Techniques for Effective Search and Exploration over Knowledge Graphs

Madhulika Mohanty (2012CSZ8279)

Abstract

Knowledge Graphs (KGs) such as, YAGO, DBPedia, and Freebase, form the backbone for applications includings chatbots, personal assistants and question answering systems. They are constructed using Information Extraction (IE) techniques over various structured and unstructured sources. KGs represent the information in the form of a graph consisting of nodes to denote entities and edges to denote relationships between these entities. They are typically stored as Resource Description Framework (RDF) triples of subject, predicate and object, with subject and object being entities and the predicate denoting the relationship between them. Each of these triples is also associated with a score during the extraction process to denote the confidence of the IE technique in the accuracy of the information represented by the triple.

KGs are queried using either structured queries or relationship queries. Structured queries comprise of triple patterns having variables in the subjects, predicates or objects. Each triple pattern consists of at least one variable and queries for triples matching the constants, and variables are bound to the corresponding values in these triples. Each set of triples matching the triple patterns in a query forms a subgraph of the original KG and is returned as an answer. Relationship queries comprise of unstructured keywords. Each keyword maps to multiple keyword nodes in the KG that have the keyword as a term in their labels. An answer to a relationship query is an interconnection between the keyword nodes (one for each keyword) and denotes how the queried nodes are related to each other. The answer graphs are ranked in decreasing order of their relevance to show only the top-k (for some value of k) most relevant answers to the users. The relevance is measured by assigning a score to each answer using the constituent triple scores.

Many of these KGs are huge in size having millions of nodes and edges. Hence, querying them effectively is non-trivial. Users querying KGs can range from beginners looking to casually explore the system without a particular information need in mind, to expert users querying for specific information needs. A common problem faced by the users is getting *empty* results. This is because of their lack of knowledge of the exact labels over nodes and edges in the KG. Also, since KGs allow schemaless addition of information in the form of triples, a given information may be represented by a variety of triples. Thus, structured queries seeking exact matches face the issue of *poor recall*. That is, *all* the desired answers are not fetched as the sought information is present in multiple forms and an exact match is not found with them. On getting unsatisfactory

results, users try to relax their queries preserving original query intent. Nevertheless, coming up with useful relaxations is also challenging for these users. In this thesis, we propose *Insta-Search*, an interactive system which helps alleviate these issues by aiding the users in querying KGs. Insta-Search helps the users by providing autocompletion of partially entered query, giving instant feedback on the results that the current query would fetch and suggesting possible relaxations. We evaluate the response times for each module to demonstrate that Insta-Search helps users at interactive speeds.

Insta-Search also supports automatic relaxations for structured queries to tackle poor recall caused by exact match semantics. Since the space of relaxations is quite large and users seek only top-k answers, existing systems employ top-k operators for early termination. However, they still process *all* the relaxations, many of whose matches do not contribute triples towards the top-k answers. We propose *Spec-QP* to eliminate this inefficiency by using a speculative approach to prune the relaxations which are unlikely to contribute triples towards the top-k answers. It makes use of precomputed statistics about the scores of the triples to speculate on the requirement of relaxations. We compare Spec-QP with an existing baseline to demonstrate its efficiency and accuracy.

After the user submits her query, Insta-Search evaluates it to return top-k ranked results. Since many of these results may be of same kind, the user is frustrated while having to scroll through them before finding a new answer. We propose *KlusTree*, which post-processes these results to present a summary of them. It uses a novel language-model (LM) based representation for the answer graphs for clustering the results based on the information conveyed by them. This enhances the result diversity in the top answers. We compare KlusTree with existing techniques for clustering graphs and demonstrate the advantages of using the LM based representation for the results.

Hence, this thesis provides an integrated holistic system to facilitate smooth and effective exploration of KGs.