Provenance-enhanced keyword-based search over structured data
STSM report

Paolo Missier and Francesco Guerra

October 23, 2014

This STSM explored the combination of formal and practical notions of data provenance, with known approaches to the problem of keyword-based search over relational databases (or, more generally, semi-structured data models such as RDF). At this early stage, our aim has been to investigate opportunities for exploiting the provenance of the results from previous searches, to improve the quality and efficiency of future keyword-based search.

The host (Ing. Guerra, Unimore) contributed his knowledge, manifested through past top-level publications [4, 1, 2], of keyword-based search, while Dr. Missier (the visiting guest) contributed knowledge on the theory and practice of data provenance. By combining our strengths, we have been able to formulate a plan for joint research collaboration, described in the rest of this report.

In current practice, a keyword search over relational data entails mapping a set K of keywords onto domain elements V_D taken from the state D of the database (either data content or data dictionary). We express this using the mapping function

$$kl : \mathcal{K} \rightarrow [V_D]$$

(for "keyword lookup") where $[V_D]$ denotes the set of all lists over elements of V_D, and $V_i = kl(k_i)$, $k_i \in K$, is a list of terms in the active domain of the database in state D. By selecting one term from each V_i, one generates $N = |V_1| \times \cdots \times |V_{|K|}|$ possible lists of domain terms, which we denote V_j^K, $1 \leq j \leq N$. The lists V_j^K are called *configurations*.

Each configuration is used in combination with the database schema constraints (mainly PK/FK), to generate a set Q_j^K of SQL queries, with the property that each $q \in Q_j^K$ joins over each of the tables that contain one of the elements in V_j^K, which we refer to as *target tables*. The term *candidate network* (CN) is used in this context to denote paths in the database schema that connect tables through the constraints, and thus translate into PK/FK joins in the queries. For each V_j^K, multiple paths and thus multiple CNs may exist that satisfy the integrity constraints and involve the target tables. Therefore, a set Q_j^K of queries can be associated to each configuration. All $q \in q \in Q_j^K$ are independently executed, and each result is delivered to the user.

To summarise, this approach results in a combinatorial expansion of the keywords, at two levels. Firstly, K maps to N configurations V_j^K, and secondly, each configuration maps to a set Q_j^K of queries. Research efforts in this area have focused on (i) reducing the combinatorial space and therefore the number of different results returned, and (ii) improving the precision and recall of the search process. Specifically, recent research [2, 4] have explored ways of learning the "intended" interpretation of the keywords according to the user who submitted the search, as this would address both problems.

Our own investigation starts from the observation that, although the keywords map to elements of the data domain, all reasoning aimed at learning user intent occurs at the schema level. Data provenance has the potential to overcome this limitation, allowing for fine-grained analysis of the role played by each tuple in D when specific sets of a keywords are used for searching. The provenance, and more specifically the *Why-provenance* of a tuple that appears in the result of a query[1] contains references to all the tuples that have participated in the computation of the result. For instance, consider tables $R(A, B, C)$, $S(D, E, F)$, $t_1 = (x, b, c) \in R$, $t_2 = (x, e, f) \in S$. The results of a join $R \bowtie_{R.A=S.D} S$ contains $t = (x, b, c, x, e, f)$. In this very simple example, the provenance of t contains references to t_1, t_2. The theory of provenance for relational data and relational algebra [3] describes an encoding of the provenance of a tuple such as t as a polynomial. For simplicity, we represent provenance only as a set $P(t) = \{t_1, t_2\}$. Following [3], we call $P(t)$ the *lineage* of t. Query engines can be instrumented to generate pairs $\langle t, P(t)\rangle$ for each t in the result $Q(D)$ of a query Q executed on state D.

Our key insight, in the context of keyword-based search, is that a search involving keyword set K generates, through the execution of multiple queries $Q \in Q_j^K$ as explained above, a set $\mathcal{P} = \{P(t)|t \in Q(D)\}$. $\mathcal{P}$ is a provenance database that grows monotonically as new searches are submitted to the system.

We have planned to continue this research collaboration, centred on experimenting with analysis of this database with the goal of improving on the efficiency as well as the accuracy of the searches. Within the next few months, we will be targeting a major publication on this topic in an international journal in the data management area.

References

[1] Sonia Bergamaschi, Elton Domnori, Francesco Guerra, Raquel Trillo Lado, and Yannis Velegrakis. Keyword search over relational databases: a metadata approach. In Timos K Sellis, Renée J Miller, Anastasios Kementsietsidis, and Yannis Velegrakis, editors, *Procs. SIGMOD 2011, Athens, Greece, June 12-16, 2011*, pages 565–576. ACM, 2011.

[1] We focus on SQL queries, but the formal theory has been extended to XML, RDF/SPARQL for instance.

[2] Sonia Bergamaschi, Francesco Guerra, Matteo Interlandi, Raquel Trillo Lado, and Yannis Velegrakis. QUEST: A Keyword Search System for Relational Data based on Semantic and Machine Learning Techniques. *PVLDB*, 6(12):1222–1225, 2013.

[3] V. Green, Todd J., Karvounarakis, G., Tannen. Provenance Semirings. In *PODS*, pages 31–40, 2007.

[4] Silvia Rota, Sonia Bergamaschi, and Francesco Guerra. The list Viterbi training algorithm and its application to keyword search over databases. In Craig Macdonald, Iadh Ounis, and Ian Ruthven, editors, *Procs. CIKM 2011, Glasgow, United Kingdom, October 24-28, 2011*, pages 1601–1606. ACM, 2011.