ATLANTIS PRESS

Advances in Economics, Business and Management Research, volume 30
International Conference on Engineering Management (Iconf-EM 2016)

Research and Design of a Rules Engine for Bank Anti-fraud Platform

Xiaoguo Wang*, Lin Zhang and Yuxiang Liu

School of Electronics and Information Engineering, Tongji University, Shanghai, China

*Corresponding author: xiaoguowang@tongji.edu.cn

Keywords: anti-fraud, rules engine, rules database, rules matching

Abstract. In view of the problem that the efficiency of processing data is not high and the rules database is more complicated with large amounts of users' historical transaction data in bank anti-fraud platform, based on the banking business and technical requirements, this paper designs a simpler and more efficient rules engine for bank anti-fraud platform. This rules engine has functions of service interface, rules database, rules management and rules matching, etc. With large amounts of data in platform, this rules engine can match rules and perform the actions specified in rules to provide technical support for anti-fraud detection.

1 Introduction

As an important part of the bank anti-fraud platform, rules engine can apply rules for large amounts of data to provide anti-fraud detection of users' transaction data and perform the actions specified in the rules by matching rules.

With the rapid development of the Internet finance, fraud of debit cards, credit cards and online banking has become the focus of monitoring the risk in the bank [1]. Existing bank anti-fraud platform has some problems that processing large amounts of data is not efficient and the rules database is more complicated [2]. Therefore, improving the rules engine in business and technology, researching and designing a simpler and more efficient rules engine are urgent demands to address the problems.

2 Function requirements

In this paper, the functions of the designed rules engine include transaction data acquisition interface, detection data storage interface, services interface about rules engine, rules management, rules database and rules matching. The frame of the rules engine is shown in Figure 1.

(1) Data interface

The users' historical transaction data in bank anti-fraud platform include debit card transaction data, credit card transaction data, online banking transaction data and other data. These data come from different sources and the storage format is not same with each other. Through the data acquisition interface, it can standardize data format for the rules engine by providing recognizable and standardized data. And the detection results of users' transaction data are stored in the database by data storage interface.

(2) Service interface

By the service interface, bank anti-fraud platform calls the rules engine to provide logic control and detect users' historical transaction data.

(3) Rules management

Depending on the different strategies, it has different operations of initializing, loading, querying, modifying, deleting, and verifying for rules.

(4) Rules database

Rules database is the basis of the rules engine and it is a collection of rules files which are established by rules engine. Rules database is maintained by rules management and it is used by rules engine.

(5) Rules Matching

The first step is modelling with rules files in rules database. Then, it will match rules with users' historical transaction data by the model. Finally, it will return the detection results for bank anti-fraud platform.

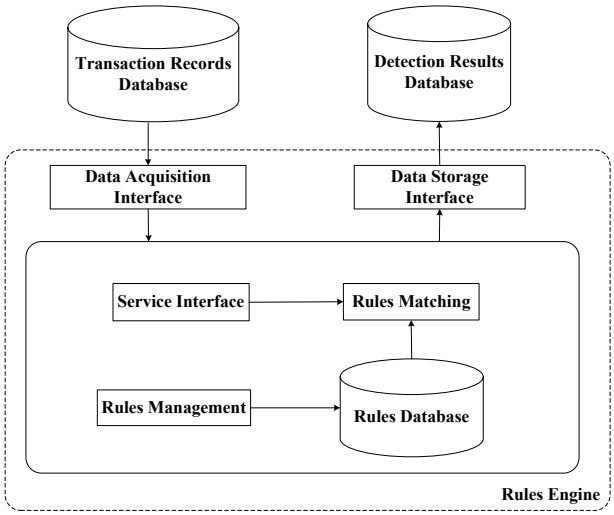

Figure 1. Frame of the rules engine

3 Design and implementation

3.1 Overall architecture

The overall architecture of the rules engine is shown in Figure 2.

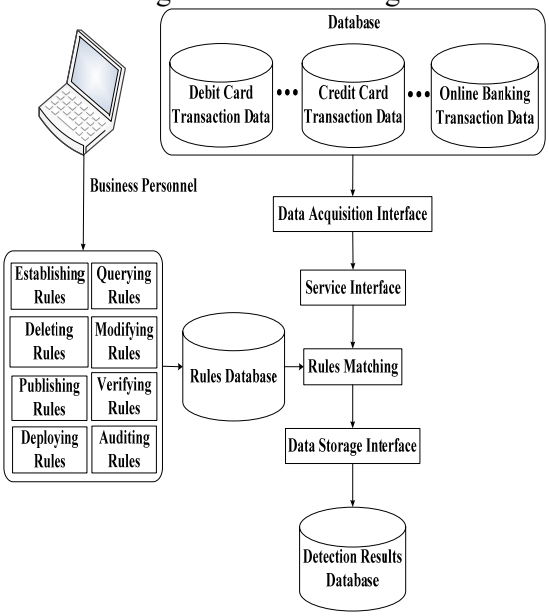

Figure 2. Rules engine overall architecture

(1) Service interface module

Service interface module using Web Service provides service access of rules engine for bank anti-fraud platform. The service interface exchanges information by XML to have business scalability.

XML provides a way to describe structured data and it has a uniform format and syntax. XML is also a scalable language. According to the basic syntax of XML, it can further define the scope and use of document formats [3].

(2) Rules management module

Rules management module includes the establishing rules, querying rules, modifying rules, deleting rules, verifying rules, auditing rules, deploying rules and publishing rules, as shown in Figure 3.

According to the demands, rules management module establishes new rules files or modifies existing rules files in rules engine. And the rules testers verify these new or modified rules. If rules don't pass verification, they will be sent back to be remodified. On the contrary, if they pass verification, they will be deployed and published. And the published rules will be called by the rules engine.

(3) Rules database module

Rules database module is a collection of rules files using HBase technology for storage. According to the characteristics of fraud, it will refine elements of fraud behaviour, and integrate these elements logically to form a rules file. When users' behaviour triggers one rule or more, bank anti-fraud platform will take some intervention for users to avoid risk.

HBase is a distributed and scalable big data warehouse using HDFS distributed processing mode, with key/value storage mode bringing real-time query capabilities as well as off-line processing or batch processing capability by MapReduce. Through HBase technology, the rules database module stores data into database to improve the efficiency of querying relevant rules in rules database for the rules engine [4].

(4) Rules matching module

Through the rules engine service interface, it will call the rules engine. Users' historical transaction data from bank anti-fraud platform will be matched by the established model and subgraph matching algorithm with the Spark platform. Then it will return the detection results.

Spark Streaming is built on a real-time stream computing framework that extends Spark Streaming large data processing capabilities. Spark Streaming uses transaction data streaming to divide into RDD with time-unit. It uses RDD to operate each block of data. Each block of data will have a Spark job process. And finally, by processing in batch it handles each time slice of data. The rules engine uses Spark Streaming technology in the process of matching users' transaction data to improve the data processing and matching speed in big data [5].

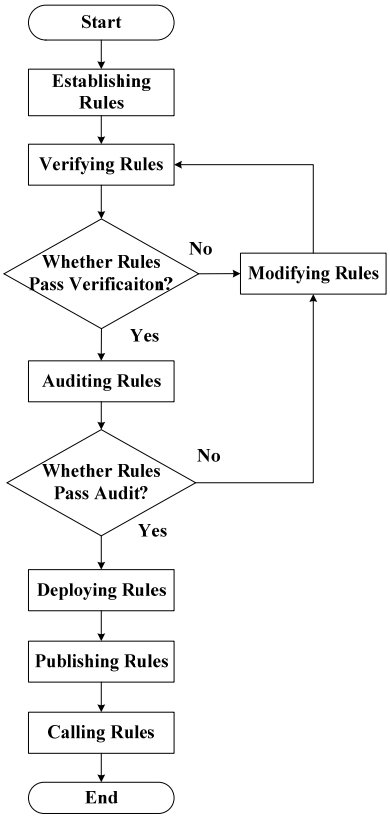

Figure 3. Rules management flow

3.2 Application architecture

For bank anti-fraud platform, the application architecture of the rules engine is divided into the following four levels, as shown in Figure 4.

(1) Data Layer

Data layer provides a secure and concurrent data mechanism to store anti-fraud data, rules database and other data, and simplify the functions of adding data, deleting data, modifying data, and querying data for the rules engine in bank anti-fraud platform.

(2) Access Layer

By mapping between objects and relational database, access layer reads and transfers relevant data in database of rules engine.

(3) Service Layer

Service layer is to provide users with different service interfaces, and complete the corresponding business functions operation through different service interfaces, such as detecting users' transaction data and calling the rules engine service.

(4) Business Layer

Business layer is a communication bridge between users and platform, which not only provides users with a medium of communication, but also to show and submit data to achieve certain logic to coordinate the operation of users and platform. In bank anti-fraud platform, it is mainly about users having access to web pages by HTTP protocol.

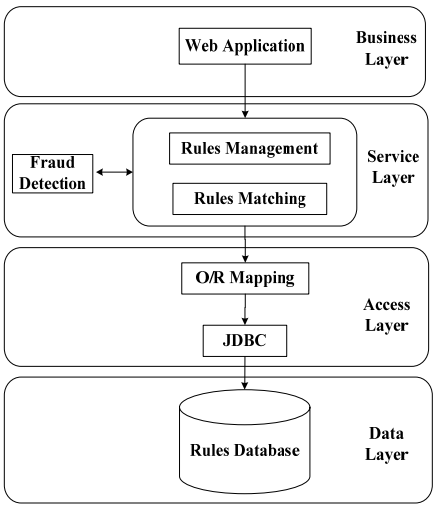

Figure 4. Rules engine application architecture

3.3 Functional flow

Functional flow for rules engine in bank anti-fraud platform is shown in Figure 5.

By data interface service, rules engine acquires users' historical transaction data and then, extracts data, converses data, and checks data quality. The standardized data which need to be detected will eventually be stored using HBase technology.

Rules engine will check the data to be detected whether the data are consistent with the requirements. If they meet requirements, the rules engine will be called. If they don't meet requirements, they will be with exception handling.

When the data to be detected meet the requirements, the rules engine will be called by service interface. The rules engine using Spark Streaming technology to detect data by rules matching. And it will return the results into the database.

When the process of calling rules engine is abnormal, it will generate an exception message and return the exception information into the database.

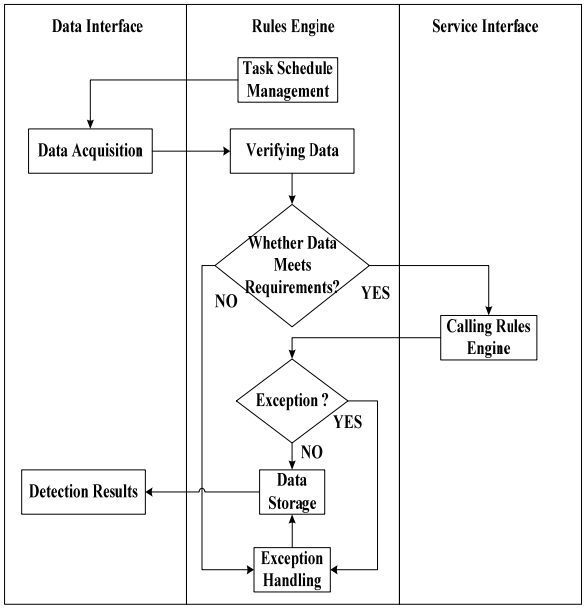

Figure 5. Rules Engine functional flow

3.4 Security mechanism

Security mechanism includes data reading security, data storage security, rules database security and monitoring security [5].

(1) Data reading security

Data reading security uses user authentication in rules engine to ensure that the user has access to read the history of the bank transaction data. And transaction data in the process of reading the data are encrypted to ensure that they remain private, as well as digital signature is to ensure that they have not been tampered.

(2) Data storage security

Data storage security uses recognition of user identity and other technologies to ensure that user has access to the data of the detection results and store data into the database. In the process of data storage, data encryption and signature will be used to ensure that data are private and can't be tampered with in the stored procedure.

(3) Rules database security

Rules engine ensures the safety of the rules database through the role management control and access control mode. Using role management control, it can set up and manage user roles to get access to the system resources, such as whether user has access to rules database. Implementation of access control, it can establish and maintain roles with business-related and permissions correspondence table to assign a role who has access to the resources, such as whether the user has the privilege of the operations of establishing, querying, modifying, deleting, auditing, deploying and publishing the rules files.

(4) Monitoring security

Rules engine needs to have complete running logs, users' monitoring operation logs, users' login logs and system abnormal logs to ensure that it can monitor the processes.

4 Conclusion

This paper researches and designs a rules engine for bank anti-fraud platform. With large amounts of users' historical transaction data, it will call the rules engine to apply rules. And it will match rules to perform the actions specified in rules to provide a simpler and more efficient anti-fraud detection with users' historical transaction data which provides a solution for bank anti-fraud detection.

References

[1] Q.H. Zhang, IEEE Comput. Soc., **2**, 181 (2009)
[2] S. Panigrahi, A. Kundu, S. Sural, A.K. Majumdar, Inf. Fusion, **10**, 354(2009)
[3] M.Y. Maarouf, S. M. Chung, IEEE Comput. Soc., **2**, 361(2008)
[4] Q.X. Li, B.H. Qiang, C.Y. Yang, J. Guangdong Univ. Technol., **3**, 8(2014)
[5] X. Meng, J. Bradley, B. Yuvaz, E. Sparks, S. Venkataraman, D. Liu, J. Freeman, D. Tsai, M. Amde, S. Owen, D. Xin, R. Xin, M. Franklin, R. Zadeh, M. Zaharia, A. Talwalkar, JMLR, **17**, 1(2016)