our tool is used for theme identification, consider the following example of an RDF dataset extracted from DBPedia (see Figure 3). This dataset contains resources describing scientists working in different domains with their organizations and their countries. Assume that the user wants to

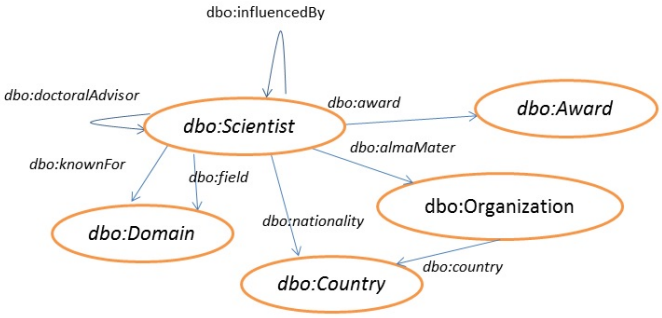

Fig. 3. Description of the RDF Dataset

identify themes in the input graph, and would like scientists from the same domain to be assigned to the same cluster. As the research domain is represented by the *dbo:field* property in our example, the user will indicate that two resources having the same value for this property should be assigned to the same cluster. He can repeat the clustering process either on the initial graph by adding new semantic parameters, or on a cluster obtained in previous iterations in order to get further details.

According to the preference set by the user, scientists of the same domain will be assigned to the same cluster. But it may happen that this property is not defined for some of the scientists in the dataset, and the user would therefore like to use another semantic criteria. For example, he could state that scientists related by the *dbo:doctoralAdvisor* property should be assigned to the same cluster.

Figure 4 shows the user interface of the system. The list of clusters is displayed on the left side and the initial RDF graph on the right side. The cluster selected in the list can be highlighted on the graph (green nodes) or opened as a new RDF graph. In Figure 4, the selected cluster represents the field of *Physics*.

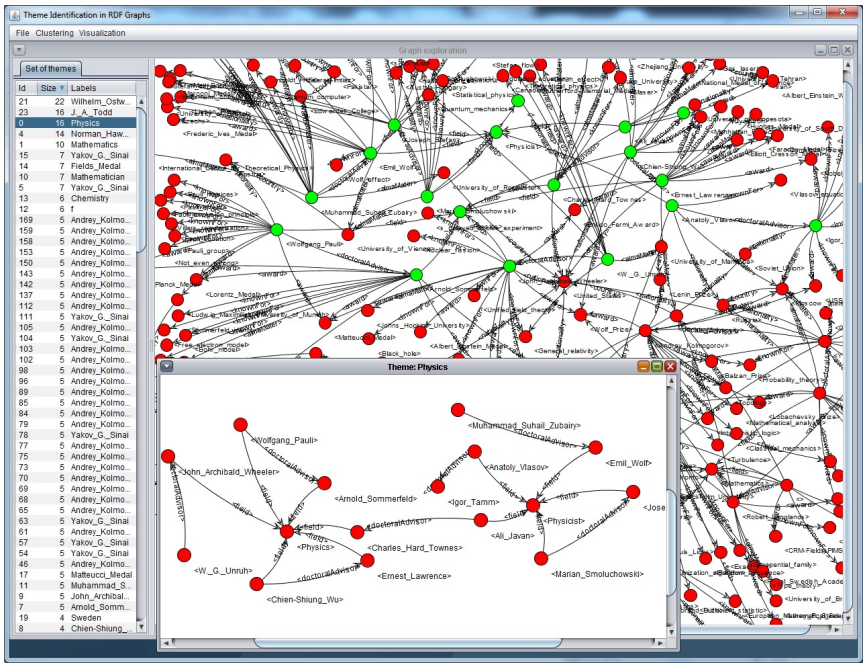

Fig. 4. Visualization of the themes

7 Related Works

Theme identification approaches have been proposed for text documents [8] or for other types of data published on the web, e.g. social networks [3], youtube documents [6] and DBpedia [7]. The goal of these approaches is to facilitate the search process and the navigation into the dataset. All of them use clustering techniques. Unlike our approach, they do not consider RDF datasets except the one described in [7]. Furthermore, they rely on text comparison to compute the distance between documents. This distance is used as the similarity measure for the clustering algorithm.

Despite the increasing amount of RDF(S)/OWL datasets available online, the problem of discovering themes have received little attention. Some works have focused on improving the quality of data by grouping resources to detect concepts and induce new classes or refine existing one [4, 5].

The closest work to ours is an approach for topic identification presented in [2]. It exploits a graph generated from an input RDF dataset, by adding new

edges between resources that have an important number of similar terms in their labels. A clustering algorithm is then applied to identify regions that are highly connected in the graph, which represent the topics. Similarly to our approach, this work is based on a clustering algorithm, but focuses only on identifying highly connected areas while we combine the density-based clustering process with semantic criteria capturing user preferences.

8 Conclusions

In this paper, we have proposed an approach for theme identification in RDF datasets. It combines a density-based clustering algorithm and semantic criteria capturing user preferences. Our approach comprises three stages: (1) preprocessing and capturing user preferences, (2) density-based clustering to form the clusters and (3) extraction of labels to describe the semantic of the cluster. Preprocessing consists mainly in simplifying the graph and removing the information which is not necessary for the clustering algorithm. Users' preferences are captured by mapping them into graph transformation primitives. Our approach differs from existing ones such as [2] in that it combines structural and semantic criteria for graph clustering. We have implemented a system for theme identification. Future works include the extension of the approach by improving label identification and providing the user with a summary of the clusters' content to describe its semantics. We are currently experimenting the use of our system on different RDF datasets in order to evaluate the precision of the clustering and the performances of the system.

References

1. G. Bader and C. Hogue. An automated method for finding molecular complexes in large protein interaction networks. *BMC bioinformatics*, 27:1–27, 2003.
2. S. Castano, A. Ferrara, and S. Montanelli. Thematic clustering and exploration of linked data. *Search Computing*, pages 157–175, 2012.
3. S. Castano, A. Ferrara, and S. Montanelli. Mining topic clouds from social data. *In Proceedings of the Fifth International Conference on Management of Emergent Digital EcoSystems (MEDES '13)*, pages 108–112, 2013.
4. K. Christodoulou, N. W. Paton, and A. A. A. Fernandes. Structure inference for linked data sources using clustering. In *Proceedings of the Joint EDBT/ICDT 2013 Workshops on - EDBT '13*, page 60. ACM Press, 2013.
5. N. Fanizzi, C. DAmato, and F. Esposito. Metric-based stochastic conceptual clustering for ontologies. *Information Systems*, 34(8):792–806, Dec. 2009.
6. U. Gargi, W. Lu, V. Mirrokni, and S. Yoon. Large-Scale Community Detection on YouTube for Topic Discovery and Exploration. *ICWSM*, pages 486–489, 2011.
7. R. Mirizzi and A. Ragone. Semantic wonder cloud: exploratory search in DBpedia. In *ICWE 2nd Int. Workshop on Semantic Web Information Management (SWIM 2010)*, pages 138–149, 2010.
8. H. Shahsavand Baghdadi and B. Ranaivo-Malançon. An Automatic Topic Identification Algorithm. *Journal of Computer Science*, 7(9):1363–1367, 2011.