Pragmatics and Discourse in Knowledge Graphs

Amar Viswanathan  
Tetherless World Constellation  
Rensselaer Polytechnic Institute  
110 Eighth Street, Troy, NY USA 12180

Geeth de Mel  
Research Staff Member  
IBM TJ Watson Research Center  
Yorktown Heights, New York

James A. Hendler  
Tetherless World Constellation  
Rensselaer Polytechnic Institute  
110 Eighth Street, Troy, NY USA 12180

Abstract

Knowledge graphs (KGs) are fast becoming the cornerstone of research for storing and retrieving information effectively due to their ability to link heterogeneous data, make inferences, and discover new knowledge without additional human input. Researchers use a plethora of KGs like NELL, DBpedia, YAGO to augment their information extraction activities. However, in such diverse and expressive graphs, the access to knowledge that matches user’s needs is not always obvious—i.e., user intent does not necessarily get translated into the query interpretation which can frustrate the user. In this work, we present a discourse enabled framework on a large scale KG as a means to start an investigation into modeling user intent for query processing. We present a data aware query reformulation strategy with a faceted interface to enable discourse that helps users to specify their needs in an intuitive manner.

Introduction

In this paper we present a vision and initial findings for pragmatically aware query reformulation based on data awareness and a means to capture user intent by a faceted discourse interface. We base our discussion and initial experiments on a large scale knowledge graph (henceforth referred to as KG) created by converting information extraction outputs from the ACE\textsuperscript{1} task. In order to capture the domain, we use a dialect from the Web Ontology Language (OWL) (Hitzler et al. 2012) as the knowledge representation formalism.

Today, knowledge graphs are changing the way in which information is stored, accessed, and utilized. Given their rich schema definitions, information querying is also becoming expressive, and this is positively affecting the traditional search landscape. For example, expressive queries such as Musicians born in Berlin before 1900 or Restaurants in New York City that serve vegetarian food can easily be answered by DBpedia (Auer et al. 2007). NELL (Mitchell et al. 2015) and YAGO (Suchanek, Kasneci, and Weikum 2007) are some other mainstream examples of such knowledge graphs. With rich schema definitions and heterogeneity in information, KGs are supposed to provide us with the means to explore and look for information in a structured and intuitive manner. However, this intuitiveness is not exploited by current search interfaces to facilitate discovery, navigation, and active participation with the user due to two reasons: (a) complexity and the heterogeneity in representation makes it difficult to expose the underlying information structures to the user in a comprehensible manner; and (b) pragmatic context in which the user asks for information is not utilized, when trying to generate an appropriate answer to the user query.

We define the pragmatic context to include features such as data awareness and availability, user profiles, related extended neighborhood knowledge and so forth. While enough work is done on click stream analysis (Gross, McGovern, and Sturtevant 2005; Bucklin et al. 2002) and user profile modeling for information retrieval (Chen and Kuo 2000), these are typically considered in isolation. Our intuition is that, one needs to take a collective view on such features so that user intent is presented to the system. Motivated by this observation, in this paper we present a framework to introduce the pragmatic context such that the user intent is better presented to the information retrieval system. Our framework is inspired by a plug-in model so that we can continually improve the system by introducing the components that can address the notion of pragmatic context.

The rest of the paper is organized as follows: In the next section we provide an illustrative scenario that motivated our work. The section Overview provides an overview to our work and discuss related work briefly. The section Framework Overview presents our framework and highlights the data-aware query relation approach we have taken to partially address the pragmatic context of a query. We conclude the document in Conclusions and Future Directions by sketching future directions.

Illustrative Scenario

Let us assume that a user is organizing in a dinner party and has been querying for recipes. As a part of the main course, the user is interested in making a spicy polenta cake, and so queries an information source. The user is then presented with a list of polenta cakes and different means of cooking them—this frustrates the user as he/she needs to skim through all the information to find the needed recipe that

\textsuperscript{1}http://www.nist.gov/speech/tests/ace
matches his/her specific need. Let us now assume that the system works the following way: it keeps a record about user’s past preferences—with a model to gracefully degrade preferences as time goes by, current search interests and selections, and the kinds of media (e.g., video, audio, text with images, and so forth) the user is interested in. When the user queries for a polenta cake recipe, it suggests a spicy polenta cake recipe from the user’s favorite website that has instructions in the form of images and text. In addition the source also can present other visual sources, since user has shown an interest in video clips in the past.

However, such adaptive search systems are still not in existence because they suffer from the following reasons: (a) lack of conceptual understanding—i.e., user needs to be knowledgeable—not the system—as to how concepts interact with one another in the underlying schema which is difficult, if not impossible for humans (Dolog et al. 2009); (b) search as a one way traffic—i.e., treats each query in isolation, not as a means for a dialog with the user (Tunkelang 2015). For example, if the system knew that the user is looking for a polenta cake to go with a beef dish, it could suggest a spicy polenta cake automatically; (c) incomplete and inconsistent data—i.e., even if the schema is complete sometimes the data is not always complete. For example, though the user is interested video clips about recipes, there are no clips about a spicy polenta cake, so the system needs to adapt and be data aware; and (d) complexity in querying KGs—i.e., even if the user knew the underlying information structures, formulating a structured query to retrieve relevant information is beyond a typical user. In addition, with the increase in schema size, the complexity of query patterns increases as well.

Overview

In order to address the above challenge, in this work we take the Gricean approach of co-operative answering (Grice 1970), where our system adapts to the user by taking in his/her query and providing alternate interpretations—specific or generic—along with the original hypothesis. This gives the user a set of queries—which augments the user’s understanding of the information available. To facilitate dialog between the user and the system, as required by Gricean maxims, we have designed a faceted discourse system that actively suggests reformulations along with the results and also builds facets based on terms present in the query.

Background and State-of-the-art

The foundations of the approach lie in the application of Grice’s principles (Grice 1970). Grice proposed that talk exchanges do not normally consist of a succession of disconnected remarks, but they are, to some degree, cooperative efforts and the participants recognize in them, a common set of goals. To be able to achieve this in the current context a system should be able to talk to the user and provide contextually relevant information or additional similar relevant queries. In order to support such, in this work we rely on query relaxations and query reformulations, which are parts of cooperative answering. Generally, reformulations for RDF (Schreiber and Raimond 2014) graphs are focused on relaxations—or generalizations—aimed at pushing more relevant content to the users (Hurtado, Poulouvassilis, and Wood 2008). Such relaxations are either deductive relaxations or use RDFS semantics—i.e. type hierarchy or property hierarchy to relax triple patterns to generate more data (Poulouvassilis and Wood 2010). While such systems work on the concept and property level, they do not consider the implications of data availability and user query context.

In order to build a dialog oriented system, we rely on reformulations based on query expansion and then reduce these reformulations by means of data aware reduction scheme. A framework which allows such is presented in the next section.

Framework Overview

Figure 1 shows the architecture of the system. The input to the system is a natural language query (henceforth represented by Q). We then generate a triple representation out of this query—tools such as PowerAqua (Lopez et al. 2009), FREyA (Damijanovic, Agatonovic, and Cunningham 2010), and NLP-Reduce (Kaufmann, Bernstein, and Fischer ) can take in a natural language query and map it to a triple representation. Since the focus of our work is on Reformulation using Relaxation, we utilize the existing tools to give us an approximate match to the entities in the triple store. We convert the best match to a SPARQL (Buil-Aranda et al. 2013) interpretation. This query then passes through the Query Reformulation module, which performs reformulation via Relaxations and Data Awareness by means of Algorithm 1.

The reformulated queries are then ranked and the top – k reformulations are used to generate data aware queries. These reformulations are then used to build facets for the user interface. This allows the user to formulate additional queries and compare results. This entire user session is captured to model the current user context. Figure 2 shows a Facet for a single concept Nation out of the query Which nations are involved in attacks. The facet shows the relevant
of facets and one can capture the dialog that the user is having with the interface. While understanding the user’s precise intent is an open problem, our intuition is that this approach would make the search more informed, constrained and meaningful.

**Data Awareness**

Typically, if one uses concept similarity to relax queries, the expanded query set may result in redundant queries as data related to them may not reside in the KG. Furthermore, when querying for conceptual data, triples that deal with datatype properties do not lead to new knowledge or reformulations (Hurtado, Poulouvasilis, and Wood 2008). Thus, we reduce the number of triples by a similarity match while accounting for data availability in the KG—data awareness in our approach removes query patterns which do not contribute towards hierarchical suggestions or zero results. We have implemented Algorithm 1 to perform this task: it takes in a set of triples and checks whether they are present in the KG or not. If the triple patterns are present, we add it to the reformulations or else these are discarded.

**Algorithm 1: Data Aware Query Reduction**

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DATAAWARE on ( t(c_i \text{ rdf:type C}) ) and ( \mathcal{KG} );</td>
</tr>
<tr>
<td>2</td>
<td>Input : triples of type ( { c_i \text{ rdf:type C} } ), Schema ( \mathcal{O} )</td>
</tr>
<tr>
<td>3</td>
<td>Output: list of triples ( t_j \in { c_j \text{ rdf:type C} } ) each ( t_j ) is semantically similar, ( j \leq i )</td>
</tr>
<tr>
<td>4</td>
<td>( T = { t_1, t_2, \ldots, t_i } ), dataAwareT ( \leftarrow \emptyset ),</td>
</tr>
<tr>
<td>5</td>
<td>while ( \exists t_i \in T ) do</td>
</tr>
<tr>
<td>6</td>
<td>Remove ( t_i ) if ( t_i ) contains datatype property and continue ;</td>
</tr>
<tr>
<td>7</td>
<td>Count ( c_i \in t_i { c_i \text{ rdf:type C} } ) on ( KG ) ;</td>
</tr>
<tr>
<td>8</td>
<td>if Count( c_i \geq 0 ) ;</td>
</tr>
<tr>
<td>9</td>
<td>Add ( t_i ) to dataAwareT;</td>
</tr>
<tr>
<td>10</td>
<td>end</td>
</tr>
<tr>
<td>11</td>
<td>return dataAwareT ;</td>
</tr>
</tbody>
</table>

**Example Walkthrough**

While our system can handle three kinds of queries - star, composite and linked, we summarize the results of this reformulation with an example composite query \( q_1 \in \{ \text{Find all nations who are involved in attacks} \} \), which looks like:

\[
q_1 \{ \text{entity event role} \} :=
\text{entity rdf:type individual}
\text{event rdf:type attack}
\]

The Table 1 shows the results of the reformulation for query \( q_1 \) using techniques developed by our algorithm. This example is queried on a sample KG that is extracted and built from 75,000 documents, which are in the

---

1. Impreciseness of the query terms is also considered as the contextual here
We have presented an extensible framework that hopes to introduce the pragmatic context of user queries as means to introduce user intent into information querying. With our initial work, we have constrained ourselves to look into data-awareness and neighborhood knowledge as means to reformulate queries. The approach has resulted in promising results which is briefly discussed in the paper.

In the immediate future we would like to investigate how query reformulation fits into the larger scheme of relevance in information research and to further discuss the pragmatic context of the user queries. We would then plan to use the generated reformulations to learn about user intent by modeling user profiles through dialog capture.

### Conclusions and Future Directions

We have presented an extensible framework that hopes to introduce the pragmatic context of user queries as means to introduce user intent into information querying. With our initial work, we have constrained ourselves to look into data-awareness and neighborhood knowledge as means to reformulate queries. The approach has resulted in promising results which is briefly discussed in the paper.

In the immediate future we would like to investigate how query reformulation fits into the larger scheme of relevance in information research and to further discuss the pragmatic context of the user queries. We would then plan to use the generated reformulations to learn about user intent by modeling user profiles through dialog capture.

### References


---

Table 1: Reformulation Steps for $q_1$ applying Algorithm 1

<table>
<thead>
<tr>
<th>TECHNIQUE</th>
<th>REFORMULATIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query Relaxation</td>
<td>74,620</td>
</tr>
<tr>
<td>Entity Aware Restriction</td>
<td>11,480</td>
</tr>
<tr>
<td>Event Restriction</td>
<td>840</td>
</tr>
<tr>
<td>Domain and Range Restriction</td>
<td>48</td>
</tr>
<tr>
<td>Similarity Restriction</td>
<td>24</td>
</tr>
<tr>
<td>Data Aware Restriction</td>
<td>24</td>
</tr>
</tbody>
</table>

ACE’05 3 schema. In $q_1$, the given query is matched against \{ENTITY, ROLE, EVENT\} from the schema. The schema built from the documents has a total of 132 classes and 233 Logical axioms, along with 37 object properties and 10 data properties. In addition to these results we have built a discourse enabled faceted user interface that eases the interaction with the KG.

Our initial experiments included both synthetic–LUBM4 and non-synthetic–ACE datasets. A total of 7 benchmark queries for the LUBM dataset based on (Huang, Liu, and Zhou 2012) were used to test the efficiency of the data awareness algorithm. In addition we created a set of 7 benchmark composite queries for the generated ACE KG. Using this we evaluated it against the relaxation algorithms of (Hurtado, Poulovassilis, and Wood 2008) and (Huang, Liu, and Zhou 2012). Initial results show that addition of data awareness results in at least 60 percent reduction in the number of reformulations generated.

---

3 http://www.itl.nist.gov/iad/mig/tests/ace/ace05/doc/  
4 http://swat.cse.lehigh.edu/projects/lubm/