Detection of Related Semantic Datasets Based on Frequent Subgraph Mining

Mikel Emaldi[1], Oscar Corcho[2], and Diego López-de-Ipiña[1]

[1] Deusto Institute of Technology - DeustoTech, University of Deusto, Bilbao, Spain
{m.emaldi,dipina}@deusto.es
[2] Ontology Engineering Group, Departamento de Inteligencia Artificial, Facultad de Informática, Universidad Politécnica de Madrid, Spain
ocorcho@fi.upm.es

Abstract. We describe an approach to find similarities between RDF datasets, which may be applicable to tasks such as link discovery, dataset summarization or dataset understanding. Our approach builds on the assumption that similar datasets should have a similar structure and include semantically similar resources and relationships. It is based on the combination of Frequent Subgraph Mining (FSM) techniques, used to synthesize the datasets and find similarities among them. The result of this work can be applied for easing the task of data interlinking and for promoting data reusing in the Semantic Web.

1 Introduction

Since the creation of Linked Open Data Cloud[3] initiative in 2007 with 12 datasets, to its last update in 2014 with 570 datasets, the number of Linked Datasets has grown enormously. This growth trend suggests that in few years, selecting appropriate datasets to link our datasets to, is going to become harder and harder. The same applies to the task of finding the dataset that may contain useful information for us, according to our needs. The work presented in this paper is focused on providing some steps forward into some of the aforementioned limitations: finding datasets to link to, finding datasets that provide support to our needs or understanding or summarizing datasets.

Our main contribution is an approach to find similarities among RDF datasets based on their graph structure, which can be used for solving the aforementioned problems. The main challenge that we have to deal with stems from the fact that, due to the size of many of the graphs derived from these RDF datasets, a direct comparison among their complete structure is not applicable. Therefore, a Frequent Subgraph Mining (FSM) based approach is proposed. FSM techniques, widely used in the domains of chemistry and biology to find similarities and correlations among different chemical compounds and molecules [2, 4, 11], allow extracting the most frequent subgraphs from a single graph or a set of graphs.

[3] http://lod-cloud.net/

Given that RDF datasets are graphs, we think that the combination of techniques based on the summarization of RDF graphs and the identification of the most frequent subgraphs can provide good results on finding related datasets.

An example of an application of the work proposed in this paper is related to dataset interlinking. As stated in section 2, when using existing dataset interlinking tools, the user have to select the input datasets whose links are going to be searched. Nowadays, there are two main approaches to select these datasets: applying the brute force for applying all the possible pairs of datasets to the interlinking tool; or requesting the user for selecting the most suitable datasets under her/his beliefs, a task that is becoming harder and harder because of the growth of the Linked Open Data Cloud. Proposed solution can ease this task suggesting a subset of related datasets, with the consequent reduction of the search space.

In summary, in this work a new approach for synthesizing and finding similarities among RDF datasets is presented. Specifically, this approach proposes the use of FSM techniques to synthesize these datasets. The approach proposed in this work can be used for easing the task of interlinking new datasets, and for improving data reuse through finding similar datasets.

The rest of the paper is organized as follows. In Section 2 the previous works in semantic dataset browsing, interlinking and summarization, and Linked Data source discovery are presented. In Section 3 some definitions and concepts about graph mining are explained. Section 4 describes our new approach based on FSM. In Section 5 proposed approach is evaluated against a set of datasets from Linked Open Data Cloud. At last, in Section 6, conclusions and future research challenges are explained.

2 Related Work

There are four research fields related to possibles usages of the work presented in this paper: semantic dataset browsing, interlinking and summarization, and Linked Data source searching.

Semantic Dataset Browsing. Works under this field provide search capabilities over linked datasets. From a set of terms, these browsers find resources in which these terms appear. Most of these works use techniques given from information retrieval field like TF-IDF (term frequency-inverse document frequency), and some works offer more complex techniques for refining the results. However, these works do not apply a previous filter on the datasets against they search the terms given by the user. Proposed work can be useful in this area when a term is found in a dataset, for prioritizing related datasets when searching for more results. In this field works like Swoogle [5], Falcons [3], Sindice [17] or Sig.ma [18] can be categorized.

Semantic Dataset Interlinking. The aim of works under this category is about given a pair of datasets, establishing links between them, based on a set

of rules defined by the user. Most of these works use different properties from resources within a dataset for establishing `owl:sameAs` links among them. One of the most important lacks of works in this area is that the user has to select the pair of datasets to establish new links between them. The solution proposed in this paper can be used to select these input datasets. Most remarkable works in this field are Silk [19] and LIMES [15].

Semantic Dataset Summarization. Although dataset summarization is not the final goal of this work, we have considered interesting to analyse most remarkable works in this field, although they are oriented for creating human-readable data summaries, instead of machine-readable summaries that are used in the proposed work. In [1], after detecting patterns in a graph, they extract labels from vertices and edges for elaborating a summary. [6] applies NER (Named Entity Recognition) techniques over literals of graphs for finding them in DBPedia. Once correspondent resources from DBPedia are found, they extract their categories for elaborating a summary.

Linked Data Source Searching These works try to find candidate datasets for interlinking. Works under this category are the most related with the work described in this paper. In [16], they extract literals from `rdfs:label`, `foaf:name` or `dc:title` properties. They search these literals in Sig.ma and group the results by source dataset. They consider that more instances a source has, more chances to be linked with original dataset has. The mayor weakness of this approach is that Sig.ma is no longer harvesting new data, so it is no a suitable solution for recently published datasets. [13] uses naive Bayes classifiers for establishing a ranking of related datasets based on correlations among them. Through this ranking the search space can be reduced. At last, in [14] they use already existing links for establishing new links among datasets. As previously mentioned, one of the objectives of our work was to solve the cold-starting problem when searching related datasets.

3 Background

The main objective of FSM is to extract all frequent subgraphs from a single graph or a set of graphs. We assume the definitions from [10]:

- **Labeled graph:** A labeled graph can be represented as $G(V, E, L_V, L_E, \varphi)$, where V is a set of vertices, $E \subseteq V \times V$ is a set of edges; L_V and L_E are sets of vertex and edge labels respectively; and φ is a label function that defines the mappings $V \rightarrow L_V$ and $E \rightarrow L_E$. G is a directed graph if $\forall e \in E$, e is an ordered pair of vertexes.
- **Subgraph:** Given two graphs $G_1(V_1, E_1, L_{V_1}, L_{E_1}, \varphi_1)$ and $G_2(V_2, E_2, L_{V_2}, L_{E_2}, \varphi_2)$, G_1 is a subgraph of G_2, if G_1 satisfies: *i)* $V_1 \subseteq V_2$, and $\forall v \in V_1, \varphi_1(v) = \varphi_2(v)$, and *ii)* $E_1 \subseteq E_2$, and $\forall (u, v) \in E_1, \varphi_1(u, v) = \varphi_2(u, v)$.

Multiple state-of-the-art tools implement FSM. In Table 1 a summary of the most relevant features of each solution is shown. These features are the following ones:

- **Single graph/Transactions:** according to [10] there are two different FSM problem formulations. In the first one, *single graph based FSM*, only a single very large graph is analyzed. In *graph transaction based FSM* the common substructures are extracted from a set of medium-size graphs (named *transactions*).
- **Directed graphs:** applying directionality to graphs increases computational cost considerably. For this reason, many of the solutions do not implement this feature.
- **Labeled vertexes:** solution allows (or not) labeled vertexes in input graphs.
- **Labeled edges:** solution allows (or not) labeled edges in input graphs.

As shown in Table 1 only SUBDUE and DPMine cover all features to be suitable for dealing with the characteristics of RDF graphs. As in this approach we want to extract the most common subgraph from each dataset, the solution that supports single graphs has been selected, i.e. SUBDUE.

Table 1: Summary of relevant features of each FSM solution. A complete description and comparison among them can be found at [10].

Solution	Single Graph / Transactions	Directed Graphs	Labeled Vertexes	Labeled Edges
SUBDUE	Single Graph	✔	✔	✔
AGM	Transactions	✔	✘	✔
FSG	Transactions	✘	✔	✔
DPMine	Transactions	✔	✔	✔
MoFA	Transactions	✘	✔	✘
gSpan	Transactions	✘	✔	✔
FFSM	Transactions	✘	✔	✔
GREW	Single Graph	✘	✔	✔
Gaston	Single Graph	✘	✔	✔
gApprox	Transactions	✘	✘	✘
(h/v)SiGraM	Single Graph	✘	✔	✔

Given a single, directed and labeled graph, SUBDUE [9] extracts the most frequent substructures. SUBDUE defines the most frequent subgraph as the subgraph that once replaced by a single node, compresses most the original graph. Assuming that G is the original graph, S is the candidate subgraph to be evaluated, $size(G)$ and $size(S)$ are the size of G and S respectively and $size(G|S)$ is the size of G compressed by S, the total compression rate can be calculated as:

$$value(S, G) = \frac{size(G)}{(size(S) + size(G|S))} \tag{1}$$

where:

$$size(G) = (|vertex(G)| + |edges(G)|) \tag{2}$$

SUBDUE can be parameterized to adapt its behavior to the different input graphs. To facilitate the understanding of the application of SUBDUE in Section 4 some of these parameters are explained:

- **inc:** this parameter allows the incremental analysis of large graphs, avoiding the consumption of all the memory of the system by large graphs and allowing the preview of partial results. To perform the incremental analysis, the input graph has to be split in different and numbered files. SUBDUE analyses these files in order, aggregating results of the files previously analyzed to the current file.
- **limit:** limits the number of candidate substructures that SUBDUE takes in consideration in each iteration. The default value is $\frac{|edges|}{2}$.
- **prune:** prunes the graph discarding useless substructures.

4 Frequent Subgraph Mining Approach

4.1 RDF Graph Synthesis Model

In order to apply the RDF graph synthesis model presented in this paper, some modifications have been done to the original RDF graph model. It is important to note that these transformations do not preserve the meaning of the RDF model, but we do not consider that property important for our approach. However, after applying these transformations, a proper interpretation of the graph can be done. As can be seen in this section, the aim of these transformations is to simplify the graph for easing the task of extracting the most common substructures. From the triples shown in Listing 1, represented graphically in Figure 1, transformations are applied to ensure the correct understanding of the presented model.

The first transformation applied to RDF graphs consists in replacing URIs from the subjects of resources from datasets. Since they are unique identifiers of resources, URIs in subjects will generate a large amount of unique nodes which do not belong to any candidate substructure, increasing the difficulty of finding frequent subgraphs. To avoid this, these URIs have been replaced by the ontological class (or classes, if it is represented by more than one class) of the resource represented by the `rdf:type` property if any, as can be seen in Figure 2. If a resource has no a `rdf:type` predicate associated, this resource is discarded.

The next transformation is about managing interlinked resources. Establishing links among resources from different datasets is one of the most important features in Linked Data publication. For this reason, a large amount of internal and external links can be found in linked datasets. Managing external links, adds to the computational cost generated by the analysis of each triple, the delay generated by retrieving the information pointed by them through the Web. Furthermore, one of the challenges of this work was to solve the cold start problem when looking for related datasets. For these reason, external links have been

```
@prefix :        <http://example.org/resource/> .
@prefix foaf:    <http://xmlns.com/foaf/0.1/> .
@prefix rdf:     <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix aktors:  <http://www.aktors.org/ontology/portal/> .

:Tim_Berners-Lee rdf:type      foaf:Person ;
                 foaf:name     "Tim Berners-Lee" ;
                 foaf:mbox     "timbl@w3.org" ;
                 foaf:homepage <http://www.w3.org/People/Berners-Lee> .

:pub1 rdf:type            aktors:Article-Reference ;
      aktors:has-title    "The Semantic Web" ;
      dcterms:creator     :Tim_Berners-Lee ;
      aktors:published-by "Scientific American" .
```

Listing 1: RDF triples used in the example model.

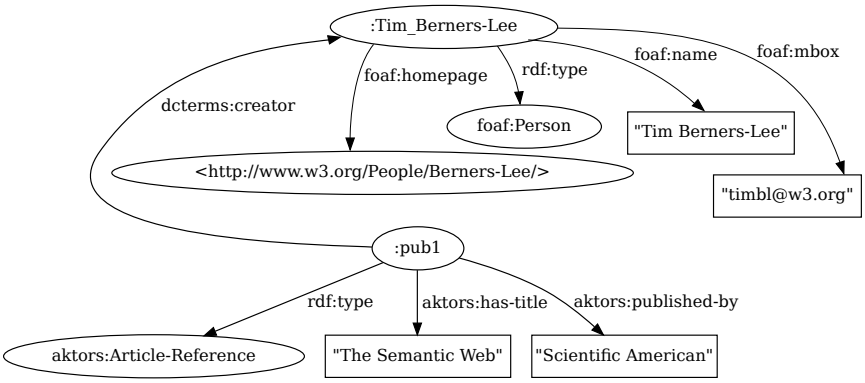

Fig. 1: Resultant graph from the triples in Listing 1.

removed. Despite this, in Section 6 the influence of existing links is briefly analyzed. In Figure 3, the resulting model after elimination of external links can be seen. In this model a structure representing a publication and its author can be seen, as a synthesis of triples presented in Listing 1.

Regarding to the literals, although they have been maintained in proposed model, there are no literals in any of most frequent structures extracted during the evaluation. The explanation of this situation is similar to the explanation given about the URIs in subjects: with so much variety of different literals, the probability to form part of a candidate substructure is minimal.

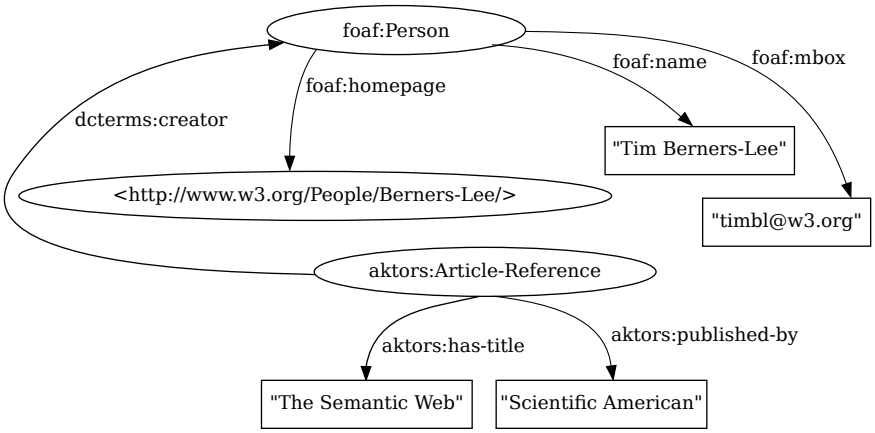

Fig. 2: Resultant graph after replacing URIs with the ontological class of the resource.

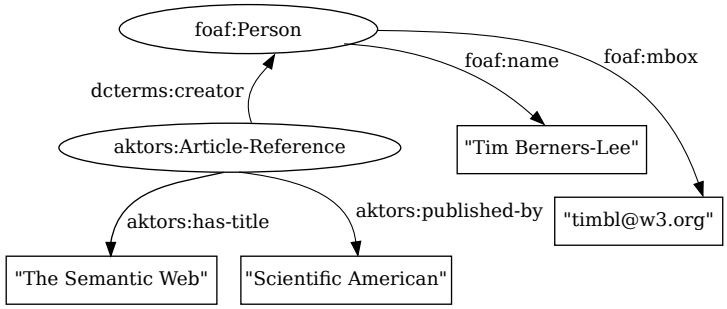

Fig. 3: Resultant graph after removing external links.

4.2 Extraction and Comparison of Most Frequent Subgraphs

We apply SUBDUE to the graph obtained as a result of the previous transformations. According to [10], SUBDUE's runtime and resource consumption do not grow linearly with the size of the input graphs, making it hard to do an estimation of the total runtime or knowing whether the process is going to finalize in a reasonable amount of time. The ideal parameters for getting a balance between affordable runtime and obtaining an appropriate number of candidate substructures are still subject of experimentation, but limiting the number of candidate substructures to 5, applying the incremental analysis capabilities and pruning the input graph seem to be appropriate parameters to start finding this balance.

Once the most frequent substructures from different datasets are extracted, the comparison among them has been done through SUBDUE's `gm` (Graph Matcher). Given a pair of graphs, this utility computes the cost of transforming the largest graph into the smallest one, returning the number of transformations

done. In this case, all transformations (addition, removal or replacing of a node) have the same cost. As this number of transformations is not normalized by default (it depends on the size of the input graphs), the normalization shown in Equation 3 has been applied to the result. Finally, the similarity between both substructures is calculated as can be seen in Equation 4.

$$Cost_{normalized} = \frac{Cost}{|vertexes_{largestGraph}| + |edges_{largestGraph}|} \tag{3}$$

$$Similarity = 1 - Cost_{normalized} \tag{4}$$

4.3 Implementation

This work has been implemented following the workflow explained next. The different stages of this workflow have been implemented as independent tasks:

- **Generation of IDs and replacement of subjects:** as can be seen in listing 2, SUBDUE has its own format for representing graphs. This format requires to assign an unique ID to each vertex. At this first step, the RDF graphs are iterated, replacing the subject of each resource by its ontological class if the property `rdf:type` is presented and an unique and consecutive IDs are assigned to each generated vertex.
- **SUBDUE file generation:** once the IDs are assigned, the relationships among generated vertices are analysed in order to generate edges. Once these edges are generated, the final SUBDUE file of each graph is generated.
- **Most frequent subgraph extraction:** at these step the most frequent subgraph of each RDF graph is extracted with SUBDUE and previously generated input files.
- **Graph matching:** at last, similarities are among these most frequent subgraphs are found with SUBDUE's Graph Matching (`gm`) tool.

The implementation of this work and baselines (subsection 5.2) can be found at `https://github.com/memaldi/lod-fsm`.

5 Evaluation

Presented approach has been evaluated against datasets from Linked Open Data Cloud. The development of the evaluation follows these steps. First, a gold standard has been created for determining the effectiveness of both developed system and baseline solutions in terms of precision and recall. These baseline solutions (or baselines) are simple solutions that solve proposed problem in a simple way, with the aim of establishing a baseline to be surpassed by the new solution. At last, the results given from proposed solution are compared with the results given by baseline solutions. The evaluation has been done only in terms of efficacy because the developed work has been designed to be launched in batch and without the interaction of the end-user, so that, the efficiency is not considered a key factor to be evaluated.

```
v 1 foaf:Person
v 2 "Tim Berners Lee"
v 3 "timbl@w3.org"
v 4 aktors:Article-Reference
v 5 "The Semantic Web"
v 6 "Scientific American"
e 1 2 foaf:name
e 1 3 foaf:mbox
e 4 1 dcterms:creator
e 4 5 aktors:has-title
e 4 6 aktors:published-by
```

Listing 2: Representation of the graph from figure 3 in SUBDUE. In files 1-6 the vertices are represented while in files 7-11 the edges are represented.

5.1 Gold Standard

For constituting the gold standard, two different sources have been checked. The first source, inspired by [13], consists on checking already existing links among datasets used in this evaluation. The links among these datasets have been extracted through the property `links:<target_dataset_id>` from The Datahub[4] entry of each dataset, as this property is requested for publishing datasets in the LOD Cloud. But, when evaluating the proposed solution, many links that are not described in The Datahub were discovered. These links could not appear in The Datahub for many reasons: related dataset have been published after the publication of the source dataset and the publisher has not checked them, or simply, the publisher did not know the existence of these related datasets. The absence of these valid links could provoke a situation in where the developed system could recommend datasets that, in fact, are valid results but considered as false positives by the gold standard.

To solve this issue, a second source have been used to form the gold standard. This source consisted on surveying different researchers on Semantic Web and Linked Data for determining the validity of these new relations among datasets. These surveys have been performed through a web application[5] that shows to researchers different pairs of datasets, to determine if there was any possible relationship between them. These datasets were represented by the title, description and resources published in their The Datahub's entry. Three options were allowed for each pair of datasets: "yes" if they consider that there was a possible relationship between them, "no" if they consider the opposite, and "undefined" if they were not sure about the possible relationships. Each pair of datasets have been evaluated by three different researchers. This approach arises another issue: the number of different pairs resulting from the combination of all the datasets

[4] `http://datahub.io`
[5] `https://github.com/memaldi/ld-similarity-survey`

employed during the evaluation ups to 2,346[6]. Considering that each pair have to be evaluated three times, this number increases to 7,038 evaluations to be done by selected researchers. Considering that this number of evaluations is too high, the number of dataset pairs have been reduced considering the evidence proposed by [6]. The authors of this work consider if a pair of datasets have common links to the same datasets, they could be related. From these evidence, only datasets that are linked to common datasets have been included, reducing the number of evaluations to 594. Once all the evaluations have been done, the Fleiss' Kappa [7] coefficient reveals an agreement among the reviewers of 41%, which means a moderate agreement according to [12]. At last, for constituting the gold standard, relations extracted from The Datahub have been complemented with relations which in the survey have been approved by at least two reviewers.

At the time of writing, an unique gold standard has been created for evaluating the developed system. However, a more suitable solution could be to develop a different gold standard depending on the topic of the datasets whose similarities are going to be extracted (biology, statistical government data, academical publications, etc.). This work is going to be attempted in the future work.

5.2 Baselines

For weighting the results given by proposed solution, three baselines have been developed. The first baseline is based on the evidence that as more ontologies are shared between a pair of datasets, more related they are. The relation degree between a pair of datasets is calculated as follows, being N the set of ontologies used to describe the dataset D:

$$score(D1, D2) = \frac{N_1 \cap N_2}{max(|N_1|, |N_2|)} \tag{5}$$

The second baseline, similarly to the first one, takes the common ontologies between a pair of datasets to establish their relation degree, but establishing a ranking based on the usage of the classes and properties of each ontology used within each dataset. The distance between different pair of rankings have been calculated through a normalized Kendall's Tau:

$$K(\tau_1, \tau_2) = \sum_{i,j \in P} \bar{K}_{i,j}(\tau_1, \tau_2) \tag{6}$$

At last, the third baseline calculates the relation degree between a pair of datasets calculating the Jaccard distance among all the triples of each dataset. Being T_1 and T_2 the pair of datasets to be compared, the Jaccard distance is calculated as follows:

$$d_J(T_1, T_2) = \frac{|T_1 \cup T_2| - |T_1 \cap T_2|}{|T_1 \cup T_2|} \tag{7}$$

[6] The complete list of used datasets can be found at `http://apps.morelab.deusto.es/iesd2015/datasets.csv`

5.3 Results

In figure 4, the results of both proposed solution and baselines are shown, in terms of precision, recall, F1-score and accuracy. As can be seen, in terms of precision, the proposed solution clearly overcomes the baselines, overpassing a value of 0.8 from a threshold of 0.4; reaching a maximum value of 0.9. On the other hand, the maximum value of recall is about 0.51, decaying from a threshold of 0.3, offering a result that is not as good as expected and being surpassed by one of the baselines. This situation is promoted by the fact that higher the threshold is, the requested similarity between pair of graphs is higher too. Thus, there are pairs of datasets detected as related by our solution but their relation degree is not as high as expected. These results show that recommendations done by the proposed solution are valid in a high percentage (low number of false positives), although there still are many related datasets that the solution omits.

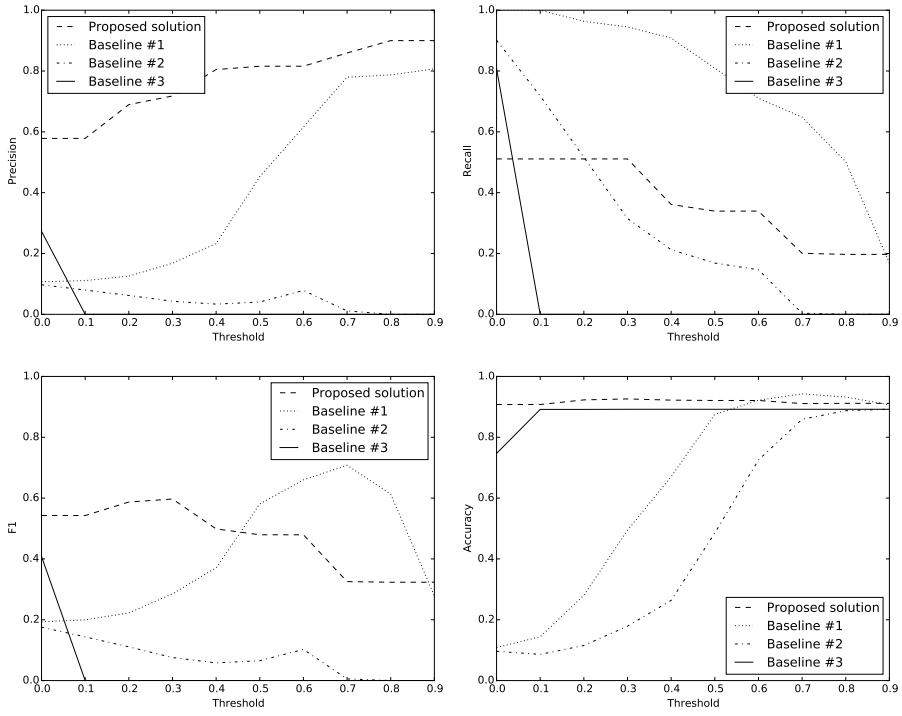

Fig. 4: Comparison among the results obtained by proposed solution and baselines.

Regarding to the good results obtained by the first baseline, there is a clarification to be done. As can be seen, many datasets used in the evaluation where produced by the RKB Explorer project [8]. These datasets have been published

using the same ontologies and methodology, so they share the same ontologies in similar proportion. As exposed in section 6, providing a more diverse evaluation set is one of the key tasks for the future work.

6 Conclusion and Future Work

In this work, a solution for recommending related datasets and easing the task of dataset linking has been presented. As exposed in Section 5, proposed solution provides precise recommendations of candidate datasets to be linked. Although the recall is not as good as expected, given that nowadays the help that a data publisher has at time of selecting related datasets for linking his datasets is very limited, we consider that is more important to recommend valid candidate datasets for interlinking, although these datasets are not all the available datasets. However, the results given by the recall are an issue in which we are currently working. At the present time, for avoiding false negatives provoked by related datasets described by different ontologies, string similarity techniques are being introduced, achieving an increase of recall between 0.10 and 0.30 regarding to the work exposed in this paper. Another task to be done in the future work is to analyse how the links generated by the own system can be used for improving the results in an iterative way. At last, regarding to the evaluation, an important future task is to include more diverse datasets in the evaluation set for avoiding the overfitting of the proposed model or any of the baselines and developing a different topic-based gold standard.

In conclusion, the promising results obtained show that most frequent subgraph mining techniques can be used to ease the task of interlink datasets from the Semantic Web.

Acknowledgments. This work has been developed within WeLive project, founded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 645845.

References

1. Böhm, C., Kasneci, G., Naumann, F.: Latent topics in graph-structured data. In: Proceedings of the 21^{st} ACM international conference on information and knowledge management. pp. 2663–2666. ACM (2012)
2. Borgelt, C., Berthold, M.R.: Mining molecular fragments: Finding relevant substructures of molecules. In: Proc. of the 2002 IEEE International Conference on Data Mining. pp. 51–58 (2002)
3. Cheng, G., Qu, Y.: Searching linked objects with Falcons. International Journal on Semantic Web and Information Systems 5(3), 49–70 (2009)
4. Dehaspe, L., Toivonen, H., King, R.D.: Finding frequent substructures in chemical compounds. In: Proc. of the 4th International Conference on Knowledge Discovery and Data Mining. vol. 98 (1998)

5. Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R., Peng, Y., Reddivari, P., Doshi, V., Sachs, J.: Swoogle: A Semantic Web search and metadata engine. In: Proceedings of the 13^{th} ACM Conference on Information and Knowledge Management. pp. 652–659 (2004)
6. Fetahu, B., Dietze, S., Nunes, B.P., Casanova, M.A., Taibi, D., Nejdl, W.: A scalable approach for efficiently generating structured dataset topic profiles. In: The Semantic Web: Trends and Challenges, pp. 519–534. Springer (2014)
7. Fleiss, J.L.: Measuring nominal scale agreement among many raters. Psychological bulletin 76(5), 378 (1971)
8. Glaser, H., Millard, I.: RKB Explorer: Application and infrastructure. In: Proceedings of the Semantic Web Challenge, in conjunction with the 6^{th} International Semantic Web Conference (2007)
9. Holder, L.B., Cook, D.J., Djoko, S.: Substructure discovery in the SUBDUE system. In: Proc. of the AAAI Workshop on Knowledge Discovery in Databases. pp. 169–180 (1994)
10. Jiang, C., Coenen, F., Zito, M.: A survey of frequent subgraph mining algorithms. The Knowledge Engineering Review 28(1), 75–105 (2013)
11. Kramer, S., Pfahringer, B., Helma, C.: Mining for causes of cancer: machine learning experiments at various levels of detail. In: Proc. of the 3th International Conference on Knowledge Discovery and Data Mining. pp. 223–226 (1997)
12. Landis, J.R., Koch, G.G.: The measurement of observer agreement for categorical data. Biometrics 33(1), 159–174 (1977)
13. Leme, L.A.P.P., Lopes, G.R., Nunes, B.P., Casanova, M.A., Dietze, S.: Identifying candidate datasets for data interlinking. In: Proceedings of the 13^{th} International Conference on Web Engineering. pp. 354–366. Springer (2013)
14. Lopes, G.R., Leme, L.A.P.P., Nunes, B.P., Casanova, M.A., Dietze, S.: Recommending tripleset interlinking through a social network approach. In: Proceedings of the 14^{th} international conference on Web Information Systems Engineering. pp. 149–161. Springer (2013)
15. Ngomo, A., Auer, S.: LIMES: a time-efficient approach for large-scale link discovery on the web of data. In: Proceedings of the 22^{nd} International Joint Conference on Artificial Intelligence. pp. 2312–2317 (2011)
16. Nikolov, A., d'Aquin, M.: Identifying relevant sources for data linking using a Semantic Web index. In: Proceedings of the Linked Data on the Web workshop in conjunction with the 20^{th} international World Wide Web conference. vol. 813 (2011)
17. Tummarello, G., Delbru, R., Oren, E.: Sindice.com: Weaving the Open Linked Data. In: Proceedings of the 6^{th} International Semantic Web Conference and the 2^{nd} Asian Semantic Web Conference. pp. 552–565 (2007)
18. Tummarello, G., Cyganiak, R., Catasta, M., Danielczyk, S., Delbru, R., Decker, S.: Sig.ma: Live views on the web of data. Web Semantics: Science, Services and Agents on the World Wide Web 8(4), 355–364 (Nov 2010)
19. Volz, J., Bizer, C., Gaedke, M., Kobilarov, G.: Silk: A link discovery framework for the web of data. In: Proceedings of the Linked Data on the Web workshop in conjuntion with the 18^{th} international World Wide Web conference. vol. 583 (2009)