Saint-Petersburg State University
Mathematics and Mechanics Faculty

Vladimir Nazarenko

Evaluating the quality of approximate queries

Term paper

Scientific advisor:
Professor Novikov B. A.

Saint-Petersburg
2014

1 Introduction

Today we should deal with enormously large amounts of data. Consider structured data packed into some database. In some cases when you evaluate a query on these data you may accept an approximate answer in exchange for reduced evaluation time. So as payment for speed you bring losses of result's quality in a wide sense. A natural question is how much of the quality did we lost?

First of all, we need to move away from a concrete implementation of query or database in order to develop some generic approach which then can be applied to any DBMS. For this reason we use such a definition as q-set as it was defined in [1]. Using this abstraction allows us to compare queries in terms of distance in some vector field. In the meantime defining appropriate measure of a distance could be difficult.

Definition 1. *Q-set is an abstraction of both a query and the query result, in more detail it is a triple, consisting of a query, set of objects and scoring function for the set of objects which takes values from 0 to 1.*

Motivation for such definition is also present in [1]. Note that we treat the scoring function primarily as objects relevance for query.

For each query we consider its evaluation plan, consisting of elementary operations, such as set-theoretic, filtering and joins. We treat this plan as follows: initial set of objects we represent as a q-set with empty query (or, equally, query which returns all of the objects) and scoring function taking the value 1 for each object. Also we represent each operation in the query evaluation plan as a q-set. In these assumptions we can define quality of a query as aggregate of qualities of elementary operations.

2 Problem statement

Consider some abstract database and abstract query over it. Suppose that you have an ability to run this query approximately and hence get the approximate result. Apparently the approximate result may differ from the precise one and the difference is the error of evaluation. Our goal is to get a quantitative measure for the outlined error.

There is also one closely related task. Consider for each data entity in the database a quality measure. This quality measure can appear in heterogeneous systems. The natural desire is not to get just the most relevant to a query objects, but also get the objects with the highest quality. So the next our goal is to add this desire to our measure for a relevance error.

3 Related work

Our work lies at the junction of the three areas: data quality, data integration and the approximate querying. All of the above are intensively studying [2]

[3]. Nonetheless at the moment of writing this paper we have not found a comprehensive quality assessment for approximate querying in other works. Some works like [4] offer theoretical bounds for some approximate operations usually based on hoeffding or chebyshev inequalities but these bounds are too wide.

There had been many research on the data quality and these works shaped the basis for our study. Authors of the paper [5] defined the characteristics of the data quality such as reliability, physical and semantic integrity and also defined the steps to evaluate data quality.

Authors of [6] developed a resource allocation algorithm for approximate query processing, but this algorithm lacks quality measure both for approximate operation and for data. Therefore the mentioned paper can be considered as motivation for our work.

In the paper [1] described notion of q-set which was hugely used in our work and motivation for such notion.

Martin et al. in their paper [7] described a framework for data integration. The point of interest in this work for us is embedding quality assessment into data integration process which is related to our problem.

4 Operation quality

As was mentioned above we primarily look at the elementary operations and treat them as q-sets.

As the simplest example let's consider a top-k operation, which in naive implementation looks through a list of scores and takes k best objects. To make this operation approximate we can search for the best elements just at first 70% of the list and consequently we can get an inaccurate answer.

So what quality assessment do we need? First of all, for the sake of convenience this assessment should be normalized which would allow us to compare two assessments. Secondly, the assessment should be non-decreasing. This feature brings us some order on the approximate q-sets so we can, for instance, compare two approximate algorithms.

Assume

q^a - q-set, returned by approximate query evaluation algorithm

q^e - q-set, returned by precisely evaluated query (or maybe we got it by another way)

Then we came up with this analogue of F-measure used to assess the error of query evaluation:

$$Precision = \frac{\sum_{i=1}^{N}(\frac{1}{2}(q_i^e)^k + \frac{1}{2}(q_i^a)^k)^{1/k}}{\sum_{i=1}^{N}\lceil q_i^a \rceil}, k \to -\infty$$

$$Recall = \frac{\sum_{i=1}^{N}(\frac{1}{2}(q_i^e)^k + \frac{1}{2}(q_i^a)^k)^{1/k}}{\sum_{i=1}^{N}\lceil q_i^e \rceil}, k \to -\infty$$

$$F^\beta = (1+\beta^2)\frac{Precision * Recall}{(\beta^2 Precision) + Quality}$$

In the same manner we can define another measure to include quality of an object to assessment:

Assume a_i is a quality measure for object (e.g. defined as reliability of the source of this object)

$$Q_i^e = q_j^e * a_i$$

$$Q_i^a = q_j^a * a_i$$

$$Quality = 1 - \frac{2||Q^e - Q^a||}{||Q^e|| + ||Q^a||}$$

$$F^\gamma = (1 + \gamma^2)\frac{F^\beta * Quality}{(\gamma^2 F^\beta) + Quality}$$

The main drawback of these formulas is that they don't answer the question where we should get q^e?

5 Further research

This work is far from the finish and will be proceeded. The directions of the further investigation are following:

- Interpreting quality loss of an operation in terms of q-set is just a one way to do it, so we can try to do it another way (now the main idea is to compute quality loss statistically) and then compare and merge the results of the number of methods.
- As was stated earlier yet one question without an answer is how to define quality of data. For now discovered ways to do it are probabilistic databases [8] and peer reviewed data.
- Crucial problem is to define quality error for common querying operations such as top k, join, set-theoretic. Solving this and the previous problems may lead to developing a theoretical model for approximate query evaluation with confidence bounds.

References

[1] Boris Novikov, Natalia Vassilieva, and Anna Yarygina. Querying big data. In *Proceedings of the 13th International Conference on Computer Systems and Technologies*, pages 1–10. ACM, 2012.

[2] Swarup Acharya, Phillip B Gibbons, Viswanath Poosala, and Sridhar Ramaswamy. Join synopses for approximate query answering. In *ACM SIGMOD Record*, volume 28, pages 275–286. ACM, 1999.

[3] Torsten Schlieder. Schema-driven evaluation of approximate tree-pattern queries. In *Advances in Database Technology—EDBT 2002*, pages 514–532. Springer, 2002.

[4] Martin Theobald, Gerhard Weikum, and Ralf Schenkel. Top-k query evaluation with probabilistic guarantees. In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pages 648–659. VLDB Endowment, 2004.

[5] Stuart E Madnick, Richard Y Wang, Yang W Lee, and Hongwei Zhu. Overview and framework for data and information quality research. *Journal of Data and Information Quality (JDIQ)*, 1(1):2, 2009.

[6] Anna Yarygina and Boris Novikov. Optimizing resource allocation for approximate real-time query processing. *Computer Science and Information Systems*, (00):63–63, 2014.

[7] Nigel Martin, Alexandra Poulovassilis, and Jianing Wang. A methodology and architecture embedding quality assessment in data integration. *J. Data and Information Quality*, 4(4):17:1–17:40, May 2014.

[8] Nilesh Dalvi and Dan Suciu. Management of probabilistic data: foundations and challenges. In *Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 1–12. ACM, 2007.