SumMER: Summarizing RDF/S KBs using Machine LEaRning[1]

Georgia Eirini Trouli[1], Georgia Troullinou[2], Lefteris Koumakis[2], Nikolaos Papadakis[1] and Haridimos Kondylakis[1,2]

[1]Department of Electrical and Computer Engineering, Hellenic Mediterranean University, Heraklion, Crete, Greece
ddk7@edu.hmu.gr, npapadak@cs.hmu.gr, kondylak@ics.forth.gr
[2]Institute of Computer Science, FORTH, Heraklion, Crete, Greece
troulin@ics.forth.gr, koumakis@ics.forth.gr

Abstract. Knowledge graphs have now become common on the web, ranging from small taxonomies for categorizing web sites, to large knowledge bases that contain a vast amount of structured content. To enable their quick understanding and exploration semantic summaries have been proposed. A key issue of structural semantic summaries is the identification of the most important nodes. Works in the area, usually employ a single centrality measure, capturing a specific perspective on the notion of a node's importance. However, combining multiple centrality measures could give a more objective view, on which nodes should be selected as the most important ones. In this paper, we present SumMER, a novel framework that explores machine learning techniques for optimally combining multiple centrality measures for selecting the most important nodes. The experiments performed show the benefit of our approach, effectively increasing the quality of the generated summaries.

1 Introduction & Solution

The explosion of the Data Web and the associated Linked Open Data (LOD) initiative have led to the generation of an enormous amount of RDF datasets that are currently widely available [1], [2]. These datasets often have extremely complex schemas, which are difficult to comprehend, limiting the exploitation potential of the information they contain. Semantic summarization has been recognized as an important tool to facilitate ontology understanding, further supporting ontology exploration and reuse. A semantic summary, according to our recent survey [3] is "a compact information extracted from the original graph, offering a way to extract meaning from data while reducing its size, and/or a graph, which some application can exploit instead of the original graph" to perform certain tasks more efficiently like query answering, source selection etc.

In this paper, we focus on structural summarization methods, which consider first and foremost the graph structure, in order to generate summaries. Recent, state of the

art, structural non-quotient summarization methods [3],[4],[5],[6],[7],[8],[9], [10] separate between the schema and the instance graph of an RDF/S KB, as the schema graph offers a first natural way to provide an overview of the KB contents – even when the schema graph is not available, schema discovery tools can be used to discover it [11], [12]. Then to proceed with the summarization task, state of the art works select the most important nodes of an RDF/S schema graph, based on an importance measure, and then link those nodes using various algorithms in order to generate a connected subgraph out of the original one.

The problem. For generic graphs, multiple centrality measures have been proposed, each one perceiving importance using different criteria. However, there is no centrality measure to dominate them all, and each one is appropriate for different notions of importance over different types of graphs. On the other hand, we have already shown that several of these centralities interrelate [7], whereas there have been approaches that exploit graph neural networks for estimating node importance [13]. Existing approaches on structural summarization, in most of the cases select a single (or just a few) centrality measure(s) that produce the best results for selecting the most important nodes for a specific ontology. However, despite the fact that centrality measures offer a complementary view on node's importance, to the best of our knowledge, so far there is no mechanism able to exploit them all.

Our solution. We argue that combining multiple such measures could give us an objective view on which nodes should be selected as the most important ones. To this direction, in this paper, we present SumMER, effectively exploiting machine-learning algorithms for optimally combining multiple importance measures for node selection. To the best of our knowledge, no other approach so far combines structural summarization techniques with machine learning for RDF/S KBs. More specifically, for generating a summary using SuMMeR, we follow the three steps, shown in Fig. 1. The first two steps are trying to identify the top-*k* most important schema nodes, whereas the last one focuses on linking the selected schema nodes, possibly introducing additional nodes to the schema summary.

Selecting top-k nodes. The first step in identifying the top-*k* nodes in G_S is to calculate for each node its importance in the graph. As already mentioned, multiple graph centrality measures have been proposed in the literature, each one capturing a different perspective on the node's importance. In this work we do not try to identify an optimal

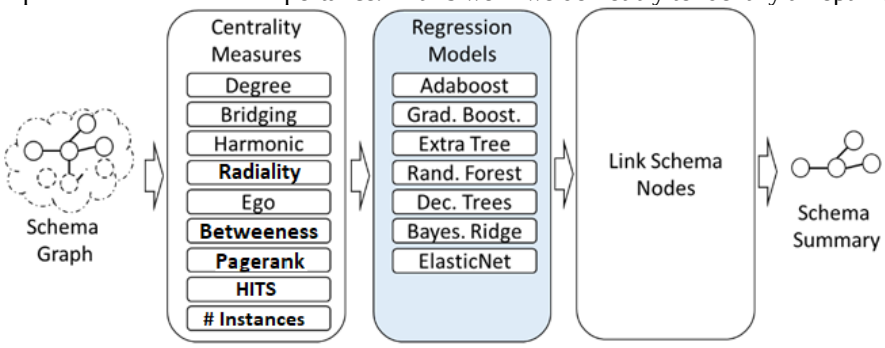

Fig. 1. SuMMeR's workflow.

centrality measure, as we believe that they offer different perspectives on a node's importance and that ideally, they should be all considered for assessing a node's importance. To this direction we exploit a diverse set of centrality measures that we calculate for each node (i.e. degree, bridging, harmonic, radiality, ego, betweenness, PageRank and HITS), shown in Fig.1. Note that many of those measures correlate as we have already shown [7] and as such within this step we select the ones not correlated (in bold) to be further exploited as features for the subsequent machine learning phase.

To explore the combination of multiple centrality measures for identifying the top-*k* schema nodes, we model the problem as a regression problem, trying to rank all schema nodes for selecting the top-*k* ones. In this paper, we explore the following machine learning algorithms: Adaboost regressor, Gradient Boosting regressor, Extra Tree regressor, Random Forest regressor, Linear regression, Decision Tree regressor, Bayesian Ridge and ElasticNet. As such for each schema node we construct a vector with the selected centrality measures as features, trying to identify the top-k most important nodes.

Linking schema nodes. Independent of the way the most important nodes are selected, the next step is to link those nodes formulating a connected schema subgraph. Similarly, to [7], we perceive this problem as a variation of the well-known Graph Steiner-Tree problem, trying to minimize the additional nodes introduced for connecting the top-*k* most important nodes.

2 Preliminary Evaluation

Next, we present an overview of the datasets used and the methodology for our experimentally evaluating the constructed summaries.

Datasets. For evaluating our approach, we use DBpedia v3.8, DBpedia v3.9, and the Semantic Web Dog Food (SWDF). For those versions we also have available query logs containing 50K user queries for v3.8, 110K user queries for v3.9 and 2.5K user queries, for SWDF provided by LSQ (https://aksw.github.io/LSQ/) that we exploit for evaluation as we shall see in the sequel. Table 1 summarizes the characteristics of the three ontology versions we use for our evaluation.

Table 1. Ontology Characteristics.

	Classes	Properties	User Queries	Storage
DBPedia 3.8	315	1323	50K	103 GB
DBPedia 3.9	497	1805	110 K	114 GB
SWDF	120	72	2.5 K	50 MB

Competitors. As ML has not been previously used for generating structural summaries we compare our approach with RDFDigest+, the latest approach for generating structural summaries that has been shown to outperform past approaches [5], [7].

Constructing a "golden standard". In order to construct a "golden standard" for the most important nodes, and to evaluate the regression models, we exploit the query logs for the three available ontology versions, calculating the schema nodes that are

more frequently queried. We assess as the most important, the ones that have a higher frequency of appearance in the queries.

Metrics. For evaluating the performance of our machine learning algorithms, we used Mean Absolute Error (MAE) as commonly used for evaluating regression problems. However, note that as we are only looking for the top-*k* nodes, we evaluate those metrics on the aforementioned *k* nodes only. In addition, we calculate for each summary its *coverage*, i.e. we calculate for each query the percentage of the classes and properties that are included in the summary. Having the percentages of the classes and properties included in the summary, the *query coverage* is the weighted sum of these percentages. As our summaries are node based we give 0.8 weight to the percentage of the classes and 0.2 weight to the percentage of the properties.

Experiments. For the evaluation of the node selection of the various algorithms, we attempt to predict the top 10%, 15%, 20%, 25%, 30% of nodes to be included in the summary. For all the experiments, we use the DBpedia v3.9 as the training dataset and the DBpedia v3.8 and SWDF as the test datasets. We perform a feature selection step where we select the non-correlated centrality measures (betweenness, radiality, page rank, hits and instances), we train the selected models on DBpedia v3.9 and then we evaluate the train versus test set. We use 10-fold cross validation for the training dataset.

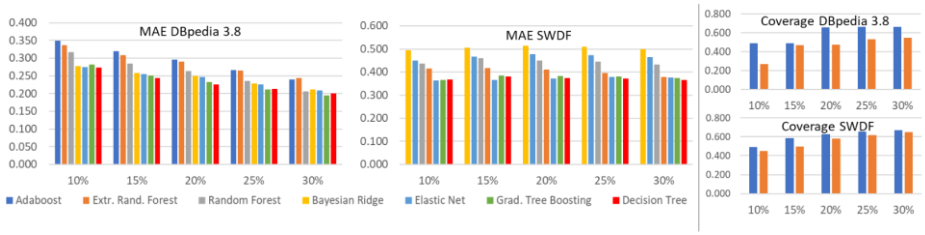

Fig. 2. MAE and Coverage for testing on DBpedia 3.8 and SWDF.

The results are shown in Fig. 2, and as presented, the Decision Tree regressor performs best in almost all cases, whereas most of the algorithms show a relatively good performance. Looking at the confusion matrices (not presented here due to lack of space) we can identify that the Decision Tree regressor is able to predict best the true positives in all summary sizes, outperforming among others the RDFDigest+ in all cases. The good performance on selecting the top-k nodes is also depicted in the subsequent calculation of the coverage for both DBpedia 3.8 and SWDF. As shown in Fig. 2 (right) SumMER is always better than RDFDigest+.

3 Conclusions

Overall, the results show that our approach is able to generate better summaries, that are able to answer more query fragments than previous works. This is true, not only in subsequent versions of the same ontology (DBpedia), but also in completely different ontologies (SWDF), showing that our approach is able to generalize into semantic graphs with different structure. Overall, Decision Trees Regression has been identified as the best performing algorithm with stability over the different KBs used. For future

work, we intend to exploit machine-learning methods learning to rank [14], and also explore methods for personalizing summaries based on user input. An interesting idea would be also to explore deep learning methods for generating summaries.

Acknowledgements

This research project was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the "2nd Call for H.F.R.I. Research Projects to support Post-Doctoral Researchers" (iQARuS Project No 1147).

References

1. Troullinou, G., Kondylakis, H., Lissandrini, M., Mottin, D.: SOFOS: Demonstrating the Challenges of Materialized View Selection on Knowledge Graphs, SIGMOD (2021), 2789–2793.
2. Agathangelos, G., Troullinou, G., Kondylakis, H., Stefanidis, K., Plexousakis, D: RDF Query Answering Using Apache Spark: Review and Assessment. ICDE Workshops (2018) 54-59.
3. Cebiric, S., Goasdoué, F., Kondylakis, H., et al.: Summarizing semantic graphs: a survey. The VLDB Journal (2019), 28(3), 295-327.
4. Pouriyeh, S., Allahyari, M., Liu, Q., et al.: Ontology Summarization: Graph-Based Methods and Beyond. Int. J. Semantic Comput. (2019),13(2), 259-283.
5. Troullinou, G., Kondylakis, H., Stefanidis, K., Plexousakis, D.: Exploring RDFS KBs Using Summaries. International Semantic Web Conference (2018), 268-284.
6. Troullinou, G., Kondylakis, H., Stefanidis, K., Plexousakis, D.: RDFDigest+: A Summary-driven System for KBs Exploration. International Semantic Web Conference (P&D/Industry/BlueSky), 2018.
7. Pappas, A., Troullinou, G., Roussakis, G., Kondylakis, H., Plexousakis, D.: Exploring Importance Measures for Summarizing RDF/S KBs. In ESWC, 2017, 387-403.
8. Vassiliou, G., Troullinou, G., Papadakis, N., Stefanidis, K., Pitoura, E., Kondylakis, H: Coverage-Based Summaries for RDF KBs. ESWC (Satellite Events) 2021: 98-102.
9. Troullinou, G., Kondylakis, H., Lissandrini, M., Mottin, D.: SOFOS: Demonstrating the Challenges of Materialized View Selection on Knowledge Graphs. SIGMOD Conference 2021: 2789-2793.
10. Vassiliou, G., Troullinou, G., Papadakis, N., Kondylakis, H.: WBSum: Workload-based Summaries for RDF/S KBs. SSDBM (2021).
11. Kellou-Menouer, K., Kardoulakis, N., Troullinou, G. et al.: A survey on Semantic Schema Discovery, The VLDB Journal (2021) (in press).
12. Kardoulakis, N., Kellou-Menouer, K., Troullinou, G., et al.: HInT: Hybrid and Incremental Type Discovery for Large RDF Data Sources. SSDBM (2021), 97–108.
13. Park, N., Kan, A., Dong, X.L., Zhao, T., Faloutsos, C.: Estimating Node Importance in Knowledge Graphs Using Graph Neural Networks. KDD, 2019, 596-606.
14. Learning to Rank, Available online: https://en.wikipedia.org/wiki/Learning_to_rank (visited May 2020)