Annotation-Based Method for Linking Local and Global Knowledge Graphs

Patrícia Cavoto[1], André Santanchè[1]

[1]Laboratory of Information Systems (LIS)
Institute of Computing (IC)
Univeristy of Campinas (UNICAMP)
Campinas – SP – Brazil

patricia.cavoto@gmail.com, santanche@ic.unicamp.br

Abstract. *In the last years, the use of data available in "global graphs" as Linked Open Data and Ontologies are increasing faster and bringing with them the popularization of the graph structure to represent information networks. One challenge, in this context, is how to link local and global knowledge graphs. This paper presents an approach to address this problem through an annotation-based method to link a local graph database to global graphs. Different from related work, the local graph is not derived from a static dataset, but it is a dynamic graph database evolving along the time, containing connections (annotations) with global graphs that must stay consistent during its evolution. We applied this method over a dataset with more than 44,500 nodes, annotating them with the values found in DBpedia and GeoNames. The proposed method is an extension of our ReGraph[1] framework that bridges relational and graph databases, keeping both integrated, synchronized and in their native representations, with minimal impact in the current infrastructure.*

1. Introduction

Real-world phenomena as biological processes, social networks and information systems have been increasingly modeled as networks, where nodes can represent individuals, computers, species, proteins, etc. and links the interaction among them. Recent research are pointing graphs as the fitted structure to store this kind of data, in which the relations among data elements are as important as the elements themselves. In the biology field, there are many uses for graphs, including metabolic networks, chemical structures and genetic maps [Vicknair et al. 2010]. The challenge is how to explore the network "behind" data available in existing information systems for analysis when data is stored in formats that do not valorize such network structure.

This challenge motivated our proposition of ReGraph, a framework inspired in the OLAP approach, which creates a special local graph database designed for network-driven analyses, aligned with an existing relational database. We applied ReGraph to taxonomic data from FishBase[2] to create FishGraph [Cavoto et al. 2015].

[1] http://patricia.cavoto.com.br/regraph/
[2] http://www.fishbase.org/

In this paper, we present an automatic annotation-based method to link our local graph database to global graphs from the Semantic Web, applied to link FishGraph data with DBpedia. Our method contributes in the data quality analysis, in the enrichment of the local database and in building the Giant Global Graph.

This is a work in progress concerning how to relate data from a local graph, stored in a graph database, with global graphs. Different from related work, our local data repository is not a static set of documents or tags to be enriched, but a dynamic graph database. It annotated content evolve along the time, bringing challenges, addressed in this research, of how to manage this hybrid graph (local and global) maintaining its consistency during the evolution.

The remainder of the paper is organized as follows. Section 2 presents the related work. Section 3 details our ReGraph framework. Section 4 presents our annotation-based approach to enrich data using ontologies. Section 5 presents our conclusions and future work.

2. Related Work

There are several contexts in which annotations are related to the Semantic Web resources (LOD and ontologies). The annotations are produced manually, semi-automatically or automatically, helping the improvement of information retrieval, knowledge reuse and information exchange [Oren et al. 2006]. There are works proposing annotations over wiki pages [Oren et al. 2006] and publishing personal notes as linked data in semantic blogs [Drăgan et al. 2010].

Several initiatives focus in how to reach semantic concepts to relate them to resources. In a survey of semantic search approaches, the authors present an overview and a classification of the existing methods for searching and browsing linked data and ontologies [Mangold 2007]. In [Alm et al. 2014] the authors propose a model to extract characteristic features from semantic annotations by importing the ontology concepts and their taxonomic relationships. Another work uses taxonomic distance measures to compute relatedness of the ontological annotations [Palma et al. 2014].

The work presented in [Santos et al. 2011] propose an architecture to discover information sources through the use of semantic search techniques in a corporative metadata repository. The process begins with an initial keyword list, followed by the query reformulation process that expands this list, adding semantically related terms and creating a new query to run on semantic annotations.

In [Amanqui et al. 2013], the authors developed a semantic search application that uses semantic web key concepts for information retrieval. They have proposed an architecture for semantic search that maps concepts of the OntoBio domain ontology to a database from the National Institute for Amazonian Research (INPA), which has collections of insects, fishes, and mammals, totalizing over 16,500 species.

As mentioned before, this work differs since it introduces a graph database perspective over the locally annotated data, which dynamically evolve along the time and must stay consistent.

3. ReGraph

As mentioned before, this method is an extension feature in our ReGraph framework, which provides a bridge integrating relational and graph databases, keeping both synchronized in their native representations. In this section, we briefly explain how the ReGraph framework works and the data conversion process from a relational to a property graph database.

3.1. The ReGraph Framework

The FishBase data is stored in a relational database. Besides the existing relational database, ReGraph produces a parallel property graph database (FishGraph), to perform network analyses and to link data with Semantic Web.

Starting from a relational database, ReGraph allows mapping its data into a property graph database, generating a *mapped subgraph*. It is also possible to further create manual and automatic annotations over this data, generating an *annotation subgraph*. Both subgraphs, *mapped* and *annotation*, are connected in the graph database. ReGraph keeps relational and graph databases in their native forms and has a synchronism module that reflects in the graph database changes executed in the relational database. The graph database is focused in the analysis on the relations among data elements.

3.2. From FishBase to FishGraph using the ReGraph framework

As previously mentioned, FishGraph concerns an application of ReGraph in the FishBase information system. We have mapped the taxonomic classification of fishes from FishBase to FishGraph - see details in [Cavoto et al. 2015]. The taxonomic classification of a species includes: Kingdom, Phylum, Class, Order, Family, Genus and Species. As FishBase has only species of fishes, it does not register Kingdom and Phylum, once that all fishes belong to the same Kingdom and Phylum. This data was compared to the taxonomic classification defined in DBpedia, generating a comparison annotation type.

In order to generate a new annotation type, we have selected also the table Country, representing countries where species are found. Figure 1 shows the graph model for the taxonomic classification and country data generated in the graph database, in which we have nodes and, associated with them, their respective properties and edges connecting it to each other.

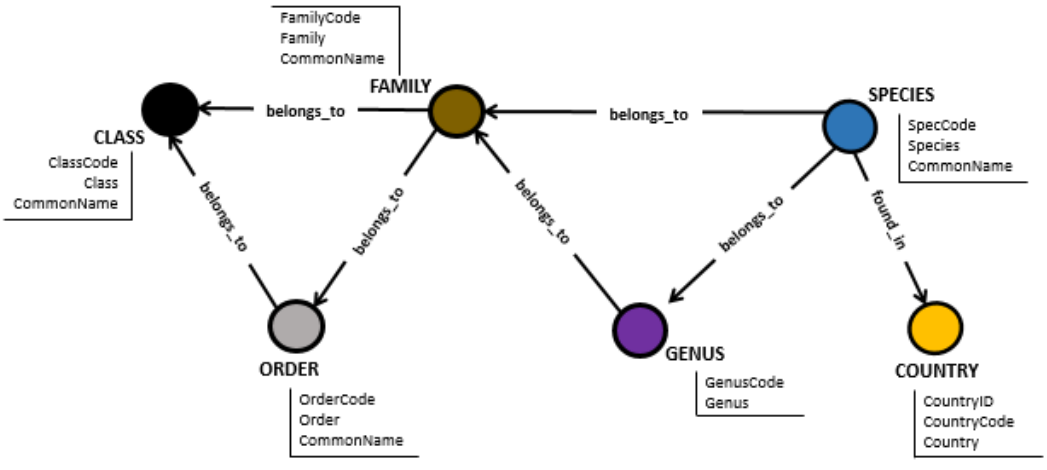

Figure 1 - Graph Model for Taxonomic Classification and Countries

We used the country information in the graph database to link them with GeoNames, a geographical knowledge base that covers all countries and contains over eight million placenames. Data retrieved from GeoNames generated new nodes and edges in the graph database, enriching it and bringing more details to the performed analyses. After the migration of the related data, we generated in the graph database 226,284 edges and 44,701 nodes, in which we have: 311 countries; 32,957 species; 10,790 genera; 572 families; 65 orders and 6 classes.

4. Automatic Annotation-Based Method

Annotations can improve the understanding and the quality of the data adding extra information. We propose a method that allows creating automatic annotations over the existent data in a property graph database. These annotations will be created through a direct connection with existing ontologies and LOD, available on the Web, e.g., GeneOntology, GeoNames and DBpedia. In this section, we detail our automatic annotation-based method and the two distinct annotation types implemented: *Comparison* and *New*. Independently of the annotation type, local data is related to Web data through a match function that compares strings to find the proper resource.

A distinctive feature of our approach is to differentiate the *annotation subgraph* (produced here) from the *mapped subgraph* (mapped from the relational database). The mapped subgraph cannot be directly changed in the graph database, since it is the product of a *one-way synchronization* originated in the relational database. Synchronization rules avoid updates in the mapped subgraph that will create inconsistencies with the *annotation subgraph*.

4.1. The Comparison Annotation Type

The main goal in the Comparison annotation type is to record comparisons of data stored in the local graph database with third party sources available on the Web. To execute this type of automatic annotation, it is necessary to define the "subject query" that will return the data from the property graph database that will be subject of the comparison.

The order of the data returned by the subject query is determinant to the correct execution of the process: (i) the first value will be the identifier of the node, helping the annotation process; (ii) the second value will be the key matched with the ontology identifiers; it will be used by the match function to retrieve data on the Web; (iii) for each of the remaining values, it is necessary specify the direct path in the ontology to reach it, linking the returned values with the specific value in the ontology; it is possible to define two paths in the ontology for each value returned by the subject query.

The result of this comparison will produce an annotation over the first node returned by the subject query. This annotation is added in the graph database as a property of the node, in which there are three possible values, annotated automatically:

- Equal: indicate elements that have the same value in the graph database and in the external ontology. This kind of annotation can improve the quality and the confidence of the data, through a double check validation.
- Not Found: represent existing elements in the graph database that was not found in the referred ontology. It can indicate: data in the graph database has spelling mistakes; the specified data does not exist in the referred ontology; data was updated in one of the sources, and was not in the other; etc.

- Divergent: represent data that have a divergence compared to the referred ontology. In can indicate: incorrect data in the graph database or in the ontology. This value is defined as a recommendation to review data. In addition, a new node is added, linked with the existing node, containing the exact data in the ontology for traceability.

4.2. The New Annotation Type

In the New annotation type, we produce new nodes, edges and/or properties, to improve the analysis and results. In this annotation type, it is necessary to specify in the "subject query" only two values: (i) the first one will be the identifier of the node, helping in the annotation process; (ii) the second one represents the key in the graph database matched with the respective identifier of a resource in the ontology; it is used by the match function to retrieve data on the Web. The second step is to define the ontology path to search.

Both data are the starting point to search in the ontology. For each information to be retrieved from the ontology and inserted in the graph database it is necessary specify: (i) ontology information: direct path in the ontology to retrieve the required information; (ii) annotation creation: how the annotation will be created in the graph database: as a node or property. The new node will be connected with the existing node by an edge that has its label also defined. In the property option, a defined property will be created over the existing node. In both cases, the value of the property will be the value found in the specified ontology.

5. Conclusions and Future Work

In this paper, we presented an automatic annotation-based method using ontologies, as an extension of our project ReGraph that connects a relational database with a property graph database, keeping both integrated, synchronized and in their native forms. It stands out for its flexibility in defining the ontologies and values that will be retrieved, compared and created, offering several possibilities to validate and enrich the graph database. Our method contrasts with the related work since it introduces a graph database perspective over the annotation-based connection between the local and global graphs. Annotations in the *annotated subgraph* stay consistent with the existing *mapped subgraph*, even after its evolution along the time.

We developed two distinct experiments to validate each proposed annotation type: Comparison and New. In the Comparison experiment, we compared almost 33,000 species of fishes from FishBase to validate their taxonomic classification with DBpedia. In the New experiment, we used the 249 countries in the graph database to retrieve their continent and information of GeonNameID and population from GeoNames.

Future work includes extending the functionality of ReGraph to allow retrieving data from other web formats and to save the link to the resource in the graph database as well as the "subject query" that generated it, helping in future repeated analysis and to track provenance.

Acknowledgments

Research partially funded by projects NAVSCALES (FAPESP 2011/52070-7), the Center for Computational Engineering and Sciences (FAPESP CEPID 2013/08293-7), CNPq-FAPESP INCT in eScience (FAPESP 2011/50761-2), INCT in Web Science (CNPq

557.128/2009-9) and individual grants from CNPq and CAPES. Thanks to FishBase.org, which provided the data used in this work.

References

Alm, R., Waltemath, D., Wolkenauer, O. and Henkel, R. (2014). Annotation-Based Feature Extraction from Sets of SBML Models. In: *Data Integration in the Life Sciences 10th International Conference, DILS 2014*, p. 81–95.

Amanqui, F. K., Serique, K. J., Lamping, F., et al. (2013). Semantic Search Architecture for Retrieving Information in Biodiversity Repositories. In: *VI Seminar on Ontology Research in Brazil, Ontobras 2013*, p. 83–93.

Berners-Lee, T. (2007). Giant global graph. Decentralized Information Group.

Bizer, C., Heath, T. and Berners-Lee, T. (2009). Linked data-the story so far. *International journal on Semantic Web and Information Systems*, v. 5, p. 1–22.

Castilho, F. M. B. M., Granada, R. L., Vieira, R., Sander, T. and Rao, P. (2011). Ontology enrichment based on the mapping of knowledge resources for data privacy management. In: *CEUR Workshop Proceedings*, v. 776, p. 85–96.

Cavoto, P., Cardoso, V., Vignes Lebbe, R. and Santanchè, A. (2015). FishGraph: A Network-Driven Data Analysis. In: *11th IEEE International Conference on e-Science 2015*, p.1-10.

Drăgan, L., Passant, A., Handschuh, S. and Groza, T. (2010). Publishing semantic personal notes as linked data. In: *CEUR Workshop Proceedings*, v. 674, p. 1–2.

FishBase Consortium (2015). FishBase. http://www.fishbase.org, July, 2015.

Mangold, Christoph. (2007). A survey and classification of semantic search approaches. *International Journal of Metadata, Semantics and Ontologies*, v. 2, n. 1, p. 1-23.

Oren, E., Delbru, R., Möller, K., Völkel, M. and Handschuh, S. (2006). Annotation and navigation in semantic wikis?. In: *CEUR Workshop Proceedings*, v. 206, p. 58–73.

Oren, E., Möller, K. H., Scerri, S., Handschuh, S. and Sintek, M. (2006). What are Semantic Annotations?. Technical Report. http://www.siegfried-handschuh.net/pub/2006/whatissemannot2006.pdf, July, 2015.

Palma, G., Vidal, M.-E., Raschid, L. and Thor, A. (2014). Exploiting Semantics from Ontologies and Shared Annotations to Partition Linked Data. In: *Data Integration in the Life Sciences 10th International Conference, DILS 2014*, p. 120–127.

Santos, V. Dos, Baiao, F. A. and Tanaka, A. (2011). An architecture to support information sources discovery through semantic search. In: *2011 IEEE International Conference on Information Reuse & Integration*, p. 276–282.

Vicknair, C., Macias, M., Zhao, Z., et al. (2010). A comparison of a graph database and a relational database. In: *ACM Southeast Regional Conference*, pp. 1–6.