B. Mitschang et al. (Hrsg.): Datenbanksysteme für Business, Technologie und Web (BTW 2017),
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2017 617

Exploring Databases via Reverse Engineering Ranking Queries with PALEO

(Invited Demonstration)

Kiril Panev[1] Sebastian Michel[1] Evica Milchevski[1] Koninika Pal[1]

A novel approach to explore databases using ranked lists is demonstrated. Working with ranked lists, capturing the relative performance of entities, is a very intuitive and widely applicable concept. Users can post lists of entities for which explanatory SQL queries and full result lists are returned. By refining the input, the results, or the queries, users can interactively explore the database content. The demonstrated system was previously presented at VLDB 2016 and is centered around our PALEO framework for reverse engineering OLAP-style database queries. How is this useful for exploring data?

Consider a user Alice who needs to make up her mind which smartphone to buy next. Alice is favoring model X, model Y, and model Z, in this order. She is interested in finding explanatory queries and in fact populated rankings that resemble this ranking. PALEO tries to determine such queries who could generate this list and looking at their structure Alice learns about the categorical constraints and ranking criteria used. Given the computed rankings, Alice can further learn about other smartphones that perform perhaps even better, depending on how much PALEO is allowed to deviate from the original input ranking. Assume PALEO returned a ranking of {X, W, Y, Z} with constraints *'storage=16GB'* and *'brand=Samsung'*, ranked by *'battery lifetime'*. What can she learn from that and how can she proceed? She can remove the constraint on the make to get additional offers, she also learned that the model W appears feasible, too. Further, she changes the ranking criteria as *'battery lifetime'* is not the most decisive criterion for her anyways, can distort the ranking slightly to see how generating queries are going to differ, etc.

Developing a system that allows working with rankings in such an exploratory fashion brings up several challenges that need to be addressed. First, subsecond response times are required to allow interactive data exploration. PALEO achieves this by precomputed statistics, decision trees, in-memory processing over a sufficient subset of the data, and low false positive rate in the candidate-query evaluation. Second, the system has to allow approximate answers to the user input, as it is not reasonable to assume that the identical ranking can be retrieved from the database content. We address this, by deeply incorporating distance-measure–based pruning into the candidate query generation. Third, a way to bring candidate queries in an order that reflects a user-perceived notion of interestingness is required. To achieve this, PALEO employs novel insights from mining Web tables corpora to derive a classifier that is able to tell whether or not a non-numerical attribute is semantically meaningful to act as a constraint to the `WHERE` clause of a query.

[1] TU Kaiserslautern, Germany, {panev,michel,milchevski,pal}@cs.uni-kl.de