Ontology-Based Understanding of Natural Language Queries Using Nested Conceptual Graphs

Tru H. Cao and Anh H. Mai

Faculty of Computer Science and Engineering
Ho Chi Minh City University of Technology, Vietnam
tru@cse.hcmut.edu.vn

Abstract. In a question answering system, users always prefer entering queries in natural language and not being constrained by a rigorous grammar. This paper proposes a syntax-free method for natural language query understanding that is robust to ill-formed questions. Nested conceptual graphs are defined as a formal target language to represent not only simple queries, but also connective, superlative, and counting queries. The method exploits knowledge of an ontology to recognize entities and determine their relations in a query. With smooth mapping to and from natural language, conceptual graphs simplify conversion rules from natural language queries and can be easily converted to other formal query languages. Experimental results of the method on the QA track datasets of TREC 2002 and TREC 2007 are presented and discussed.

1 Introduction

A natural language interface is always desirable for a question answering system ([12]). Using a controlled language for the interface could ease the problem but still tightens users to a restricted grammar ([10], [11]). Although performance of machine natural language understanding for the general case appears to be saturated after many years of research, limiting the domain of discourse to only questions and querying phrases can make a difference. There are various approaches to conversion of natural language queries into more formal representations regarding the two following issues.

First, it is about whether rigorous syntactic parsing is applied to a query expression before it is mapped to a target language sentence. The disadvantages of the parsing approach are time consuming and requiring grammatically correct inputs, which is thus not robust to ill-formed queries. It is also not practical to require a user to always input a question without grammatical errors. Moreover, it may still face to the problem of syntactic ambiguity, i.e., one sentence having more than one applicable syntax tree.

Second, it is about whether a knowledge base (KB) is employed in the mapping. For example, with the query “What county is Modesto, California in?”, given no knowledge base, Modesto and California can be tagged only as proper
nouns and thus the implicit relation expressed by the comma between them cannot be interpreted. In contrast, with a knowledge base, they can be recognized as named entities (NE) of the types city and province, respectively, whence the relation can be mapped to one being a sub-region of the other.

For instance, [9] implemented an ontology-based search system whose queries were lists of classes and instances and translated into expressions of SeRQL. They were better than lists of normal keywords, but not as natural as human expressions. Meanwhile, accepting natural language queries, [3] followed the rigorous parsing approach using lambda calculus as an intermediate formal language for the mapping. However, the focus of that work was on efficient porting interfaces between different domains rather on the mapping itself.

The approach in [7] could be considered as closer to the syntax-free one. It used pattern matching of a natural language query to subject-property-object triples in a knowledge base, before converting the query to one of SPARQL. For the example query therein “What is a restaurant in San Francisco that serves good French food?”, it first searched for those triples whose subjects, properties, and objects could match with “restaurant”, “in”, and “San Francisco”. That method thus could not produce a mapping if the KB did not contain such a triple for the named entity San Francisco, although it existed in the KB. We argue that the understanding step should not be mixed up with the answering step. That is, a query can have a mapping to a target language although there is no matched answer to it in a knowledge base of discourse.

Recently, [15] also followed the syntax-free approach to convert natural language queries into SeRQL expressions. It used the named entity recognition engine of GATE ([4]) and the PROTON ontology of KIM ([8]), but supplemented it with more entity types and relation types. The method was however just tested on the authors manually collected 36 questions.

Meanwhile [2] developed a method that did not rely on a strict grammar of querying sentences but did use an ontology and knowledge base for understanding natural language queries. Knowledge was provided not only for answering queries but also for their conceptual understanding, before they could be mapped to a target language. Conceptual graphs (CG [13]) were proposed as an intermediate language to convert queries to. The method was tested on the QA track datasets of TREC 2002 and TREC 2007 with hundreds of diverse questions.

Since the root of the difficulty of machine natural language understanding is the big gap between natural language and a machine executable one, using an intermediate language like SeRQL or CGs is a way to ease the problem. We choose CGs because they could be mapped smoothly to and from natural language, and used as an interlingua for conversion to and from other formal languages ([14]). Tim Berners-Lee, the inventor of the World Wide Web, concluded in [11] that CGs could be easily integrated with the Semantic Web. It was also shown in [16] that there was a close mapping between CGs and the RDF language.

There was research on automatic generation of CGs from natural language texts in a specific domain, e.g. the rule-based method in [6] and the machine