American Journal of Sciences and Engineering Research www.iarjournals.com

American Journal of Sciences and Engineering Research
E-ISSN -2348 – 703X, Volume 4, Issue 1, 2021

Application of Decision Tree to Finding Table-Scan Stored Procedures in Db2 for z/OS

MohammadJavad Haji Boluri[1],Saeed Seyed Agha Banihashemi[2]
[1,2]*School of international relations.*

ABSTRACT:*Nowadays, many banking businesses around the globe are using Db2 for z/OS database system in order to providethe highest quality service for their customers. Many of these banking systems are using stored procedures to provide different services to their customers, and most times, the number of these stored procedures is very high. While businesses are expanding and developing more stored procedures, optimizing the currently stored procedures is a critical activity for DBAs man. However, analyzing and finding crude stored procedures, especially the ones that cause a table scan on the database system, is a frustrating and time-consuming job, especially if there are store procedures available in the system. For example, analyzing nine thousand stored procedures and finding the ones thatare scanning the table for fetching data is impossible. As a result, an evaluation system is needed to analyze stored procedures. In this paper, an evaluation stored procedure system is developedbased on the decision tree. By using this system, finding table scan stored procedures will become easier for DBAs man.*

Keywords:*Database, Db2, Machine learning, Decision Tree, Stored Procedure*

I. INTRODUCTION

Recently, Machine Learning (ML) algorithms have been used to tackle fundamental problems in many industries and companies. It helps them to reduce their expenditures and save a lot of time. For example, a clustering-based decision tree was applied by Adriano Lino[1] to found and classified student SQL query errors in the database. Also, for evaluating students' knowledge that applies to higher education, X. Wang[2] proposed an evaluation model based on a decision tree algorithm to find the best students among a lot of application forms. Moreover, Network anomaly detection was proposed by Muniyandi[3] to distinguish normal and anomaly activities with the use of decision tree algorithms. These papers use a decision tree for their classifier to provide a better system for their industries. In this paper, we propose a stored procedure evaluation system that can classify table scans and normal stored procedures in the Db2 database system.

Nowadays, most banking companies around the world, using IBM DB2 database systems for managing their customer's data. DB2 database system is one of the best relational databases in the market and uses different facilities and tools to manage databases. It supports transactions via Web servers, Customer Information Control System (CICS), and distributed data facility (DDF) from remote clients on numerous platforms. The distributed data facility (DDF) is what enables client applications to access DB2 data or call Stored Procedure. A stored procedure is a compiled program that can execute SQL statements and is stored at a local or remote DB2 server. You can invoke a stored procedure from an application program. However,monitoring stored procedures to find a crude one is very important for the banking system to provide the best services to their clients. But in large-scale systems that there are from five thousand to ten thousand stored procedures, we need some tools to provide a facility for distinguishing unwell stored procedures (SP) from the others. The DB2 Performance Monitor (DB2PM) product provides accounting reports from the DB2 subsystem activities that provide information about system activities such as Stored Procedures. But interpreting these reports is a very time-consuming and frustrating job when there is a bunch of data to analyze. To tackle this problem, we use a Decision Tree algorithm to evaluate stored procedures.

Decision tree learning is a method for approximating discrete-valued target functions, in which the learned

 Received- 20-12-2020, Accepted- 02-01-2021

function is represented by a decision tree. Learned trees can also be re-represented sets of if-then rules to improve human readability. These learning methods are among the most popular of inductive inference algorithms and have been successfully applied to a broad range of tasks from learning to diagnose medical cases to learn to assess the credit risk of loan applicants.

In this paper, a stored procedure evaluation system is built by the Decision Tree learning algorithm. By using this algorithm, we could save a lot of time and analysis stored procedures quickly to find a crude one.

II. The Construction of Stored Procedure Evaluation Model

2.1. Data preparation and selection of features

For producing data about stored procedures, the DB2PM program is used. This program can produce accounting reports base on System Management Facility (SMF) data which is provided statistics about Stored procedures. After the report has been produced, data is gathered by Rexx program and inserted into the optimization tables. The reports contain 4 major parts. The first part indicates statistics data about important stored procedure parameters. For example, how many times SP executes and how much CPU is consumed during its operation. The second part provides information about how many data manipulation language (DML) operationswere executed. The third and fourth parts provide information about the buffer pool and locking information respectively. From each part, the important parameter is chosen and finally,a dataset based on imperative parameters is created.

For creating dataset 150 stored procedures, which the main function of them are select operation, are chosen base on average CPU elapsed time and average CPU consumption. The procedures are stored in three groups. The first group hasthe highest average of CPU elapsed time and consumption. The second group has moderate usage of CPU and the third group has the least CPU consumption. The main reason for this selection is Table Scan. Because the select statement has more I/O operation than other DML predicate and causes much more CPU consumption. However, it can be seen that sometimes some stored procedures have little CPU consumption but do a table scan operation in the database. After creating the groups, for creating dataset a CP_CPU_SU, aLOCK_LATCH, a SUSPEND, a LOCK_REQUEST_AVG, a GETPAGES_AVG, and aSELECT_AVG parameters are chosen.The CP_CPU_SU is the most important feature that indicates how much the stored procedure consumes CPU for its execution.The LOCK_LATCH field provides information about SP wait times to use resources. In a database system, if one program uses resources to manipulate them, the other program should wait until its operations over. The SUSPEND provides information about the number of times SP wait for using resources. The LOCK_REQUEST_AVG indicates how much locks are requested to use resources.The GETPAGES_AVG and SELECT_AVG indicate how many get pages have occurred and how many select statements are executed respectively. These fields are very important to distinguish store procedures' functionality from each other. Finally, to find out which of the scanned resource table had caused a problem for the system,each SP is analyzed by Visual Explain tools -an IBM tool for analyzing stored procedures. The result of the assessment is recorded in the SP_OK field for each sp. If the result is zero, it indicates that the SP scanned the table and if the result is one, it is indicated that the SP has not scanned the table. An example of a prepared dataset can be seen below.

Table 1. An example of a table

SPNAME	CP_CPU_SU	LOCK_LATCH	SUSPEND	LOCK_REQUEST_AVG	GETPAGES_AVG	SELECT_AVG	SP_OK
SP01	500277	0.013441	2.885383	12	2206855	7	0
SP02	154597.53	0.000925	42.152373	3	368793	1	0
SP03	55793	0.002449	7.877189	1	433864	1	0
SP04	45832.57	0.042436	0.729102	4	424430	4	0
SP05	40336	0.000126	0.22903	22	25830	1	0
SP06	32524.5	0.000831	7.40549	9	206784	8	0

SP07	32106.68	0.000067	0.000231	3	201938	1	0
SP08	31701.24	0.000245	0.000248	3	11608	1	1
SP09	31350.2	0.001077	4.975289	1	270178	2	1
SP10	31013.09	0.000247	2.727196	17	187803	1	0
SP11	21035.04	0.000261	0.002568	12	52906	4	0
SP12	20035.68	0.000922	0.00519	13	33637	9	0
SP13	19017	0.004917	11.738129	5	79524	2	1

2.2. Decision Tree Model

For creating The Decision Tree model,CP_CPU_SU,LOCK_LATCH, SUSPEND, LOCK_REQUEST_AVG, GETPAGES_AVG, and SELECT_AVG are chosen as features and SP_OK is chosen as the target value. The dataset is then split into training and testing data.

After choosing the features and target data, the decision tree algorithm can be applied to it. Firstly, a dataset is prepared as a CSV file in which data are separated with a comma and the data frame method is used in the PANDAS package to handle the table structure as frames. Then, data is checked to prevent probable issues by ensuring that there are no null and noisy data. After the preprocessing stage, all the necessary required package libraries which are compatible with python are installed. Next, in the computational stage, a tree model is created base on the train data set. Finally, for evaluating the model, the test data set is evaluated by the created model.

Overall, 112 stored procedures are used for the training set out of which 18 are table scans and 94 have no problem. Because the GETPAGE_AVG has 0.27 Gini, the decision tree algorithm has chosenit as the root of the tree. SELECT_AVG and LOCK_LATCH are the first deep of the tree with 0.043 and 0.397 Gini respectively. The Decision tree model is shown in figure 1.

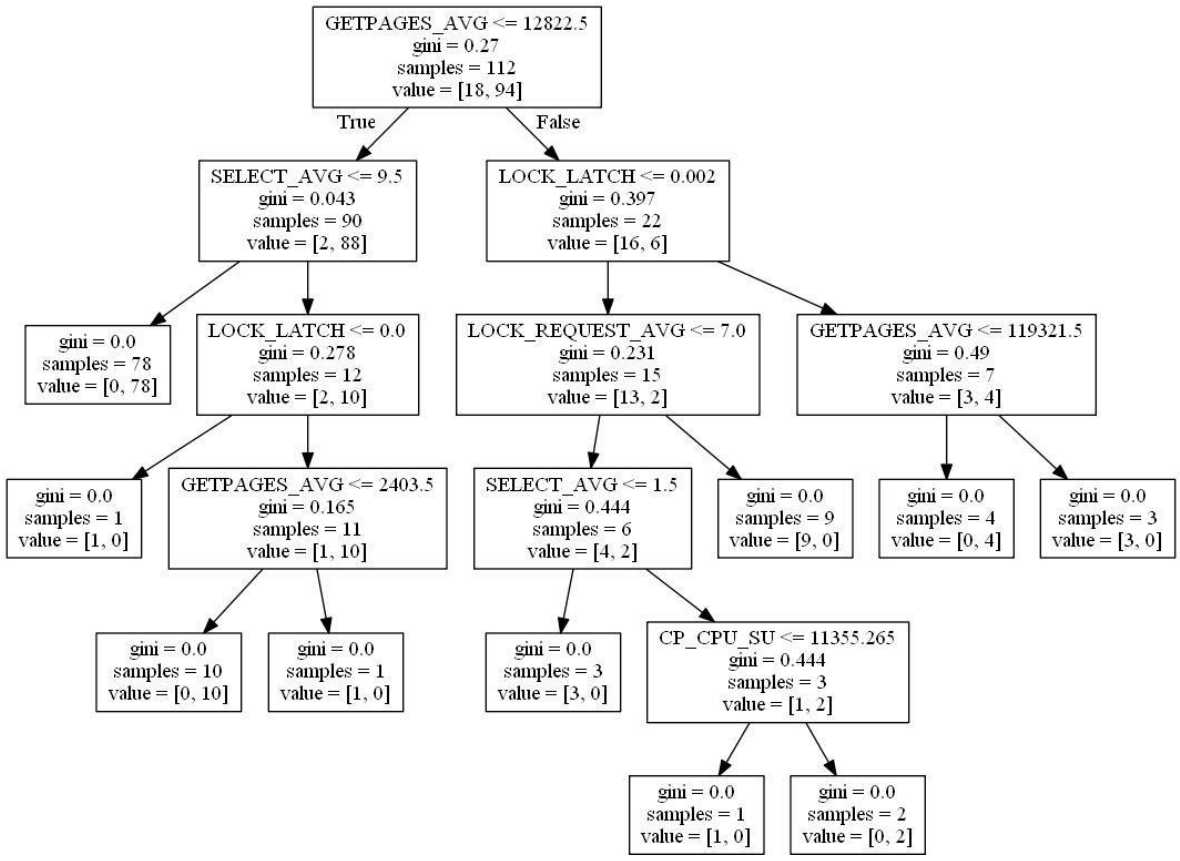

Fig. 1. Decision Tree Model

III. Analysis and Realization of the Model

After creating the model, the test dataset was entered into the model and the result is shown in table 2.

Table 2. Testing Result

Interval	Testing set	Percentage
Correct	37	0.97%
Wrong	1	0.03%
Total	38	

It can be seen from the resultsthat the model can give quite accurate results compared to the stored procedure evaluation system. To test the model efficiency, the decision tree model was applied to five thousand stored procedures and it was found thatthe modelwas practical and evaluated the table scan procedure very well.

The evaluation system is designed to evaluate table scans stored procedure by using a decision tree model. For this purpose, a program was developed which gets a stored procedure report as a CSV file, and after processing, creates a CSV file of table scan stored procedure. A screenshot of the application figure can be seen in figure 2.

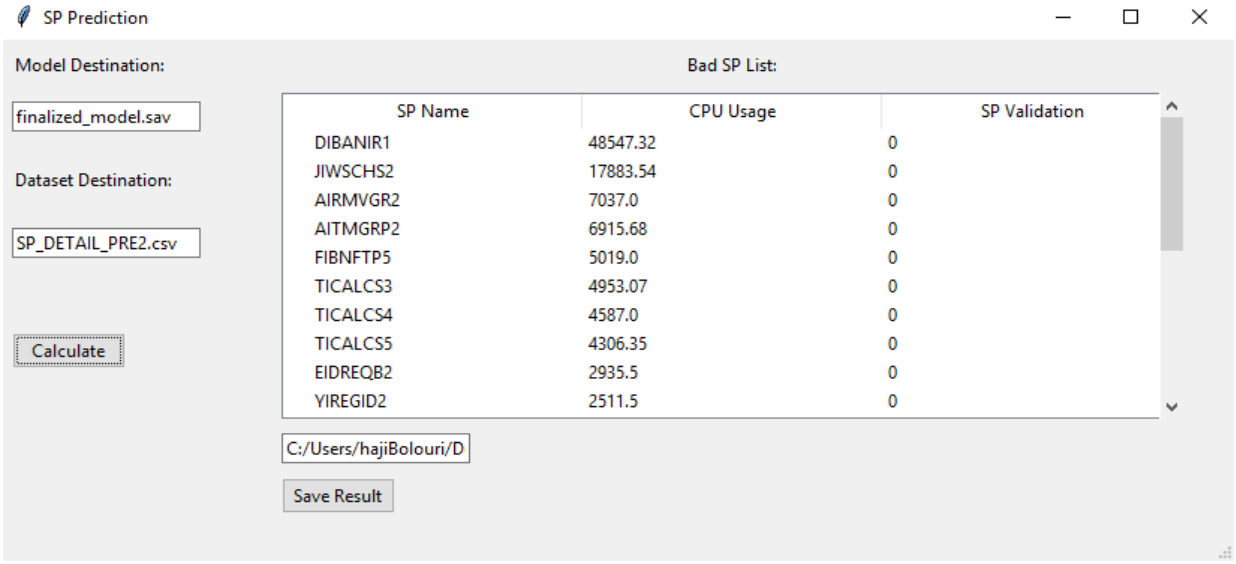

Fig. 2. Application Figure

IV. CONCLUSIONS

This paper proposes the application of the decision tree for evaluating table scan stored procedures in the DB2 database system. DB2PM accounting report was used to produce a dataset from the available system and the decision tree algorithm was used to create a model and evaluate table scan stored procedures. As a result, it can be seen that get pages are the most important parameter to indicate whethera table scan was used or not. Moreover, locking request is another major feature that can be used to find table scan SPs. Furthermore, it was shown thatusing machine learning algorithmscan help DBAs evaluate their system functionality very easily and save much time and effort. In conclusion, database systems can use machine learning algorithms to solve optimization problems and analyze issues with ease.

V. Future work

The evaluation stored procedure system can be enhanced for further analysis by applying different machine learning algorithms to the prepared dataset. Additionally, the provided model can be improved by choosing better features based on different DB2PM reports.

VI. REFERENCES

1. A. Lino, Á. Rocha, L. Macedo and A. Sizo, "Application of Clustering-Based Decision Tree Approach in SQL Query Error Database," Future Generation Computer Systems, vol. Volume 93, pp. Pages 392-406, April 2019.
2. X. Wang, C. Zhou and X. Xu, "Application of C4.5 decision tree for scholarship evaluations," Procedia Computer Science, vol. Volume 151, p. 179–184, 2019.
3. A. P. Muniyandi, R.Rajeswari and R.Rajaram, "Network Anomaly Detection by Cascading K-Means Clustering and C4.5 Decision Tree algorithm," Procedia Engineering, vol. R.Rajaram, p. R.Rajaram, 2012.