# RDivF: Diversifying Keyword Search on RDF Graphs

Nikos Bikakis[1,2], Giorgos Giannopoulos[1,2⋆], John Liagouris[1,2⋆], Dimitrios Skoutas[2], Theodore Dalamagas[2], and Timos Sellis[3]

[1]National Technical University of Athens, Greece
[2]"Athena" Research Center, Greece [3]RMIT University, Australia

**Abstract.** In this paper, we outline our ongoing work on diversifying keyword search results on RDF data. Given a keyword query over an RDF graph, we define the problem of diversifying the search results and we present diversification criteria that take into consideration both the content and the structure of the results, as well as the underlying RDF/S–OWL schema.

**Keywords:** Linked Data, Semantic Web, Web of Data, Structured Data.

## 1 Introduction

As a growing number of organizations and companies (e.g., *Europeana*, *DBpedia*, *data.gov*, *GeoNames*) adopt the *Linked Data* practices and publish their data in RDF format, going beyond simple SPARQL endpoints, to provide more advanced, effective and efficient search services over RDF data, has become a major research challenge. Especially, since users prefer searching with plain keywords, instead of using structured query languages such as SPARQL, there has been an increasing interest on keyword search mechanisms over RDF data [1,2,3].

Most of the proposed works return the *most relevant* RDF results, in the form of *graphs* or *trees*. Relevance, in this case, is typically defined in terms of (a) *content similarity* between the elements comprising a result and the query terms and (b) *result compactness*, which means that smaller trees or graphs are preferred. The drawback is that this leads to result sets that are often characterized by a high degree of redundancy. Moreover, significant information is often lost, since graph paths that connect two entities and might denote a significant relation between them are omitted to satisfy the compactness requirement. Moreover, most approaches do not consider the rich *structure* and *semantics* provided by the RDF data model. For instance, an effective RDF keyword search method should treat RDF properties (edges) as first-class citizens, since properties may provide significant information about the *relations* between the entities being searched.

⋆ This research has been co-financed by the European Union (European Social Fund ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund.

As an example, consider a user searching for "*Scarlett Johansson*, *Woody Allen*" over the DBpedia dataset. An effective approach should, at least initially, consider *all* the possible ways these two entities are related. Since there exist various roles and relations between these two entities, e.g., Woody Allen may appear as either a director or an actor, this leads to a large and complex result set, containing several overlapping or similar results. The plethora of different relation combinations requires a mechanism that reduces information redundancy, allowing the system to return to the user a more concise and also more meaningful and informative result set. This can be achieved by introducing a diversification step into the retrieval and ranking process. Ideally, the system should return to the user results that cover different aspects of the existing connections between these entities, e.g., a movie where they played together, a movie directed by Woody Allen where Scarlett Johansson appears, an award they shared, etc.

Although the diversification problem has been extensively studied for document search [7,8,9], the structured nature of RDF search results requires different criteria and methods. Most of the approaches regarding keyword search on graphs [1,2,5] limit their results to trees (particularly, variations of Steiner trees); only few allow subgraphs as query answers [3,4]. Among them, [3] is the most relevant to our work; however, it does not address the diversification problem and it also does not consider the schema of the data. A different perspective is followed in [6], where a keyword query is first interpreted as a set of possible structured queries, and then the most diverse of these queries are selected and evaluated.

In this paper, we introduce a diversification framework for keyword search on RDF graphs. The main challenges arise from the fact that the structure of the results, including additional information from the underlying schema, needs to be taken into account. This is in contrast to the case of diversifying unstructured data, where the factor of content (dis)similarity is sufficient. In our framework, called RDivF (RDF + Diversity), which we are currently developing, we exploit several aspects of the RDF data model (e.g., resource content, RDF graph structure, schema information) to answer keyword queries with a set of diverse results. To the best of our knowledge, this is the first work addressing the issue of result diversification in keyword search on RDF data.

## 2 Diversifying RDF Keyword Search

Assume an *RDF graph* $G(V, E)$, where $V$ is the set of *vertices* and $E$ the set of *edges*. Optionally, $G$ may be associated with an RDF schema, defining a hierarchy among classes and properties. Let $q = \{\{t_1, t_2, \ldots, t_m\}, k, \rho\}$ be a *keyword query* comprising a set of $m$ *terms* (i.e., keywords), a parameter $k$ specifying the *maximum number of results* to be returned, and a parameter $\rho$ that is used to restrict the *maximum path length* between keyword nodes (i.e., vertices), as will be explained later. Assume also a function $\mathcal{M}\colon t \rightarrow V_t$ that maps a keyword $t$ to a set of graph nodes $V_t \subseteq V$.

**Definition 1. (Direct Keyword Path).** Assume two nodes $u, v \in V$ that match two terms $t, s$ of a query $q$, i.e., $u \in V_t$ and $v \in V_s$. Let $P$ be a path between $u$ and $v$. $P$ is called a *direct keyword path* if it does not contain any other node that matches any keyword of the query $q$.

**Definition 2. (Query Result).** Assume an RDF graph $G$ and a query $q$. A subgraph $G_q$ of $G$ is a *query result* of $q$ over $G$, if: (a) for each keyword $t$ in $q$, there exists exactly one node $v$ in $G_q$ such that $v \in V_t$ (these are called *keyword nodes*), (b) for each pair of keyword nodes $u, v$ in $G_q$, there exists a path between them with length at most $\rho$, (c) for each pair of keyword nodes $u, v$ in $G_q$, there exists at most one direct keyword path between them, and (d) each non-keyword node lies on a path connecting keyword nodes.

The above definitions leed to query results that contain pair-wise connections among all the terms in the query. That is, in our framework, we are interested in results that can be graphs and not only spanning trees, which is the typical case in previous approaches. This is based on the intuition that we want to emphasize on the completeness of relationships between query terms rather than on the criterion of minimality. Note that the aspect of minimality is still taken into consideration in our definition by means of the conditions (c) and (d) above.

Now, assume a function $r\colon (G_q, q) \rightarrow [0, 1]$ that measures the *relevance* between the query $q$ and a result $G_q$, and a function $d\colon (G_q, G'_q) \rightarrow [0, 1]$ that measures the *dissimilarity* between two query results $G_q$ and $G'_q$. Let also $f_{r,d}$ be a monotone objective function that combines these two criteria and assigns a score to a result set $\mathcal{R}$ w.r.t. the query $q$, measuring how relevant the included results are to the query and how dissimilar they are to each other. We assume that $|\mathcal{R}| > k$. Then, the goal of the diversification task is to select a subset of $k$ results so that this objective function is maximized. Formally, this can be defined as follows.

**Definition 3. (Diversified Result Set).** Assume an RDF graph $G$, a query $q$, and the functions $r$, $d$, and $f_{r,d}$ as described above. Let $\mathcal{R}$ denote the result set of $q$ over $G$. The *diversified result set* $\mathcal{R}_k$ is a subset of the results $\mathcal{R}$ with size $k$ that maximizes $f_{r,d}$, i.e., $\mathcal{R}_k = \underset{\mathcal{R}' \subseteq \mathcal{R}, |\mathcal{R}'|=k}{\operatorname{argmax}} f_{r,d}(\mathcal{R}')$.

Following this approach, in order to select a diversified result set for keyword queries over RDF graphs, one needs to determine appropriate functions for $r$, $d$, and $f_{r,d}$. Regarding the latter, [8] presents several objective function and studies their characteristics. The same functions can also be used in our case, since this aspect is independent from the nature of the underlying data. Therefore, we focus next on specifying the relevance and dissimilarity functions, $r$ and $d$, in our setting.

## 3 Diversification Criteria

The main challenge for diversifying the results of keyword queries over RDF graphs, is how to take into consideration the semantics and the structured nature of RDF when defining the relevance of the results to the query and the

dissimilarity among results. In this section, we outline a set of criteria for this purpose, which can be used for specifying the functions $r$ and $d$, as described above.

The relevance of a result to the query takes into consideration two main factors. The first factor refers to text-based matching between the nodes in the result graph and the keywords in the query. This aspect is essentially covered by the function $\mathcal{M}$ that maps query terms to graph nodes. This function can be modified to return, for each graph node, a degree of match $m \in [0, 1]$ between this node and a corresponding query keyword. In addition, a threshold $\tau$ can be specified in the query, so that only nodes with $m \geq \tau$ are returned. The second factor refers to the fact that results should be concise and coherent. One step to ensure this is the minimality criterion included in Definition 2. Furthermore, we need to consider structural and semantic homogeneity of the result, so that the results can be more meaningful to the user. This is an intra-result measure, capturing the homogeneity among the nodes, edges and paths in the result graph. For example, this would assign a higher score to a path where all the edges are labelled with the same property. Moreover, RDF schema information can be taken into account, i.e., scoring based on class or property hierarchy and least common ancestors.

The dissimilarity among results can be defined by comparing paths between corresponding pairs of keyword nodes. This takes into account both structural properties, e.g., path lengths or common subpaths, and semantic information, i.e., classes and properties corresponded to the nodes and edges along the path. The main objective here is, for each result, to obtain paths that are similar to other paths in the result, but dissimilar to paths in other results. This objective is not restrained to textual similarity only, but takes also into account the semantic similarity of classes and properties inferred by the schema.

**Acknowledgement.** This work was partially supported by *GeoKnow* project (FP7, GA no. 318159).

## References

1. Tran T., Wang H., Rudolph S., Cimiano P.: Top-k Exploration of Query Candidates for Efficient Keyword Search on Graph-Shaped (RDF) Data. In ICDE 2009.
2. Zhou Q., Wang C., Xiong M., Wang H., Yu Y.: SPARK: Adapting Keyword Query to Semantic Search. In ISWC 2007.
3. Elbassuoni S., Blanco R.: Keyword Search over RDF Graphs. In CIKM 2011.
4. Li G., Ooi B-C, Feng J., Feng J., et.al: EASE: An Effective 3-in-1 Keyword Search Method for Unstructured, Semi-structured and Structured data. In SIGMOD 2008.
5. He H., Wang H., Yang J., Yu P.: BLINKS: Ranked Keyword Searches on Graphs. In SIGMOD 2007.
6. Demidova E., Fankhauser P., Zhou X., Nejdl W.: DivQ: Diversification for Keyword Search over Structured Databases. In SIGIR 2010.
7. Drosou M., Pitoura E.: Search Result Diversification. In SIGMOD Rec., 39(1), 2010.
8. Gollapudi S., Sharma A.: An Axiomatic Approach for Result Diversification. In WWW 2009.
9. Stefanidis K., Drosou M., Pitoura E.: PerK: Personalized Keyword Search in Relational Databases through Preferences. In EDBT 2010