Check Out the Big Brain on BRAD: Simplifying Cloud Data Processing with Learned Automated Data Meshes

VLDB 2023

Author
Tim Kraska
MIT CSAIL
Amazon Web Services
kraska@mit.edu
timkrask@amazon.com

Author
Tianyu Li
MIT CSAIL
litianyu@mit.edu

Author
Samuel Madden
MIT CSAIL
madden@csail.mit.edu

Author
Markos Markakis
MIT CSAIL
markakis@mit.edu

Author
Amadou Ngom
MIT CSAIL
ngom@mit.edu

Author
Ziniu Wu
MIT CSAIL
ziniuw@mit.edu

Author
Geoffrey X. Yu
MIT CSAIL
geoffxy@mit.edu

Reviewer
王偉力
資訊工程研究所碩一
R12922116

ABSTRACT

The paper introduces BRAD, a novel system designed to simplify cloud data processing by automatically integrating and managing diverse data systems into an optimized data mesh. Leveraging machine learning, BRAD intelligently routes queries to the most suitable data systems, enhancing both performance and cost efficiency without requiring deep system knowledge from the user.

1. PROBLEM DEFINITION

The paper addresses the complexity and inefficiency faced by organizations that use heterogeneous data systems to handle various workloads. These systems often require manual configuration and optimization that is too complicated for human experts, which may lead to suboptimal performance and high costs.

2. PRIOR WORK

Previously, data management approaches primarily focused on either enhancing individual system performances or manually integrating multiple systems, which does not scale well with the increasing complexity and size of modern data ecosystems.

3. SOLUTION

As shown in Figure 1, BRAD creates an instance-optimized data mesh by automatically integrating various data systems. It manages these systems in a way that abstracts their complexities from the end-users, who interact with what appears to be a single unified system. This is achieved through a combination of machine learning techniques and cloud-based technologies.

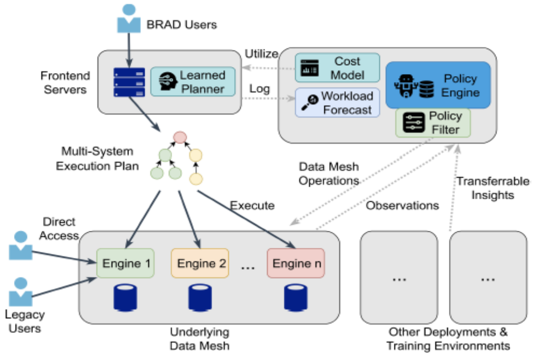


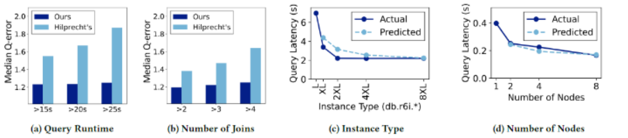
Figure 1: BRAD intelligently serves a unified data API with specialized engines, leveraging the modern cloud to provide an auto-scaling, management-free DBMS

4. RESULT

The initial results indicate that BRAD can not only accurately predict the query execution time, but it also can significantly improve the performance and cost efficiency of data processing tasks under the user-defined constraints.

4.1 Prediction Accuracy

As Figure 2 shows, compared to the prior work (Hilprecht's), the proposed cost model predicts more accurately to unseen IMDB queries with various runtimes (3(a)) and join predicates (3(b)) respectively. (Q-error is defined as max{predicted/true, true/predicted} - better estimates have a Q-error closer to 1). Moreover, the prediction accuracy of query runtime on instances with unseen types (3(c)) or node counts (3(d)) is also beyond expectations.



(a) Query Runtime

(b) Number of Joins

(c) Instance Type

(d) Number of Nodes

Figure 2: Cost model performance overview: (a), (b) Our cost model generalizes to unseen queries with longer runtimes or more join predicates. (c), (d) We accurately predict query runtime under different instance types and/or number of nodes.

4.2 Performance-cost Improvement

As Figure 3 illustrates, BRAD can correctly predict (the dashed lines on the graph) that all three queries will run under 10 seconds on one dc2.large node—Redshift's most economical instance type. BRAD applies this change and reduces the mesh's monthly Redshift cost by 4× (bottom graph).

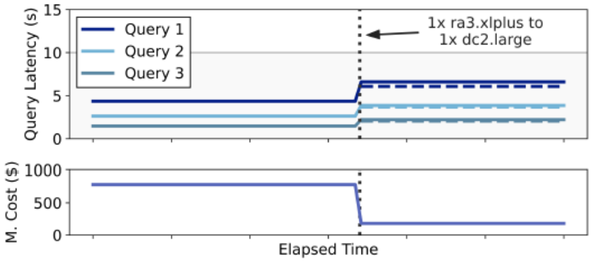


Figure 3: BRAD changes Redshift instance types to reduce cost (bottom) as it correctly predicts that query latency will remain below a user-specified ceiling (top, shaded).

5. CRITIQUE

In my opinion, the most important contribution of this paper is that it introduces the envision of BRAD - using machine learning techniques to autonomously manage cloud data systems. This epoch-making concept of BRAD may inspire other researchers in the field of DBMS, and one may build a perfect version of BRAD one day based on their work.

6. EXTENSION

In the paper, the authors said that they have tried to translate different SQL dialects between systems via LLMs, but the results suffer from performance overhead and hallucination risk. They suggest that developing robust, interpretable translation schemes with LLM-like techniques instead of static rules is a promising research direction to address this challenge. If I am to build a project based on this paper, I will try several prompt engineering techniques to possibly decrease the overhead or reduce the hallucination risk.