STSM Report

Andrea Calì

Dept of Computer Science
Birkbeck, University of London, UK
andrea@dcs.bbk.ac.uk

Abstract. This is a brief report of the research activities carried out under the COST-KEYSTONE grant during the *Short-Term Scientific Mission* to the Università Roma Tre. In particular, I worked with Prof. Riccardo Torlone. The topic of the visit was keyword searches on Deep Web (a.k.a. Hidden Web) data sources. The Deep Web is constituted by data that are accessible through Web pages, but are not indexable by search engines, being returned in dynamic pages. We notion of keyword search in the context of web data sources accessible through HTML forms, and propose a preliminary framework for it.

Background

The idea of querying relational databases using keywords emerged a decade ago as a way to provide an high-level access to data and free the user from the knowledge of query languages and data organization. From then, a lot of work has been done in this field on the practical side. The problem has been recently formalized in [1], where the proposed framework is independent of the organisation of the data in relational tables; such framework is based on the weak instance model [2].

The common approach before [1] is as follows: the database is viewed as a graph G in which the nodes represent tuples and the edges represent *foreign key* (and only foreign key) references between them, a query is a set of strings Q (the keywords), and the result is a subgraph G' of G whose nodes contain the keywords in Q. However, different definitions have been proposed for the structure of the results. Usually, it is assumed that G' is a tree, even if alternative semantics have been proposed. In general, two conditions are imposed, in different ways. The former simply requires that all the keywords should appear in the result, while the latter requires that the result should have a limited size. In this framework, query answering usually relies on rather complex, graph-based techniques.

The term *Deep Web* (sometimes also called *Hidden Web*) refers to the data content that is created dynamically as the result of a specific search on the Web. In this respect, such content resides outside Web pages, and is only accessible through interaction with the Web site – typically via HTML forms. It is believed that the size of the Deep Web is several orders of magnitude larger than that of the so-called *Surface Web*, i.e., the Web that is accessible and indexable by search engines.

Usually, data sources accessible through Web forms are modelled by relations that require certain fields to be selected – i.e., some fields in the form need to be filled in. These requirements are commonly referred to as *access limitations* in that access to data can only take place according to given patterns.

In such contexts, computing the answer to a user query cannot be done as in a traditional database; instead, a query plan is needed that provides the best answer possible

while complying with the access limitations. During the STSM in question, we tried to address the problem of keyword search on Deep Web sources.

Problem definition

Preliminaries

We now formalize the problem we deal with. We model data sources as relations of a database. A *database schema* $\mathcal{R}$, or simply *schema*, is a set of *relation schemata* $\{r_1, \ldots, r_n\}$; a relation schema r is a set of attributes $\{A_1, \ldots, A_m\}$, each associated with an *abstract domain* Δ_{A_i}, $1 \leq i \leq m$. The abstract domain denotes the set of values that can appear in the corresponding attribute; rather than denoting the concrete value type (such as string or integer) that can appear, the abstract domain represents the type at a higher level of abstraction (for instance, *car* or *country*). The *arity* of a relation schema r is the number of its arguments; r/m denotes that r has arity m.

A relation instance for a relation schema $r = \{A_1, \ldots, A_m\}$ is a set of tuples of the form $\langle c_1, \ldots, c_m \rangle$, where $c_i \in \Delta_{A_i}$, $1 \leq i \leq m$. We assume all values of Δ_{A_i} belong to an infinite domain Δ. A database instance D over a database schema $\mathcal{R} = \{r_1, \ldots, r_n\}$ is a set of relation instances $\{r_1^D, \ldots, r_n^D\}$, where r_i^D denotes the relation instance of r_i in D.

Keyword queries

A *keyword query* is a set of values (constants) in Δ. Notice that the constants do not need to belong to any specific abstract domain; this reflects the assumption that a user posing a keyword query (where the keywords are obviously the constants in the set) does not know the abstract domain of the attributes, or the attributes at all.

Let $\mathcal{A}$ be the set of all attributes of relations in $\mathcal{R}$. We now provide the semantics of a database instance for a schema $\mathcal{R} = \{r_1, \ldots, r_n\}$ along the lines of [1]. We rely on the notion of *weak instance* $D_\mathcal{R}$ of a relation $\bar{r}/|\mathcal{A}|$ on all attributes in $\mathcal{R}$. A weak instance $\bar{r}^{D_\mathcal{R}}$ for $\mathcal{R}$ has values in $\Delta \cup \Delta_N$, with $\Delta \cap \Delta_N = \emptyset$, where Δ_N is an infinite set of values that denote unknown constants of Δ. We define $\bar{r}^{D_\mathcal{R}}$ as follows. Let $\mathbf{A}_i$ be the set of attributes of r_i, for $1 \leq i \leq n$, and $\pi_\mathbf{A}(r^D)$ the projection on the set of attributes $\mathbf{A}$ of the relation instance r^D. We now define a weak instance $\bar{r}^{D_\mathcal{R}}$ from a given instance D for $\mathcal{R}$ as any instance such that

$$\pi_{\mathbf{A}_i}(D_\mathcal{R}) \supseteq r_i^D$$

for all i such that $1 \leq i \leq n$.

Access limitations.

An access pattern for a relation schema $\mathcal{R} = \{A_1, \ldots, A_m\}$ is a mapping sending each attribute A_i into an access mode, which can be either input or output.

Now we define the notion of *reachable instance*, i.e., the portion of the semantics that can be extracted by observing the access patterns.

Definition 1 (Reachable instance). *For a database instance $\mathcal{I}$ over a database schema $\mathcal{D} = \{\mathcal{R}_1, \ldots, \mathcal{R}_n\}$ and a set of access patterns Π, the reachable instance $reach(\mathcal{I}, \Pi)$ is the (smallest? largest? ce ne sta uno solo?) subset of the semantics $\mathcal{I}^*$ of $\mathcal{I}$ such that $\pi_{\mathcal{R}_i}(reach(\mathcal{I}, \Pi)) = reach(r_i^\mathcal{I}, \Pi)$ for $1 \leq i \leq n$.*

Definition 2 (Answer). *An answer to a keyword query q against a database instance $\mathcal{I}$ over a schema $\mathcal{D}$ with access patterns Π is a tuple $t \in reach(\mathcal{I}, \Pi)$ such that*

$$\forall c \in q \; \exists A \in \mathcal{D}^* \textbf{ s.t. } c \in \pi_A reach(\mathcal{I}, \Pi).$$

Future work and challenges

We here list the main challenges that we have identified during the STSM in Rome. Advancements have already been made in the topic, in terms of technical results and algorithms for keyword query processing, but the new material is not yet mature for presentation in this document.

Challenge 1 The definition of answers to keyword queries needs to be extended in order to take into account tuples that are *linked* through common values (not necessarily by foreign key constraints). This is useful independently of the issue of access limitations, and notably different attributes will have a different degree of importance in such links. For instance, if we are querying a database in search of a criminal, it would be useful to see people who have worked in the same firm as the criminal; instead, people who lived in the same country as the criminal would be too weakly linked (semantically) to be interesting.

Challenge 2 We aim at studying the interaction between access limitations and the *relevant parts* of the reachable instance; indeed, not all the tuples in the reachable instance are interesting to a certain query; it is likely that the query serves as a

Challenge 3 Finally, the complexity of answering keyword queries in the context here presented will be studied. This will provide a theoretical basis for the study on scalability (hopefully with experiments) that we shall carry out.

The STSM has provided an occasion of close interaction and research work with Prof. Torlone, and the results (those obtained so far as well as those to come) will be submitted to an international conference (venue to be decided). Furthermore, we are exploring the possibility of a joint proposal for a research grant; this could be under the H2020 scheme (if suitable partners are found) or – more probably, at least in first instance – to a British grant scheme such as those of the Leverhulme Trust.

References

1. Riccardo Torlone: Towards a new Foundation for Keyword Search in Relational Databases. *Proc. of AMW 2014*.
2. Alberto O. Mendelzon: Database States and Their Tableaux. *ACM Trans. Database Syst.* 9(2): 264-282 (1984).