Scale Independence: Using Small Data to Answer Queries on Big Data

Floris Geerts

Department of Mathematics & Computer Science, University of Antwerp, Belgium
floris.geerts@uantwerpen.be

Abstract

Large datasets introduce challenges to the scalability of query answering. Given a query Q and a dataset D, it is often prohibitively costly to compute the query answers $Q(D)$ when D is big. To this end, one may want to use heuristics, "quick and dirty" algorithms which return approximate answers. However, in many applications it is a must to find exact query answers. So, how can we efficiently compute $Q(D)$ when D is big or when we only have limited resources?

One idea is to find a small subset D_Q of D such that $Q(D_Q) = Q(D)$ where the size of D_Q is independent of the size of the underlying dataset D. Intuitively, when such a D_Q can be found for a query Q, the query is said to be *scale independent* [1, 2, 9]. Indeed, for answering such queries the size of the underlying database does not matter, i.e., query processing is independent of the scale of the database.

In this talk, I will survey various formalisms that enable large classes of queries to be scale independent. These formalisms primarily rely on the availability of access constraints, a combination of indexes and cardinality constraints, on the data [8, 9]. We will take a closer look at how, in the presence of such constraints, queries can often be compiled into efficient query plans that access a bounded amount data [6, 8], and how these techniques relate to query processing in the presence of access patterns [3, 4, 7]. Finally, we illustrate that scale independent queries are quite common in practice and that they indeed can be efficiently answered on big datasets when access constraints are present [5, 6].

1998 ACM Subject Classification H.2.4 [Database Management] Systems – Query Processing, H.2.3 [Database Management] Languages – Query Languages, H.2.2 [Database Management] Physical Design – Access methods

Keywords and phrases Scale independence, Access constraints, Query processing

Digital Object Identifier 10.4230/LIPIcs.ICDT.2016.2

Category Invited Talk

References

1 Michael Armbrust, Kristal Curtis, Tim Kraska, Armando Fox, Michael J. Franklin, and David A. Patterson. PIQL: Success-tolerant query processing in the cloud. *PVLDB*, 5(3):181–192, 2011.

2 Michael Armbrust, Eric Liang, Tim Kraska, Armando Fox, Michael J. Franklin, and David A. Patterson. Generalized scale independence through incremental precomputation. In *Proc SIGMOD 2013*, pages 625–636, 2013.

3 Michael Benedikt, Julien Leblay, and Efthymia Tsamoura. Querying with access patterns and integrity constraints. *PVLDB*, 8(6):690–701, 2015.

4 Michael Benedikt, Balder ten Cate, and Efthymia Tsamoura. Generating low-cost plans from proofs. In *Proc. PODS 2014*, pages 200–211, 2014.

19th International Conference on Database Theory (ICDT 2016).
Editors: Wim Martens and Thomas Zeume; Article No. 2; pp. 2:1–2:2
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

5 Yang Cao, Wenfei Fan, Jinpeng Huai, and Ruizhe Huang. Making pattern queries bounded in big graphs. In *Proc. ICDE 2015*, pages 161–172, 2015.

6 Yang Cao, Wenfei Fan, Tianyu Wo, and Wenyuan Yu. Bounded conjunctive queries. *PVLDB*, 7(12):1231–1242, 2014.

7 Alin Deutsch, Bertram Ludäscher, and Alan Nash. Rewriting queries using views with access patterns under integrity constraints. *TCS*, 371(3):200–226, 2007.

8 Wenfei Fan, Floris Geerts, Yang Cao, Ting Deng, and Ping Lu. Querying big data by accessing small data. In *Proc. PODS 2015*, pages 173–184, 2015.

9 Wenfei Fan, Floris Geerts, and Leonid Libkin. On scale independence for querying big data. In *Proc. PODS 2014*, pages 51–62, 2014.