Data & Knowledge Engineering 69 (2010) 261–278

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Computing intensional answers to questions – An inductive logic programming approach

Philipp Cimiano [a,*], Sebastian Rudolph [b], Helena Hartfiel [c]

[a] Semantic Computing Group, CITEC, Universität Bielefeld, P.O. Box 10 01 31, D-33501 Bielefeld, Germany
[b] Institute AIFB, Karlsruhe Institute of Technology (KIT), D-76131 Karlsruhe, Germany
[c] Lindenbaum GmbH, Erbprinzenstr. 4-12, D-76133 Karlsruhe, Germany

A R T I C L E I N F O

Article history:
Available online 13 October 2009

Keywords:
Natural language interfaces
Intensional answers
Applications of inductive logic programming

A B S T R A C T

Research on natural language interfaces has mainly concentrated on question interpretation as well as answer computation, but not focused as much on answer presentation. In most natural language interfaces, answers are in fact provided extensionally as a list of all those instances satisfying the query description. In this paper, we aim to go beyond such a mere listing of facts and move towards producing additional descriptions of the query results referred to as intensional answers. We define an intensional answer (IA) as a logical description of the actual set of answer items to a given query in terms of properties that are shared by exactly these answer items. We argue that IAs can enhance a user's understanding of the answer itself but also of the underlying knowledge base. In particular, we present an approach for computing an intensional answer given an extensional answer (i.e. a set of entities) returned as a result of a question. In our approach, an intensional answer is represented by a clause and computed based on inductive logic programming (ILP) techniques, in particular bottom-up clause generalization. The approach is evaluated in terms of usefulness and time performance, and we discuss its potential for helping to detect flaws in the knowledge base as well as to interactively enrich it with new knowledge. While the approach is used in the context of a natural language question answering system in our settings, it clearly has applications beyond, e.g. in the context of research on generating referring expressions.

1. Introduction

Question answering systems for unstructured [1] or structured information [2] – typically referred to as natural language interfaces (NLIs) in the latter case – have been a focus of research for a few decades now (see [3,4] for somewhat older overviews of NLI research). They are a crucial component towards providing users with intuitive access to the vast amount of information available world wide in the form of resources as heterogeneous as web sites, classical and Semantic Web databases, RSS feeds, blogs, wikis, etc.

However, most of the prevalent work has concentrated on providing *extensional* answers to questions. In essence, what we mean by an extensional answer here is a list of instances that satisfy the query.[1] For example, a question like: *Which states have a capital?*, when asked to a knowledge base about Germany, would deliver an (extensional) answer consisting of the 16

* Corresponding author.
E-mail address: Cimiano@TechFak.Uni-Bielefeld.DE (P. Cimiano).

[1] Actually, the bindings of the query variables in a query (seen as a logical formula) to instances.

0169-023X/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.datak.2009.10.008

federal states (‘‘Bundesländer”): *Baden-Württemberg, Bayern, Rheinland-Pfalz, Saarland, Hessen, Nordrhein-Westfalen, Niedersachsen, Bremen, Hamburg, Sachsen, Brandenburg, Sachsen-Anhalt, Mecklenburg-Vorpommern, Schleswig-Holstein, Berlin*, and *Thüringen*. While this is definitely a correct answer, it is certainly not maximally informative. An arguably more informative answer would – beyond a mere listing of the matching instances – also *describe* the answers in terms of relationships which can help a user to better understand the answer space. To a question ‘‘*Which states have a capital?*” a system could, besides delivering the full extension, also return an answer such as ‘‘*All states (have a capital)*”, thus informing a user about a distinguishing characteristic that all the elements in the answer set share, i.e. the one of being a state.

The predominant view in question answering research is that answers are essentially lists of answer items. In fact, there has not been much research on aspects of answer presentation, in particular on:

- Integrating additional external data for presentation and visualization purposes (e.g. mashups).
- Structuring the answer space, i.e. by grouping/clustering answers.
- Constructing explanations or proofs for answers.
- Providing partial answers, and
- Delivering non-extensional (e.g. intensional) answers.

While all above mentioned aspects are worthwhile to be investigated further, this paper will be focused on the last point. We will in fact refer to such descriptions which go beyond the mere enumeration of the answer items as *intensional answers* (IAs). The intension in this sense is thus a description of the answer items in terms of a *distinguishing* set of properties *common* to all of them. In other words, the description is satisfied by all answer items but by no item that is not contained in the answer set. In the above case, the common property of the answers is that they represent exactly the set of all federal states in the knowledge base.

It could be certainly argued that providing intensional descriptions ‘‘over-answers” the question.

As a counterargument to this objection it is important to mention that, first of all, the extension might simply be too large for a user to read it entirely and to make sense of it. In this case, a compact representation of the answer in terms of an intensional description might be useful.

Second, an intensional answer can indeed provide new but relevant knowledge to the user. For example, let us consider the question ‘‘*Which states does the Spree flow through?*”, with answers: *Berlin* and *Brandenburg*. The intensional answer to this question, as generated by our approach, is: ‘*(By the way) All the states which the Havel flows through*’. From this answer, the user learns that the Havel and the Spree flow through exactly the same states and that the Havel flows through Berlin and Brandenburg in particular.

Third and most importantly, as we will see in the remainder of this paper, IAs can help to detect flaws in the knowledge base as well as suggest refinements. In the case of the above mentioned question ‘‘*Which states have a capital?*”, the user could be asked whether to add the axiom that ‘‘*All states have a capital*” into the knowledge base, thus refining it in a query-driven way. The confirmation by a user would be necessary as the axiom has been derived inductively and it is not clear if it holds universally or only in the given state of the knowledge base.

We present an approach for computing intensions of queries given their extensions and a particular knowledge base. Our research has been performed in the context of the natural language interface ORAKEL (see [2]) and the aim has been to integrate the component for providing intensional answers into the system. Notwithstanding, the approach presented here could be used in any setting where intensional answers are appropriate (and as we will see in the related work section it has strong connections to research on generating referring expressions). We describe an approach based on inductive logic programming (ILP) and in particular based on bottom-up clause generalization which computes a clause covering the extension of the query and thus representing an intensional answer in the sense that it describes it in terms of features which only the elements in the extension have in common. In particular, the system iteratively calculates least general generalizations (LGGs) for the answers by adding one answer item at a time and generalizing the clause computed so far.

The paper is structured as follows: in Section 2, we describe the ORAKEL system in more detail in order to provide the reader with some background of our research. We discuss the approach for generating intensional answers in more detail in Section 3. In Section 4, we describe the empirical evaluation of our approach, which has been carried out based on the dataset previously used in the evaluation of the ORAKEL system (see [2]). We analyze in detail to what extent the intensional answers produced by the system are ‘‘useful” and also discuss how the intensional answers can help in detecting errors in the knowledge base. We discuss related work in Section 5 and conclude in Section 6.

2. Background: the ORAKEL system

The ORAKEL system is a portable and compositional natural language interface to knowledge bases. It is portable as it can be easily adapted to new ontologies and knowledge bases by non-experts. This has been corroborated before by experimental data (see [2]). It is compositional in the sense that it interprets the question by constructing a logical formula representing the question, accomplishing this in a compositional way, i.e. on the basis of the ‘‘meaning” of the constituents of the question and the way they are (syntactically) combined (as made explicit by a syntactic analysis of the sentence).