Query-Driven Learning for Next Generation Predictive Modelling & Analytics

University of Glasgow
School of Computing Science

Fotis Savva, University of Glasgow
f.savva.1@research.gla.ac.uk

Essence: Pervasive & Distributed Intelligence

Introduction

The aim of this work is to address the exponential increase in data and computational times by **Approximate Query Processing (AQP)**. However, instead of a sampling based approach (S-AQP) [2] we use a **Query-Driven Learning (QDL)** [1] approach. We train **Machine Learning (ML)** models that are able to estimate the answers of future queries using historical workloads.

Contributions:

1. Offer a light-weight, complimentary aggregate estimation engine that can be stored locally.
2. Agnostic to data backend. Can be used alongside relational databases, S-AQP etc.
3. Highly accurate and efficient estimations

Query-Driven Methodology

QDL uses past and incoming queries to learn query patterns and be able to build ML models that can estimate the results of new queries.

Query Representation :

- Each Aggregate Query (AQ) is represented as a vector by extracting its filtering parameters.
- Any aggregate (COUNT/AVG/SUM, etc) is supported...
- Each query is represented as $q = (\boldsymbol{m}, y)$

Partitioning (Clustering):

- The set of queries, $C = \{(\boldsymbol{m}, y)\}_{i=1}^{n}$,is partitioned for better result estimation.

$$C = \cup_{i=1}^{K} C_i$$

$$C_i \cap C_j = \emptyset \,, i \neq j$$

- Each subset has a representative $\boldsymbol{W} = \{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_k\}$

ML Model Association

- Every subset is used to train a supervised regression model.
- A set of ML models is created $M = \{f_1, \ldots, f_K\}$ which are associated with the representatives W

Answer Prediction

- A prediction is made based on an ensemble scheme incorporating the predictions of the closest representative.

$$\hat{y} = \sum_{k=1}^{K} \mathrm{I}_k f_k(\boldsymbol{m})$$

- Where I_k is an indicator function evaluating to true if : $w_k = \arg\min_{i \in [K]} \|\boldsymbol{m} - \boldsymbol{w}_i\|$

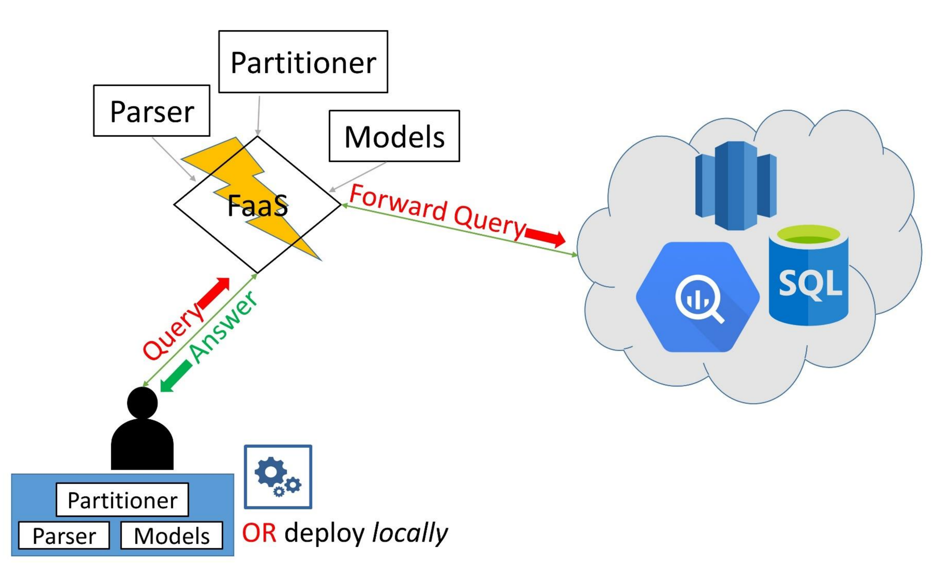

Figure 3. Possible Cloud Deployment

System Architecture

Parser (Query Vectorization)

- The query is initially parsed and a vectorised representation is produced

Partitioner (Query Clustering)

- The partitioner maps the query to the closest representative and **associated** model

- During ***training mode*** the queries are monitored and their answers are retrieved from the Data Store/S-AQP
- In ***prediction mode*** the answers can be estimated via the models

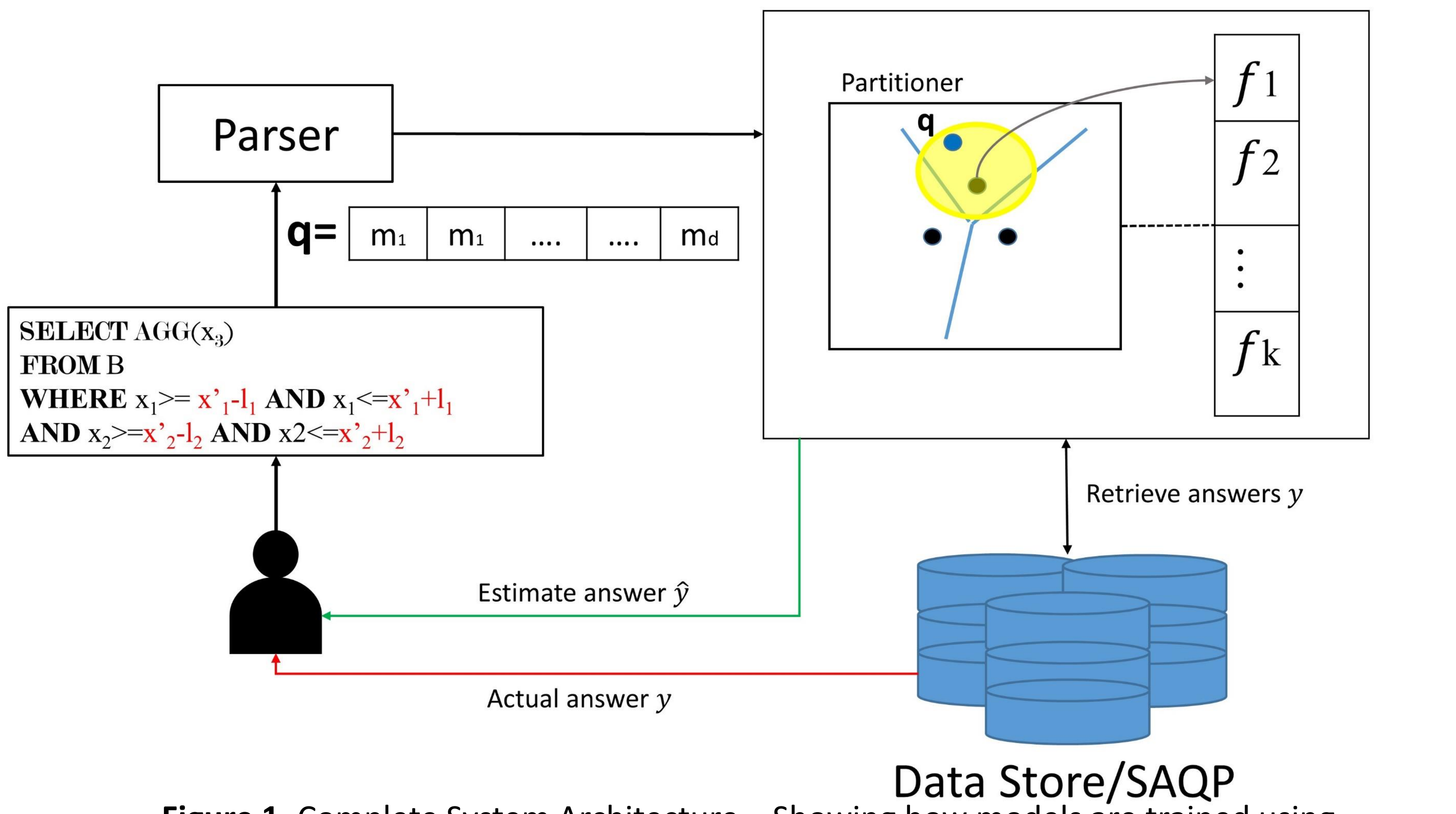

Figure 1. Complete System Architecture – Showing how models are trained using parsed queries and how predictions are served through models.

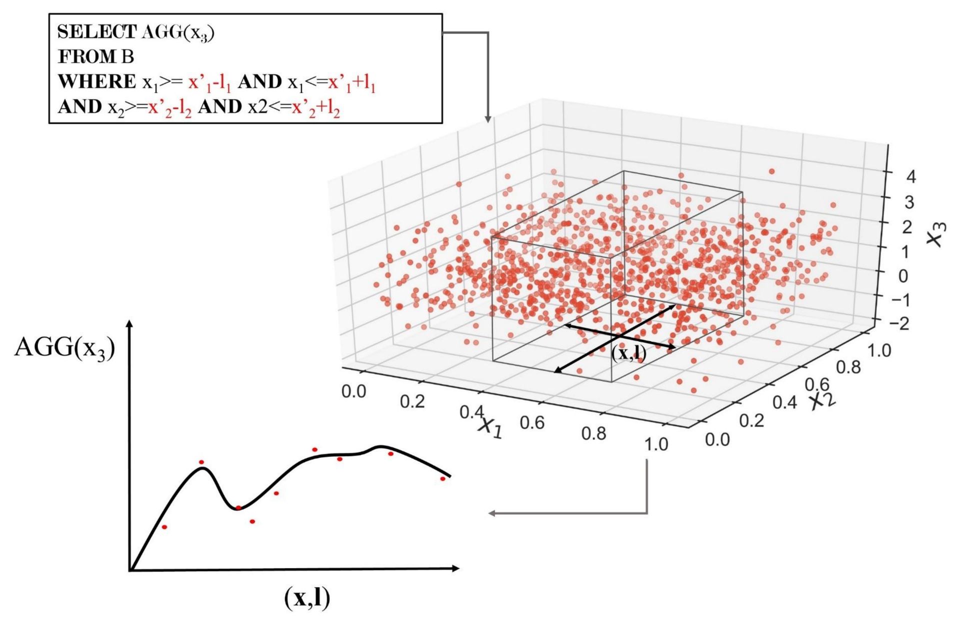

Figure 2. Representing range queries in multi-dimensional space and then associating these vectors with their results.

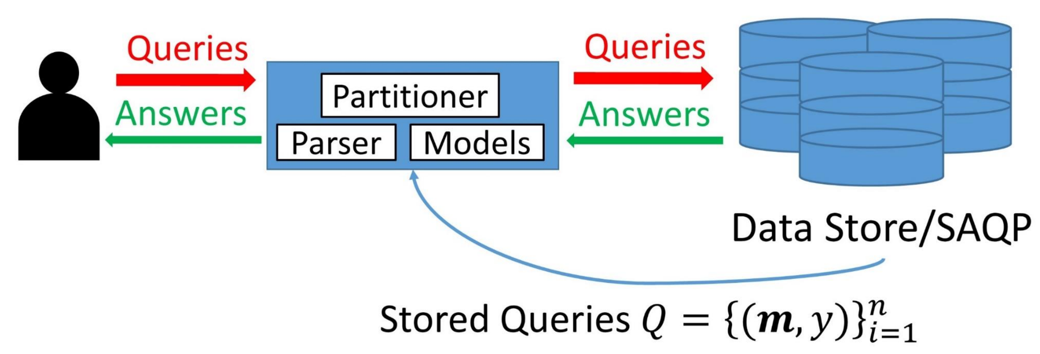

Figure 4. Training Mode

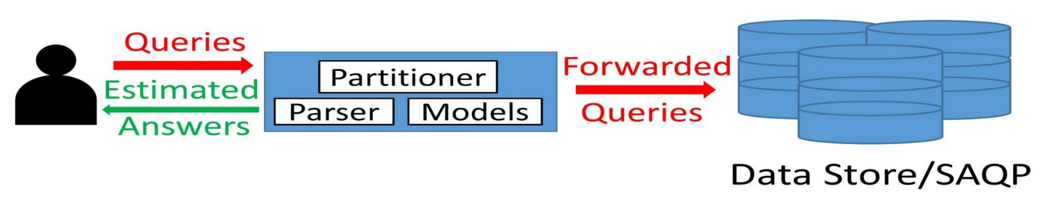

Figure 5. Prediction Mode

Evaluation Results

- Datasets Used: **Crimes** and **TPC-H (1GB)**
- Partitioner : K-Means - Model : XGBoost
- Experiments ran single threaded on Linux Ubuntu 16.04 using an i7 CPU at 2.20GHz with 6GB RAM

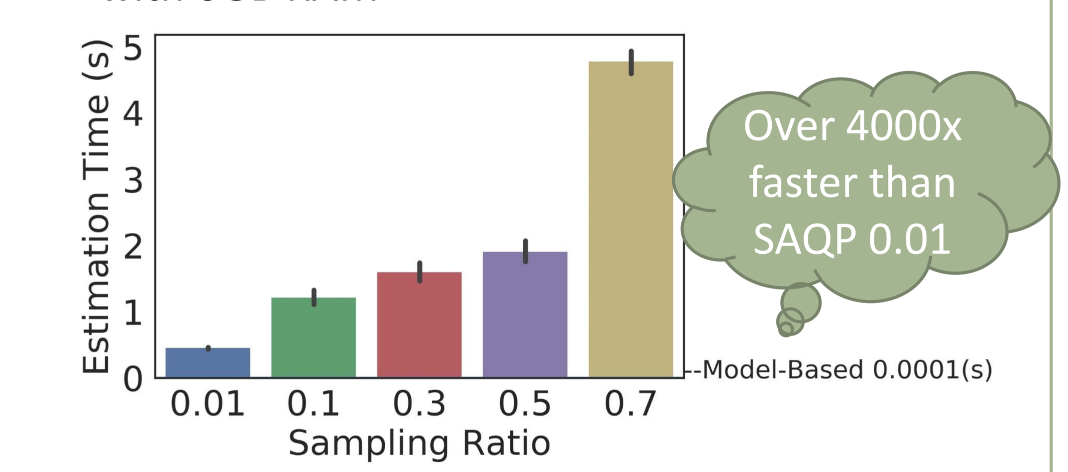

Figure 6. Performance comparison of state-of-the art SAQP (VerdictDB[2]) and model-based approach.

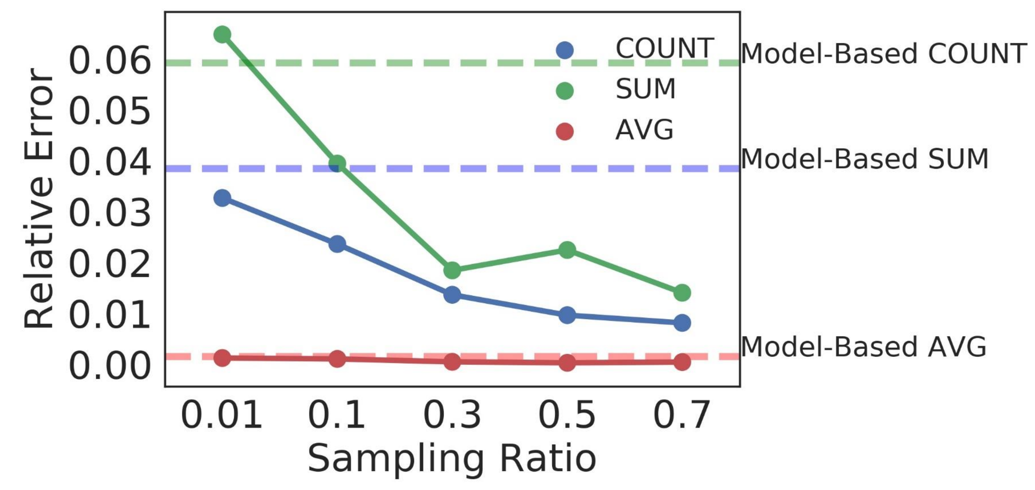

Figure 7. Relative Error for **Crimes** using SAQP VerdictDB[2] and model-based approach..

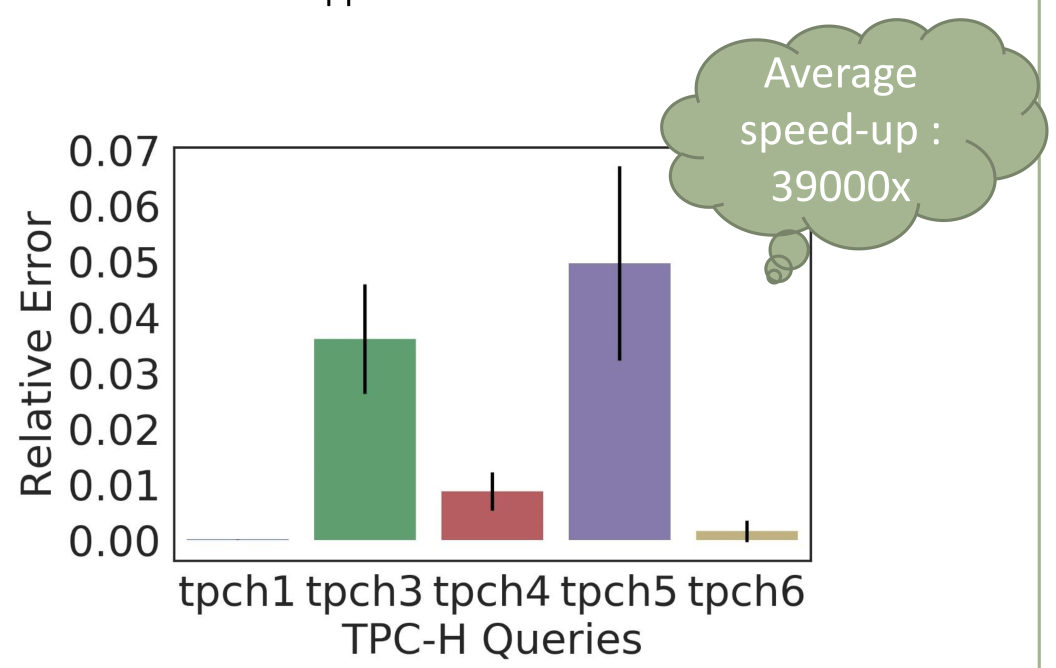

Figure 8. Relative Error for sample of **TPC-H** using model-based approach..

Bibliography

1. Anagnostopoulos, Christos, Fotis Savva, and Peter Triantafillou. "Scalable aggregation predictive analytics." *Applied Intelligence* 48.9 (2018): 2546-2567.
2. Park, Yongjoo, et al. "VerdictDB: universalizing approximate query processing." *Proceedings of the 2018 International Conference on Management of Data*. ACM, 2018.

Contact

Fotis Savva
School of Computing Science, University of Glasgow
Email: **f.savva.1@research.gla.ac.uk**