

imate_query_processing_engines_with_machine_learning_models.docx

by Mohammed Anbar

Submission date: 21-Mar-2024 10:48AM (UTC+0800)

Submission ID: 2277289513

File name: imate_query_processing_engines_with_machine_learning_models.docx (66.68K)

Word count: 959

Character count: 5623

How to enhance approximate query processing by machine learning

Abstract

In terms of enhance data retrieval and processing the query response time is a core required in data environment such as streaming system, big data and analytics solution. In many applications, the ability of query functions well. However, to supporting huge data especially big data is expensive, because query response times are too long, methods based on running queries in parallel, and the high-performance requirement cannot be provided by regular method that calculate accurate answers.

In my literature review, I give a summary about approximate query processing (AQP), which seeks to quickly discover an approximate result as near of correct result by Machine learning.

“Feature selection: Looking through the space of potential feature subsets is a crucial first step in training high-quality machine learning models. Feature selection is one of the most expensive (yet important) processes in a machine learning workflow because of the exponential number of alternatives. Like in the earlier examples, it is acceptable to rule out a set of features if it can be determined that it is inferior than another set. This eliminates the need to calculate the precise prediction error of the model under these feature subsets”.[1]

Synthesis Matrix

Study	Authors	Year	Research Design	Key Findings/results	Limitations
1	Walter Cai Magdalena Balazinska Dan Suciu[2]	2019	Paper	Develop the use of bounds as apractical tool for multi join query optimization. In doing so, they introduce a method of further tightening state of the art query bounds via random hash partitioning and data sketching. “This approach to significantly narrowing state-of-the-art query	They do not focus on optimization time efficiency

				bounds via random hash partitioning and data sketching is useful when using bounds as a practical tool for multi join query optimization. This is a trade-off area between an increased partition budget and faster plan execution"[2]	
2	Kaiyu Guoliang[3]	2018	Paper	OLA scripts can be extended to support multiple tables on relational databases. Especially, the join queries are hard to predict in advance, and OLA is a typical method to solve such problem.	it is too expensive to compute the exact join size and we should estimate the join size.
3	Qingzhi Ma Yoav Freund Robert Schapire[4]	2021	Paper	AdaBoost was firstly proposed to solve classification problems, and was applied to solve regression problems later on	AdaBoost It is sensitive to noisy data and outliers, as these can disproportionately influence the weight distribution in the training process.
4	Tianqi Chen, Sameer Singh Ben Taskar Carlos Guestrin[5]	2015	Paper	GBoost usually uses only the first-order information, incorporate the second-order information in gradient boosting for conditional random fields, and improve its accuracy	Typically, the base models consist of a collection of regression and classification trees. Support for other regression models is lacking.
5	Tianqi Chen Carlos Guestrin[6]	2016	Paper	XGBoost is a state-of-art boosting method, and is widely used for competitions due to its fast training time and high accuracy. The objective of XGBoost is calculate loss function, and controls how close predictions	XGBoost has certain drawbacks, such as being a time-consuming, greedy algorithm and not requiring multi-threaded optimization.

				are to the targets. and regularization term, which controls the complexity of the model. Cache access patterns, data compression and sharding are essential elements for building a scalable end-to-end system for tree boosting.	
6	Qingzhi Ma Ali M. Shanghooshabad Mehrdad Almasi Meghdad Kurmanji Peter Triantafillou[7]	2020	Paper	More recently, learning methods for AQP have been developed. Such initiatives, which can lead to faster query response times and increased accuracy, include DBEst, DeepDB, deep generative models, etc. Models typically have lower space overheads than samples do.	DBEst keeps track of a regressor and a density estimator for every group of a grouping attribute. The size of the models can explode if there are a lot of groups, which would reduce their appeal for many inquiries.

Related Work

Many researches are ongoing to enhance the functionalities of RDBMSs by ML models. Cardinality estimation is a challenging subproblem for cost-based query optimizers and remains a primary challenge for even highly sophisticated systems. In use of bounds as a practical tool for multi join query optimization, we can tightening state of the art query bounds via random hash partitioning and data sketching [1], [2], [3], [4], [5], [6], [7], [8].

References

- [1] B. Mozafari and N. Niu, "A Handbook for Building an Approximate Query Engine," <https://web.eecs.umich.edu/~mozafari/php/publications.php>, 2015. Accessed: Mar. 05, 2024. [Online]. Available: <https://web.eecs.umich.edu/~mozafari/php/publications.php>
- [2] Walter Cai, Magdalena Balazinska, and Dan Suciu, "Pessimistic Cardinality Estimation: Tighter Upper Bounds for Intermediate Join Cardinalities," in *Proceedings of the 2019 International Conference on Management of Data*, in SIGMOD '19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 1553–1570. doi: 10.1145/3299869.3324958.

- [3] K. Li and G. Li, "Approximate Query Processing: What is New and Where to Go?: A Survey on Approximate Query Processing," *Data Sci Eng*, vol. 3, no. 4, pp. 379–397, Dec. 2018, doi: 10.1007/s41019-018-0074-4.
- [4] Q. Ma, "Approximate Query Processing Using Machine Learning," 2021. [Online]. Available: <http://wrap.warwick.ac.uk/162529>
- [5] T. Chen, S. Singh, B. Taskar, and C. Guestrin, "Efficient Second-Order Gradient Boosting for Conditional Random Fields," 2015.
- [6] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.
- [7] Q. Ma, A. M. Shanghooshabad, M. Almasi, M. Kurmanji, and P. Triantafillou, "Learned Approximate Query Processing: Make it Light, Accurate and Fast," ACM, 2020. Accessed: Mar. 05, 2024. [Online]. Available: <https://wrap.warwick.ac.uk/159826/1/WRAP-Learned-approximate-query-processing-make-light-accurate-fast-2021.pdf>
- [8] T. Lee, K. Nam, C. S. Park, and S.-S. Kim, "Exploiting Machine Learning Models for Approximate Query Processing," in *2022 IEEE International Conference on Big Data (Big Data)*, Dec. 2022, pp. 6752–6754. doi: 10.1109/BigData55660.2022.10020252.

ORIGINALITY REPORT

50%
SIMILARITY INDEX

48%
INTERNET SOURCES

45%
PUBLICATIONS

15%
STUDENT PAPERS

MATCH ALL SOURCES (ONLY SELECTED SOURCE PRINTED)

21%
★ wrap.warwick.ac.uk
Internet Source

Exclude quotes Off
Exclude bibliography On

Exclude matches Off

imate_query_processing_engines_with_machine_learning_mo

GRADEMARK REPORT

FINAL GRADE

/0

GENERAL COMMENTS

PAGE 1

PAGE 2

PAGE 3

PAGE 4