National College of Ireland

Machine Learning Approach For Predicting The SQL Query Resource Usage

MSc Research Project

Data Analytics

Sanket Sud

Student ID: 21123888

School of Computing

National College of Ireland

Supervisor: Aaloka Anant

National College of Ireland
MSc Project Submission Sheet
School of Computing

Student Name: Sanket Shivaji Sud
Student ID: x21123888
Programme: Master Of Science **Year: 2022-2023**
Module : Reaserch Project
Supervisor Submission : Aaloka Anant
Due Date : 15-12-2022
Project Title : Predicting The SQL Query Resource Usage Using ML
Word Count : 4227 **Page Count: 18**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:
Date:
PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Machine Learning Approach For Predicting The SQL Query Resource Usage

Sanket Sud
21123888

Abstract

The growth of the internet over the century happened because of a lot of information available on it. The most crucial item is data, which can serve as a source of knowledge. There is always more information to access and process. Query processing is taken into consideration while acquiring access to a large amount of data. We must consider the CPU and memory requirements for the query execution while processing the data. We can take the machine learning approach into account for forecasting the resources for a certain query. This will be useful for large setups that process a lot of data and require a lot of computing power. Machine learning helps distribute the workload of the requests. This method aids in scaling and resource management for both small- and large-scale data.

1 Introduction

The fundamental goal of the database is to store a lot of data in an effective manner that is quick and simple to access. When analyzing and retrieving data from databases, query statements are crucial. There are many various sorts of databases, but there are primarily two: structured databases, which help to store data with a framework like rows and columns, and unstructured databases.

Examples of structural databases are MySQL and Postgres. Different query statements are used to acquire and process the data from this database using SQL (Structured Query Language), which is utilized to access the data. Data modifications and alterations are regularly made to the data using DML processesKandov et al. (2021). The findings of this Tang et al. (2021) demonstrate that users are not consistently using all of the resources allotted to the database system throughout all of its operations. Resource usage depends entirely on the

query statement users execute within a given time frame. Users' activity fluctuates throughout the day.

We are considering the run time of the query in this research. The query is executed after several steps. Data Acquisition (From, Join), Row Filter (Where), Grouping (Group by), Group Filter (Having), Return Expressions (Select), Order & Paging (Order by & Limit / Offset), Seser et al. (2022).

The time it takes to run a SQL query depends on how much data there is to process. When queries run out of memory or Arzamasova et al. (2020), they typically fail. Based on the user-provided statement and table metadata, we can forecast the query execution time and memory usage.

These benefits of this forecast include: 1. CPU and memory usage: The user learns how much CPU or memory will be used by a specific query statement. 2. Resource allocation: Prediction times and CPU consumption provide insight into resource usage and needs. The distribution of resources depends entirely on certain user uses during a given period of time.

We are adopting a machine learning strategy for query cost prediction in place of the conventional approach, which is beneficial. The accuracy of the model's predictions rises along with the size of the metadata, something that is not feasible with conventional query execution Slivinskas et al. (2000). The metadata contains all of the information related to the processing and execution of queries. The more metadata there is, the more useful it is for training the model;

2 Related Work

This study's primary goal is to determine how much of each resource is being used when a query statement is conducted against a database. To understand the resources needed to run a specific query, it is important to carefully analyze and examine the different elements that can affect how well a query performs. Unstructured databases Hassan (2021) are one of two forms of databases. In this project, SQL statements are the major focus which is part of structure databases. The two most crucial aspects we are concentrating on in this study are query optimization and cost analysis, which encompasses memory consumption and time

requirements. Many studies have been done to determine query costs based on various environments, databases, and organizational data utilization. To reduce resource usage, the machine learning approach is utilized to calculate the cost of the query for various organizations.

2.1 SQL query performance and resource allocation.

The databases are specially managed and maintained by the database management system (DBMS). Relational database management (RDBMS)is the conventional method for storing data . The essential component of RDBMs is the database structure, which can be either columnar or row-based.An (2009) The database's performance is significantly influenced by the data's format. Big data enters the picture as the size of the data grows throughout the decade. To manage such a massive volume of data, the hive databases, which are based on the SQL engine, are utilized.Thusoo et al. (2010)

The processing of large amounts of data necessitates the location of more resources. A single SQL query requires a minimum setup ranging from 512 KB to 2 GB. The 2019 server must have a minimum of 2 GB of RAM, 6 GB of memory, and 2.0 GHz of processor power. The configuration and resource usage both rise as the data volume rises. This configuration is thought of as the minimal resource allocation. The user must always set this allocation to the maximum in order for the query to process more quickly and receive a response more quickly.Windows (2010)

Even though these resources aren't constantly used, they nonetheless need to be allocated for the system to function better. Many server operations are carried out to process data, some of which don't need all the resources, and the opposite is also true. In some circumstances, the query execution may exhaust the server's resources and fail.Tang et al. (2021)

2.2 Predicting the cost of queries using traditional approach

Different forms of fired queries can be used for cost prediction. Resource managers can perform workload allocation using QPP (Query performance pre-

diction) so that interactive behavior is accomplished or particular QoS targets are reached. Instead of considering the entire amount of labor, optimizers might choose between different plans based on projected execution latency. The schema created during the database is primarily included conventionally. Using the transitive closure approach, the cost of recursive queries is projected in various environmentsAndres & Viegmont (1991). The new method for query cost prediction is used in paper Teshome & Chung (2010), which first considers the conventional hard disk-based cost prediction for queries before considering flash memory-based databases. The advantages of cost prediction are demonstrated in the study Wu et al. (2013). It focuses on the Postgres database in particular to analyze the cost of queries against it.

2.3 Prediction Using Machine Learning

Machine learning can be used to anticipate costs for a certain setting. The study Tang et al. (2021) concentrates on Twitter to forecast the cost for questions using a machine learning strategy that focuses on the classification of the query into 3 different types mainly low, medium, and high resource-utilizing queries. Low queries use less than 1 Mb of memory, whereas medium queries use 1 Mb to 1 GB of memory. This paper Zhi Kang et al. (2021), which examined resource usage by queries in both cloud environments and traditional databases, looked at the pattern of resource consumption. It also emphasizes the differences in query utilization and resource usage patterns between different businesses. To determine the cost of the searches, deep learning pipelines are utilized. The focus of the study Guo & Xu (2021) is on various cost prediction queries. The Cost Prediction Range query (CPRQ) consists of two things: first, it extracts the query feature based on training data, and second, it forecasts CPU and memory usage based on test data. In this experiment, a database of moving objects is utilized to forecast costs. The focus of this paper's Burdakov et al. (2020) is on estimating the SQL cost for the Spark platform. To comprehend the query's structure and analyze its cost, it is broken into sub-questions.

2.4 Model evolution for predictions

The cost of the query can be predicted using a variety of models. The Bag of Words (Bow) model is the main topic of this study Tang et al. (2021). The SQL query is viewed as a number using this paradigm. To analyze the statement, take into account the statement's conversion to a number. This also uses an NLP strategy to turn comparable words into vectors. Additionally, Bayesian Nets, Support Vector Machines, and Linear and Multiple Regreavailable data and meditation. The results of this model are experimented with and improved.Akdere et al. (2012). The model employed Kernel Canonical Correlation Analysis (KCCA) and Support Vector Machines (SVMs) to provide great accuracy in the data.

The model in this paper Guo & Xu (2021) are mainly polynomial regression, decision tree, random forest, and KNN with Euclidean distance. The main motive to use this model is the first model should balance the result for an unbalanced dataset and the second should give high accuracy even with less sample size

3 Research Methodology

The research is focused on the time required and memory consumption when a query statement is running. Based on the kind of queries that are executed and their characteristics, the memory consumption and time required can be utilized to examine both the maximum and minimal resource allocation. This project employs the Cross-Industry Standard Process for Data Mining (CRISP-DM) approach. The CRISP-DM approach adheres to professional norms. This process aids in breaking down the project's research into manageable steps.

3.1 Business and Data Understanding

Machine learning is used to solve business as well as technical problems. In light of this scenario, machine learning can be used to advance technology. The available data and metadata have a significant impact on the usage of machine learning in the technology sector. The research in this field demonstrates that ML may be used extremely effectively, yielding high-quality results while saving

time and human labor. The project's main goal is to make resource allocation for data processing better With the help of machine learning, we can estimate the cost of a SQL query while also getting a general prediction of how much memory will be used during execution. The server can plan its resource allocation using this information. Consider the example as a particular query consuming more memory than the memory available on the server then the query fails to run. If we can predict the memory usage before running, we can allocate the extra resources.

The data used in this project is metadata generated after running queries on Big Query. The dataset used in the project is GitHub users and its public. We are focusing on DML (Data Manipulation Language) in this project Insert, delete, and update operation. The query run on the dataset is autogenerated. The metadata contains almost 30k records and 28 columns.

3.2 Preparation of the Data

The data used in this project is the metadata of the queries. The big query is used to produce the data. The main feature of BigQuery is we can run SQL queries on big data and it also gives running time as well as the memory used after the query is run. The steps involved in this to generate the data are as follows:

1. Generate the query for the dataset mainly including DML operations.
2. Run the query statement (Insert, update, and delete) on the selected dataset using python Script
3. Gather the project history (including all the metadata related to the queries) of the dataset.
4. Creating a new BigQuery dataset to store the project history.
5. Fetch the newly created dataset from google colab

The data generated after the above step is used as a dataset for the research project. This dataset contains almost 28 columns and it also includes 8 custom

attribute which has a good correlation with the dependent variable. We are 18 attributes which are having good correlation out of which 16 attributes are independent and 2 are dependent.

3.3 Modeling

The project includes the use of google colab for coding. The google colab helps to write and execute python code on the browser. Google colab provides an environment where we can run save and execute code. This way we don't have to utilize offline resources. We are using the different models in this project to improve the accuracy of the prediction and to reduce the delay in the response.

K-nearest neighbors (KNN):

The KNN algorithm is used to find the k data points which are nearest to the actual distance. The KNN can be used for classification as well as regression. We are using KNN for regression to find K points that are nearest to actual points. KNN is a supervised model in which we find k Points from the training dataset and applied to the target variable. Need to find the mean of the target value of this K- nearest neighbors.

$$\hat{y}(x) = \frac{1}{k}\sum_{x_i \in N(x)} y_i$$

The model used in paper Guo & Xu (2021) uses Euclidian distance to find the k point. The KNN can be applied to this type of metadata as the main queries can be considered as low medium and high time as well as memory usage which can be clustered according to the parameter. The main motive to use regression instead of classification is to find the definite value of the time and memory required. This value can be predicted using the mean absolute value of the result obtained by the K point. There are a lot of advantages of using KNN regression for prediction which meets the research requirement. It's simple to apply. we are deciding K value which comes with high flexibility and fewer boundaries. The prediction is done on run time so the model keeps improving. There is no need to tune hyperparameters cause only one parameter is available which is the K value

Linear Regression

Linear regression is used to find the relationship between dependent and independent variables by fitting a linear equation. We can predict dependent variables using linear regression. the paper Guo & Xu (2021) uses polynomial regression for the prediction which is considered a special multilinear regression. The formula for linear regression is as follows : Y is the dependent variable, b is the slope,

$$Y = a + bX$$

and X is considered as the variable. The main advantage of this regression can satisfy the requirement of the research data. The algorithm is easy to implement and easy to scale. The more data high the accuracy. No hyperparameter tuning require in this model.

Ridge Regression:

Ridge regression is considered a model-tuning method. This is helpful when data have multicollinearity. These models are used to correct the predicted values based on the L2 Regularization. Used to reduce the error values for predicted values and try to fit the line with minimum error as the data increases or is predicted by the model. The formula for the ridge regression is as follows: Where Y is the

$$Y=XB+e$$

dependent variable, X stands for the independent variables, B for the predicted regression coefficients, and e for residual errors. The main advantages of ridge regression are helpful in this research project. It helps to regularize the coefficient estimated by the Linear regression. It tries to reduce the error produced by Linear regression. helpful for the data less dependent attribute.

3.4 Evaluation

The result of the model is evaluated based on the different parameters and the type of data used in this research as well as the need of the business. the

accuracy of the model needs the o to be calculated to understand the correctness of the model for the particular data. This accuracy helps to improve the model in the future and also helps to choose the correct model for this metadata dataset. There are some statistics available to get the correctness of the model such as Root Mean Square Error (RMSE), Mean Square Error(MSE), and Mean Absolute Error (MAE). Several factors affect models such as error value for example Bias Error, Variance Error, and Random Error. The different matrices used to measure the performance of the model contain confusion matrix accuracy and precision. We are focusing on the following parameter to find the correctness of the model.

- Root Mean Square Error (RMSE) is used as standard matric for getting the error.
- The mean square error (MSE) square value of the difference between the predicted and actual value

3.5 Deployment

The research's findings are all contained in the deployment phase. The KNN Regression model and Linear Regression with Ridge Regression for regularization were used to predict the amount of time and memory needed for the queries. All of the models, forecasts, and pipelines used to create the dataset are included in the deployment.

4 Design Specification

We are using a python programming language for the research project and a cloud-based platform used to run python. The model used is mainly K Nearest neighbor, Linear regression, and ridge regression. The dataset generated is directly accessed from BigQuery. The dataset can also be available in CSV file format. The data used to create the model and run into google colab. The matplotlib is used to plot and analyze the data using figures.

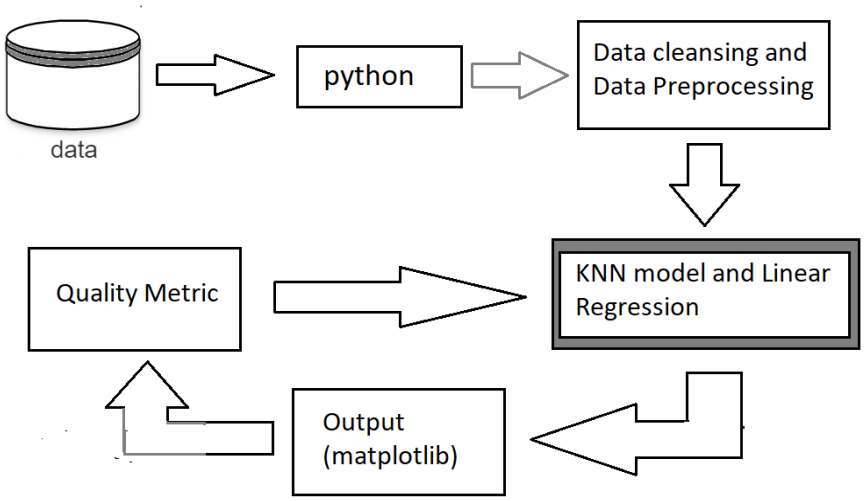

Figure 1: Design Flow Overview

5 Implementation

The implementation includes direct access to BigQuery using google colab. we can connect the BigQuery using 2 ways:

1. Connect using the google colab library which can directly use authentication.

2. Connect using the service account created on the platform providing all the access to the service account.

BigQuery provides a lot of features to run the query and export the data. It also provides the feature to access the project history. The project history contains a lot of information about the query once it's completed. The project history contains almost 108 columns which have all the information of the query.

5.1 Data transformation

The dataset contained information queries run on 5 different tables. The dataset contains almost 108 columns which have lots of information about the running query out of which 28 columns are useful. The dataset contains all the nested information within the columns which need to be extracted and cleansed

properly. There are 8 custom attributes used for the prediction. The custom-generated attribute has a good correlation with the dependent variable.

The following are the attribute used from the dataset:

- total_bytes_processed: Total bytes used in running query
- total_slot_ms : time used by each slot in a microsecond
- status_input : status of the query,
- time_require : This is the time required to execute the query
- statement_type: Type of statement (Delete, Insert, Update)
- state : state of the query (Active or Inactive)
- priority : Priority of the query
- cache_hit: Cache hit status

Following are the custom-generated attribute:

- Number_of_keyword :Number of keywords used in the query
- Number_of_word_in_query: Number of words in query
- Number_of_operations: Number of operations used in the query
- Number_of_char_in_query: Number of chars in the query
- Number_of_integer_attribute_in_table
- Number_of_string_attribute_in table
- Number_of_rows_in_table
- Number_of_column_in_table

The total_time_require for the query is the difference calculated from start time and end time. The number of keywords can be calculated by comparing the list of keywords given by the SQL. The following are visual analysis:

The following are the time required according to statement type:

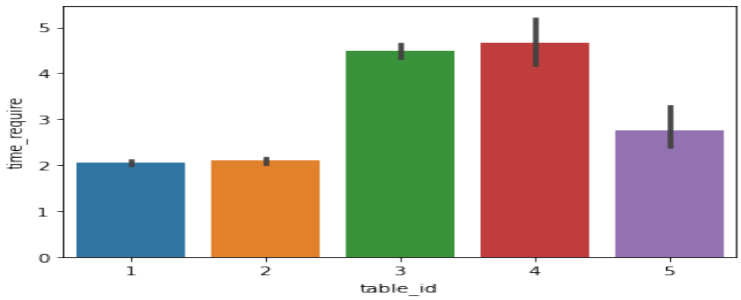

Figure 2: Table Id vs Time Require

Figure 3: Statement Type vs Time Require

5.2 Model Development

In this project, we are using two models for prediction one of them is KNN (K nearest neighbor) in which we find the K point in the data. The data is split into two parts which we consider 70 % as train data and 30 % as test data. The data include the operation on 5 different tables with different columns and rows. We are also using 4 tables for training data and 1 table info as testing data. This shows that the result can be applied to any table. The project includes the sklearn library for modeling. we are using the following libraries to build the project and model creation.model_selection, JSON, google colab, google cloud.

5.2.1 K-nearest neighbors (KNN)

We are using KNN for predicting the time required for the query execution. The model is implemented using different test cases and attributes. The main focus of KNN was to reduce errors. We are trying to find the K points. The KNN model is implemented from the sklearn library. We are tuning the value of K and finding the correct value which gives minimum error. The value of the K needs to be tuned.

5.2.2 Linear Regression

We are using Linear regression to predict the time required for query execution. The model was implemented using different test cases. The main focus in Linear regression is to reduce the error. The model is implemented using the sklearn library. We are focusing on variance errors and bias errors and also trying to reduce them.

5.2.3 Ridge Regression

The ridge regression is used for the regularization of the linear regression. It helps reduce errors using predicted results. The main motive for using this technique is to increase accuracy.

6 Evaluation

We are talking about the models' evolution in this section. It is discussed why linear regression and KNN should be used. The model is subjected to several experiments to produce high accuracy and minimize inaccuracy. Additionally, we are adjusting the parameters to get good and sklearn.

6.1 Experiment / Case Study 1

In the first experiment, we tried to train the based on the KNN model. The dataset is split into different datasets for training and testing. We are splitting data into 80 percent into training and 20 percent into testing. We are predicting

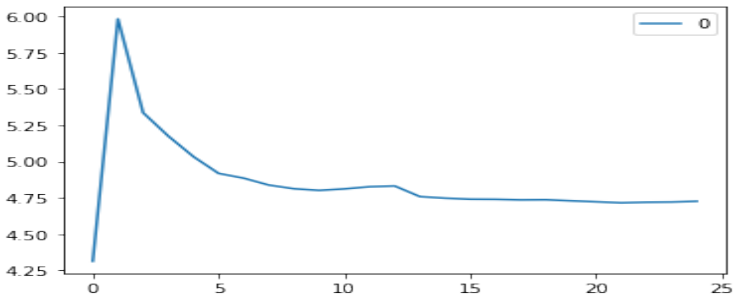

Figure 4: Design Flow Overview

the error for all the K values from 1 to 25. The minimum error for the value of the K is 15 for the first experiment.

6.2 Experiment / Case Study 2

In the second experiment, we tried to train the data based on the KNN model. The dataset is split into different datasets for training and testing. We have a dataset that contains information on the metadata related to the 5 different tables, which 4 tables' metadata is used for training and the last tables dataset for testing. We are predicting errors for the values for the K values from 1 to 25. Finding the

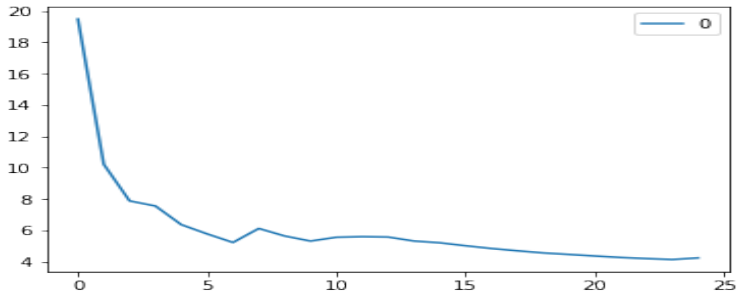

Figure 5: Design Flow Overview

minimum error looping through these values.

6.3 Experiment / Case Study 3

In the third experiment, we tried to apply the linear regression model to the data. The data set is split into two different for training and testing. The main motive behind using this model is to reduce the error rate which we have in the KNN model. We are splitting data into 80 percent into training and 20 percent into testing. The ridge regression is applied to the model for normalization purposes.

6.4 Experiment / Case Study 4

In the fourth experiment, we tried to apply the Linear regression model to the data. We have a dataset that contains information on the metadata related to the 5 different tables out of which 4 tables' metadata is used for training and the

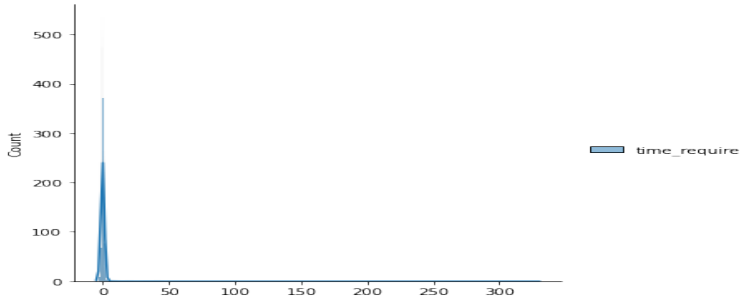

Figure 6: KDE for Actual vs Predicted

last tables dataset for testing. The ridge regression is applied to the model for normalization purposes.

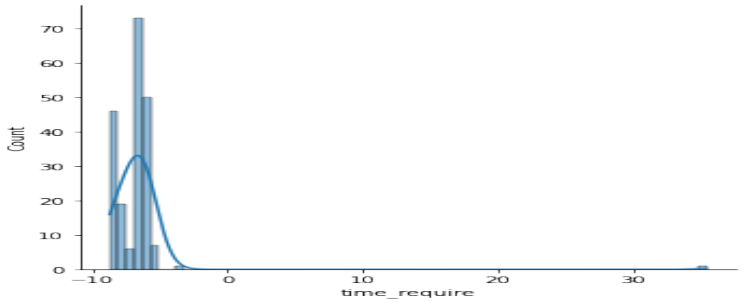

Figure 7: KDE for Actual vs Predicted

The following table shows the experiment and the results :

Experiment Number	Error Value
Exp 1 (KNN)	6.14
Exp 2 (Linear)	-34.88
Exp 3 (KNN)	4.82
Exp 4 (Linear)	-95.32

7 Conclusion and Future Work

The research project focuses on the question “Can resource allocation in DML SQL queries be predicted using machine learning?”. The main objective of the research is to create a machine learning model which is helpful for the prediction of the memory usage and total bytes processed by the SQL queries. We are using two models for the prediction which include the KNN model and Linear regression for simple prediction. Both of these models give less error and good performance. The KNN model performs better and gives a good result as compared to Linear regression. The data shows that linear regression model performance improves when normalized using ridge regression. We are also performing 4 different experiments for the dataset for a model to improve. The result of the evolution shows that we can predict the time and total bytes used using the machine learning model, which concludes our question. We are limiting the scope of the project to DML operations because they are frequently used for data processing. The resource consumption and time requirement prediction help the organization or user to schedule the resource usage according to need. It also helps to reduce costs and improve performance.

The future scope of the project is to implement different SQL statements like Data Definition Language(DDL), Data Query Language(DQL), Data Control Language(DCL), and Transaction Control Language(TCL). This prediction can reduce costs by minimizing resource utilization for data processing.

References

Akdere, M., Çetintemel, U., Riondato, M., Upfal, E. & Zdonik, S. B. (2012), Learning-based query performance modeling and prediction, *in* '2012 IEEE 28th International Conference on Data Engineering', pp. 390–401.

An, M. (2009), Column-based rle in row-oriented database, *in* '2009 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery', pp. 309–315.

Andres, F. & Viegmont, Y. (1991), Estimating recursive query costs for various parallel environments, *in* '[1991] Proceedings The Fifteenth Annual International Computer Software Applications Conference', pp. 365–372.

Arzamasova, N., Böhm, K., Goldman, B., Saaler, C. & Schäler, M. (2020), 'On the usefulness of sql-query-similarity measures to find user interests', *IEEE Transactions on Knowledge and Data Engineering* **32**(10), 1982–1999.

Burdakov, A., Proletarskaya, V., Ploutenko, A., Ermakov, O. & Grigorev, U. (2020), Predicting sql query execution time with a cost model for spark platform.

Guo, S. & Xu, J. (2021), 'Cprq: Cost prediction for range queries in moving object databases', *ISPRS International Journal of Geo-Information* **10**(7).

Hassan, M. A. (2021), Relational and nosql databases: The appropriate database model choice, *in* '2021 22nd International Arab Conference on Information Technology (ACIT)', pp. 1–6.

Kandov, I., Aleksandrov, A. & Goranov, G. (2021), The use of different sql servers for resource optimization on mobile sensor systems, *in* '2021 12th National Conference with International Participation (ELECTRONICA)', pp. 1–3.

Seser, T., Pleština, V. & Marjanica, F. (2022), Performance analysis of sql prepared statements in crud operations, *in* '2022 7th International Conference on Smart and Sustainable Technologies (SpliTech)', pp. 1–5.

Slivinskas, G., Jensen, C. & Snodgras, R. (2000), Query plans for conventional and temporal queries involving duplicates and ordering, *in* 'Proceedings of 16th

International Conference on Data Engineering (Cat. No.00CB37073)', pp. 547–558.

Tang, C., Wang, B., Luo, Z., Wu, H., Dasan, S., Fu, M., Li, Y., Ghosh, M., Kabra, R., Navadiya, N. K., Cheng, D., Dai, F., Channapattan, V. & Mishra, P. (2021), Forecasting sql query cost at twitter, *in* '2021 IEEE International Conference on Cloud Engineering (IC2E)', pp. 154–160.

Teshome, S. & Chung, T.-S. (2010), Query cost estimation for read intensive flash memory based database systems, *in* '2010 International Conference on Electronics and Information Engineering', Vol. 1, pp. V1–496–V1–498.

Thusoo, A., Sarma, J. S., Jain, N., Shao, Z., Chakka, P., Zhang, N., Antony, S., Liu, H. & Murthy, R. (2010), Hive - a petabyte scale data warehouse using hadoop, *in* '2010 IEEE 26th International Conference on Data Engineering (ICDE 2010)', pp. 996–1005.

Windows, M. (2010), Sql server memory configuration.
URL: *https://learn.microsoft.com/en-us/sql/database-engine/configure-windows/server-memory-server-configuration-options?view=sql-server-ver16*

Wu, W., Chi, Y., Zhu, S., Tatemura, J., Hacigümüs, H. & Naughton, J. F. (2013), Predicting query execution time: Are optimizer cost models really unusable?, *in* '2013 IEEE 29th International Conference on Data Engineering (ICDE)', pp. 1081–1092.

Zhi Kang, J. K., Gaurav, Tan, S. Y., Cheng, F., Sun, S. & He, B. (2021), Efficient deep learning pipelines for accurate cost estimations over large scale query workload, *in* 'Proceedings of the 2021 International Conference on Management of Data', SIGMOD '21, Association for Computing Machinery, New York, NY, USA, p. 1014–1022.
URL: *https://doi.org/10.1145/3448016.3457546*