International Journal of Informatics and Information Systems ISSN 2579-7069
Vol. 5, No. 2, March 2022, pp. 69-75

Data Mining Integration with PostgreSQL Extension by K-Means, ID3 and 1R Method

Tri Wahyuningsih [1,*], I Ketut Gunawan [2], Abdullah Dwi Srenggini [3], Henry Riyandi [4]

Magister Informatics Program, Universitas Raharja, Indonesia
[1] tri.wahyuningsih@raharja.info [*]; [2] iketut@raharja.info; [3] abdullah.d@raharja.info; [4] henryriyandi@raharja.info
* corresponding author

(Received: December 6, 2021 Revised: January 25, 2022 Accepted: February 15, 2022, Available online: March 22, 2022)

Abstract

Data mining is a tool that allows users to quickly access large amounts of data. The purpose of this study was to analyze the integration of data mining technique algorithms into the PostgreSQL database management system. The method used in this research is K-Means, ID3 and 1R, the tools used to implement data mining using RapidMiner and PostgreSQL tools. In this study, the number of rows to be analyzed is 100,000 records, 500,000 records, and 1,000,000 records. The results obtained are the algorithm implemented to validate the data by using an experimental design that serves to observe the time that the analysis of the algorithm that has been integrated into the DBMS is smaller than the results from Rapidminer. As the number of records increases, data analysis becomes difficult using RapidMiner.

Keywords: Data Mining Techniques; Database Management System; Partition; Response Time

1. Introduction

Current technological developments are influenced by a collection of very large, complex, and unstructured data sets that make it difficult to handle if only using ordinary database management tools or traditional data processing applications. Big Data is a large and infinite collection of data with various types and sources of data, so to manage that data, a method and tools are needed that work accordingly [1]. Currently, performance is something that is absolutely necessary in information technology, given the increasing need for accuracy and speed of information delivery. One of the assessments in the performance of a database is the response time [2]. Big data is an innovation that is very useful to be able to provide developments in various fields and communities. To be able to display data quickly, a way to display the data is needed by using a query model that is faster and more efficient in order to simplify and speed up human work in terms of displaying the required data.

There are many database management system (DBMS) tools that are used to implement big data analysis techniques. However, most of these tools have paid copyrights, and for corporations or organizations the price is very expensive [3]. Tools such as WEKA and YALE RapidMiner are open source under the GNU General Public License (GPL), with the GPL we can use, copy, distribute and even modify them. When analyzing large amounts of data we are faced with the problem that the process becomes very complicated and slow.

To overcome this problem, several companies such as Microsoft and Oracle have developed modules in their database management systems that include data mining techniques, these techniques are used to speed up response time because there is no need to change data to perform analysis. This way, there is no need to train staff in using other data analytics tools that provide data analysts direct but controlled access, thereby accelerating productivity while maintaining data security.

Data mining techniques are used to examine large databases as a way to find new and useful patterns [4]. For example, searching for individual records using a database management system or searching a particular web page

through a query to all search engines, searching for information that is closely related to information retrieval. Data mining techniques can be used to improve the capability of information retrieval systems. Data mining is a tool that allows users to quickly access large amounts of data.

Today many companies are turning to open source software to ensure economic benefits[5]. One of them is switching to PostgreSQL database technology because the PostgreSQL database is the most advanced open source database management system in the world. Although developed as open source, PostgreSQL supports most SQL transactions and offers modern commercial database features including complex queries, foreign keys (FK), triggers, ready-to-update views, transactional integrity, concurrency control over various versions of other programming languages. Postgresql also has other additional features, namely there are data types outside the SQL standard, functions, operators, aggregate functions, indexes and procedural programming languages [6], [7]. PostgreSQL is developing as an open source alternative that is increasingly in demand because it has very high performance. PostgreSQL is also a database system that is reliable in managing large data to be accessed by many users. However, this system has not integrated data mining techniques [8]. Therefore, it is necessary to have an independent PostgreSQL database management system to analyze data using data mining techniques.

2. Literature Review

Various studies on the comparison of response times have been studied before. Journals and research that discusses the similarity of theories and research subjects are used as references in this study. The following are previous studies that discuss response time:

Research conducted by Amelec Viloria, Genesis Camargo Acuña, Daniel Jesús Alcázar Franco, Hugo Hernández-Palma, Jorge Pacheco Fuentes, Etelberto Pallares Rambal regarding the integration of data mining techniques using IR, PRISM and ID3 algorithms into a postgreSQL DBMS by analyzing the number of records 100002 , 5000010 and 1000020 the results show that the response time is faster by using postgreSQL.

Research conducted by Rohmat Gunawan entitled "Measurement of Query Response Time in Stored Document-Based NoSQL Databases" which has been published in the Siliwangi Journal Vol.4, Number 2 of 2018. This research contains the measurement of response time of queries performed on NoSQL database-based stored documents. By comparing the number of records 100, 200, 400, 800, 1600, 3200, 6400, and 12,800. The experimental results in this study indicate that the read data query in the nosql database has the fastest response times compared to queries for the create, update and delete processes.

Research conducted by Ragil Martha, Yanuar Firdaus, and Kusuma Ayu Laksitowening entitled "Comparative Analysis of Response Time and Throughput on XML and DBMS as Data Storage. By comparing the number of records 5, 10, 20, 50, 100 and 125. The results obtained are that the response time and throughput of DBMS as a data storage medium is better than XML, because DBMS already has better architecture and indexing than XML.

Research conducted by Noviyanti P, Agatha Deolika, Siti Hartinah, Celine Aloyshima Haris, Tutik Maryana and Nindy Devita Sari entitled "Comparison of Query Response Time on Query View and Cross Product Models" which has been published in the Journal of Information Systems and Information Technology vol. 7 Number 2 of 2018. By comparing the number of records of 100, 500 and 1000 the results obtained are that the query view is faster and more efficient with a comparison that is not too significant.

3. Methodology

The method used in this study is the K-Means, 1R and ID3 algorithm. Tools used to implement data mining using RapidMiner and PostgreSQL. In this study, the number of rows to be analyzed is 100,000 records, 500,000 records, and 1,000,000 records. The number of records, tools and partitioning used to implement data mining is defined as a separate variable. Response time and algorithm results are identified as dependent variables. For more details can be seen in table 1 and table 2.

Table. 1. Independent variables

Variable	Variable Type	Operational	Category
Total Record	Independent	Number of row in to be analyzed	-100000 -500000 -1000000
Tool	Independent	Tool used to apply data mining	Rapidminer. Postgresql
Partitioning	Independent	Table in Postgresql Dataset in Orange / Rapidminer	Partition Non-Partition

Table. 2. Dependent variable

Operationalization of dependent Variable	
Variable	Unit of Measurement
Response Time	Time Interval (Second)
Algorithm Result	Degree of Agreement (Yes / No)

Based on table 2, the response time variable is measured in seconds, while the results of the relationship between the number of records and the results for the K-Means,1R and ID3 algorithms are initialized with "yes" or "no".

3.1. Data Mining Extension

Data Mining Extension is a query language used for data mining. In making data mining extensions in PostgreSQL we use PL/Python (Python procedural language). PL/Python is an extension where we can write python code and run it on PostgreSQL. Figure 1 shows the steps in creating an extension:

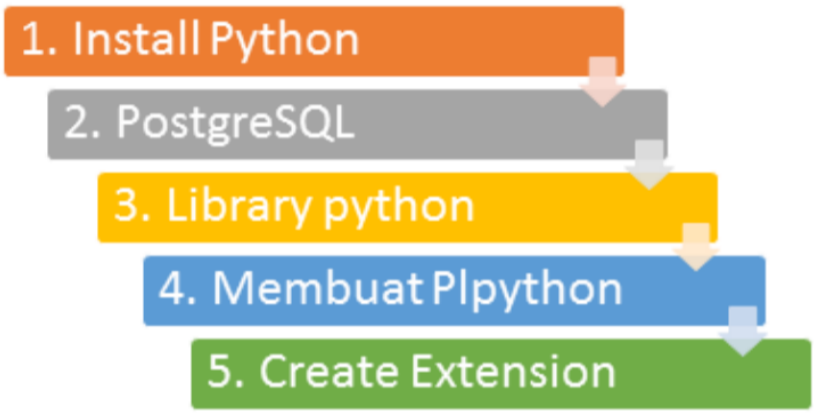

Figure. 1. The steps in making an extension

Figure 1 describes the steps in creating an extension, which consists of:

1) Install python, the python used is python 2 or python 3.
2) PostgreSQL, the minimum is Postgres 9.0 and above.

3) Python libraries: Python is a programming language that has many libraries. Libraries are used to assist in processing or working on tasks. libraries that are widely used to process data in python include Pandas, Sklearn, Scikit-Learn and others. Pandas is used to process data such as join, distinct, group by, aggregation. Sklearn is a free Python library that deals closely with machine learning procedures. Sklearn consists of algorithms such as support vector engine, gradient boosting (machine learning technique for regression and classification problems), k-means (clustering algorithm), random forest (algorithm for data classification), and DBSCAN. Scikit-Learn provides a number of features for data science purposes such as regression algorithms, naive bayes algorithms, clustering algorithms, decision tree algorithms and others.
4) Make Plpython, Make Plpython according to the algorithm function created.
5) Create extension
6) Python k-means pl script image

```
CREATE OR replace FUNCTION kmeansplpy(tabelinput text, kolom text[], kluster_jml int) RETURNS bytea AS
$$
from pandas import DataFrame
from sklearn.cluster import KMeans
from cPickle import dumps

semua_kolom = ",".join(kolom)
if semua_kolom == "":
    semua_kolom = "*"

rv = plpy.execute('SELECT %s FROM %s;' % (semua_kolom, plpy.quote_ident(tabelinput)))
frame = []

for i in rv:
    frame.append(i)

df = DataFrame(frame).convert_objects(convert_numeric =True)
kmeans = KMeans(n_clusters=kluster_jml, random_state=0).fit(df._get_numeric_data())
return dumps(kmeans)

$$ LANGUAGE plpython3u;
```

Figure. 2. K-Means main function script

```
CREATE OR replace FUNCTION GetKmeansCentroid(tabelm text, kolom text, idmodel int) RETURNS real[] AS

$$

from pandas import DataFrame
from cPickle import loads

rv = plpy.execute('SELECT %s FROM %s WHERE id = %s;' % (plpy.quote_ident(kolom), plpy.quote_ident(tabelm), idmodel))
model = loads(rv[0][model_column])
ret = map(list, model.cluster_centers_)
return ret

$$ LANGUAGE plpythonu;
```

Figure. 3. The script takes the center of the centroid

3.2. Partition

One option that PostgreSQL offers to improve performance in this case is table partitioning, which allows better performance when querying tables [9]. In the database there are two types of tables that can be used, namely partition tables and non-partitioned or non-partitioned tables. Partitioning philosophy is to break the table into several segments (partitions or sub partitions), where conventional tables only have one segment. This technique reduces the number of physical reads on the database when the query is run. In PostgreSQL, the existing partition types are range and list [10]. A partition by range is a partition created using a defined range based on any non-overlapping column between the ranges of values assigned to different child tables. Partition by list is a partition created by value. If the

table is partitioned using a range, the query will scan specifically for the segment where the data is located, not all data records are scanned, so that the query process is faster [11].

4. Result and Discussion

4.1. The ratio of the number of record variables to the result time

In the first case, we tested how response times run in the Rapidminer and PostgreSQL tools when controlling the Number of Records variable for each calculation considered. Table 3 shows that using the K-Means algorithm, the comparison of Rapidminer and PostgreSQL tools for 100,000 records is 15.3 seconds for Rapidminer and 2.4 for PostgreSQL. For the number of records 500,000 shows a comparison of 20.61 seconds using the Rapidminer tool and 10.2 for PostgreSQL. As for the number of records, 1,000,000 shows an error for Rapidminer tools and for PostgreSQL it has a response time of 19.4 seconds. This means that it shows that as the number of records increases, the analysis time for the K-Means calculation increases. By using Rapidminer response times are higher than investigations made using the calculations built into the PostgreSQL database. For category or level 1000,000 records, the check should not be done with Rapidminer because it returns an error due to the amount of data that is too large. Table 4 using the ID3 algorithm shows the same results as the K-Means calculation, where as the number of records increases, the response time also increases. Likewise for a variable with a record number of 1,000,000 also an error occurs. Table 5 clarifies the appropriate direct relationship between the number of records.

Table. 3. Results of record variable manipulation for K-Means

Result of manipulating the variable number of record for the KMeans algorithm		
Total Record	Rapidminer	PostgreSQL
-100000	15.3	2.4
-500000	20.61	10.2
-1000000	…	19.4

Table. 4. Results of record variable manipulation for ID3

Result of manipulating the variable number of record for the ID3 algorithm		
Total Record	Rapidminer	PostgreSQL
-100000	7.23	1.6
-500000	22.4	7.5
-1000000	…	16.22

Table. 5. Results of record variable manipulation for 1R

Result of manipulating the variable number of record for the 1R algorithm		
Total Record	Rapidminer	PostgreSQL
-100000	11.3	1.4
-500000	17.61	8.2

-1000000	…	17.4

4.2. The ratio of the variable number of records to the response of the algorithm

In case number 2, the behavior of the variables resulting from the algorithm is analyzed by manipulating the number of records to be analyzed. By analyzing the results of Tables 6, 7, and 8, it can be concluded that with the increase in the number of records, data analysis becomes difficult with rapidminer.

Table. 6. The results of the relationship between the number of records and KMeans

Result of the relationship between the number of record and the result for the KMeans algorithm		
Total Record	Rapidminer	PostgreSQL
100000	Yes	Yes
500000	Yes	Yes
1000000	No	Yes

Table. 7. The results of the relationship between the number of records and ID3

Result of the relationship between the number of record and the result for the ID3 algorithm		
Total Record	Rapidminer	PostgreSQL
100000	Yes	Yes
500000	Yes	Yes
1000000	No	Yes

Table. 8. The results of the relationship between the number of records and 1R

Result of the relationship between the number of record and the result for the 1R algorithm		
Total Record	Rapidminer	PostgreSQL
100000	Yes	Yes
500000	Yes	Yes
1000000	No	Yes

5. Conclusion

By using this technique there is a function that can be developed to take advantage of one of the manager's optimization mechanisms to improve the response results of the K-Means, ID3 and 1R algorithms. The algorithm that is implemented to validate the data is using an experimental design that serves to observe that the analysis time of the algorithm that has been integrated into the DBMS shows a smaller response time than the results using RapidMiner. The increasing number of records shows that data analysis using rapidminer tools becomes difficult. The suggestions for further development are: Data cleaning is expected to be integrated in the database. There is a graph to present the

results of the analysis. Data mining techniques to improve the response results from the algorithm are not only with K-Means, ID3, and 1R but can be with other methods.

References

[1] Maryanto, B. (2017). Big Data dan Pemanfaatannya dalam Berbagai Sektor. Media Informatika, Vol. 16 No. 2.

[2] Krishna, S. 1992, "Introduction to database and knowledge-base systems", Jakarta, World Scientific

[3] Viloria A., Lis-Gutiérrez JP., Gaitán-Angulo M., Godoy A.R.M., Moreno G.C., Kamatkar S.J. (2018) Methodology for the Design of a Student Pattern Recognition Tool to Facilitate the Teaching - Learning Process Through Knowledge Data Discovery (Big Data). In: Tan Y., Shi Y., Tang Q. (eds) Data Mining and Big Data. DMBD 2018. Lecture Notes in Computer Science, vol 10943. Springer, Cham.

[4] Vasquez, C., Torres, M., Viloria, A.: Public policies in science and technology in Latin American countries with universities in the top 100 of web ranking. J. Eng. Appl. Sci. 12(11), 2963–2965 (2017).

[5] Boehm, B., Abts, C. y Chulani, S., "Software development cost estimation approaches-a survey", Annals of Software Engineering 10, 2000, pp. 177-205

[6] Wieczorek, I. y Briand, L., Resource estimation in software engineering, Technical Report, International Software Engineering Research Network, 2001.

[7] Piotrowski, A.P., 2017. Review of Differential Evolution population size. Swarm Evol. Comput. 32, 1–24. https://doi.org/10.1016/j.swevo.2016.05.003

[8] Kaya, I., 2009. A genetic algorithm approach to determine the sample size for attribute control charts. Inf. Sci. (Ny). 179, 1552– 1566. https://doi.org/10.1016/j.ins.2008.09.024

[9] Chauhan, Sonam S, Deskmukh P R., Literature Review on Information Extraction by Partitioning. International Journal of Computer Science and Mobile Computing. Vol 2. 2013.

[10] Gaitán-Angulo M, Jairo Enrique Santander Abril, Amelec Viloria, Julio Mojica Herazo, Pedro Hernández Malpica, Jairo Luis Martínez Ventura, Lissette Hernández-Fernández. (2018) Company Family, Innovation and Colombian Graphic Industry: A Bayesian Estimation of a Logistical Model. In: Tan Y., Shi Y., Tang Q. (eds) Data Mining and Big Data. DMBD 2018. Lecture Notes in Computer Science, vol 10943. Springer, Cham.

[11] Lubis, J. H. (2017). Analisa Performansi Query pada Database Smell. Jurnal Mantik Penusa, ISSN:2088-3943.