CREDIT CARD FRAUD DETECTION USING BIG DATA FRAMEWORK

[1]Namrata Pandey, [2]Rajeshwari S, [3]Shobha Rani BN, [4]Mounica B
[1]B.E. Student, [2]B.E. Student, [3]B.E. Student, [4]Sr. Asst Professor
[1]Dept. of Information Science and Engineering
[1]New Horizon College of Engineering, Bangalore, India

Abstract : In this paper, we are implementing a credit card fraud detection system, by using big data technologies. Credit card is one of the most divisive products among all the financial tools available. The usage of credit cards has become common in today's world and huge volume of transaction happens online. The increase in these transactions has also come with many apprehensions on the authenticity of the transactions. In today's world, there have been various phishing attacks over the internet. This needs to be dealt with caution. Be focus on designing an online credit card fraud detection framework with Big data technology. To accomplish that, we propose a workflow which satisfies most design ideas of current credit card fraud detection systems. We implement it with largest Big data technologies like Hadoop, Spark, Apache Kafka etc. A prototype is implemented and tested with a synthetic dataset, which shows great potentials of achieving the above goals.

IndexTerms **– Credit Card Fraud Detection, Big Data, HMM Model**

I. INTRODUCTION

Data is being increased at an exponential rate every second with the rapid development of e-commerce and internet. The banking sector has become a very important sector in our present generation where every human has to deal with bank either online or physically. With this development, online transactions has become one of the most important ways of trading. Credit card is one of the ways used for performing online transaction. A credit card is nothing but a payment card issued to users to enable the cardholder to pay a merchant for the required goods and services.

In dealing with banks and credit cards, the customers face the chances of being trapped by fraudsters. In online transaction usually the credit cards are used as virtual cards. An fraudster only needs to know a few important details of the card like card id, security code etc to make the fraudulent transaction. While the genuine card holder often does not realise that their credit card details has been leaked, which causes a significant financial loss to both the cardholder and the credit card company.In the decades, many researchers have developed many credit card fraud detection systems.

The main challenge in most of the ccfds is improving the detection accuracy and the capacity of computation with the explosive growth of trading data. As the number of users are being increasing day by day , the transactions has bought heavy workloads to these detection systems. The number of transactions per second can reach upto millions and the historic data stored can reach upto PBs or even EBs.

In the recent times, big data and cloud seems to be the key of solving challenge of computational capacity , data storage and security. Considering an example, before the usage of big data , the companies could analyze only 2% of the historical transactions and update the model every 2 or 3 days. Now with the help of big data technologies the models are updated every 1 or 2 hours. Cloud storage is a model that enables us to store digital data in logical pools. This cloud storage makes the data available and accessible for our use. The services provided by the cloud can be accessed through a web service application programming interface or any application that uses API.

In e-commerce credits cards are used more frequently, which makes the transaction and payment easier and creates many small transactions. In this paper, we try to address the challenge through a hybrid framework.

- We propose a credit card fraud detection workflow, which can fuse different detection models to improve accuracy. It contains most of the common design ideas in latest CCFDSs, which make it much easier to integrate detection algorithms into the workflow;
- Credit card transaction can be captured over time based on the behaviour of the cardholder's spending incidents. The events are stored as transaction history which would be used to analyse the possible detection of fraud.
- We implement the framework with the latest big data technologies like spark, kafka.we make use of cloud services for the storage of all the data. With these technologies, we are able to handle the burst amount of data and build a scalable and reliable system. Experimental results show that this system has the potential to achieve a sustainable performance.

II. HMM MODEL

There have been many fraud detection algorithm developed in the past few years. Most algorithms can be divided into supervised and unsupervised algorithm. Supervised algorithm includes neural network, logistic regression model, etc. Compared with

supervised algorithm, the unsupervised approaches are widely used in credit card fraud detection. The unsupervised algorithm recommends the breakpoint analysis to identify the changes in spending behaviour.

We try to fuse existing algorithm, to achieve the goal, instead of developing a new algorithm. Here we are using hidden markov model (HMM) to detect fraud transactions.

Hidden markov model is perfect solution for detecting fraud transaction. The most important benefit of HMM is reduction in the number of false positive alarm i.e., considering a genuine transaction as a fraud transaction. Using HMM the cardholder's spending behaviour can be observed to detect fraud. Every arriving transaction is stored in the fraud detection system. The HMM model stores the transaction datasets in clusters, depending on the type of transaction. These clusters will be hidden from the outside world. Different types of transaction will be stored in different clusters.

This will try to find out the differences in the cardholder historical transactions. It will separate the data based on location, type of transaction and IP/mac address.

III. SYSTEM ARCHITECTURE

SPRING FRAMEWORK

We are using spring framework for the implementation. Spring is the most mainstream application improvement structure for big business Java. A large number of designers around the globe utilize Spring Framework to make high performing, effortlessly testable, and reusable code. Spring system is an open source Java stage. It was at first composed by Rod Johnson and was first discharged under the Apache 2.0 permit in June 2003.Spring is lightweight with regards to size and straightforwardness. The fundamental rendition of Spring structure is around 2MB.

The centre highlights of the Spring Framework can be utilized as a part of building up any Java application, however there are augmentations for building web applications over the Java EE stage. Spring system focuses to make J2EE advancement simpler to utilize and advances great programming hones by empowering a POJO-based programming model.

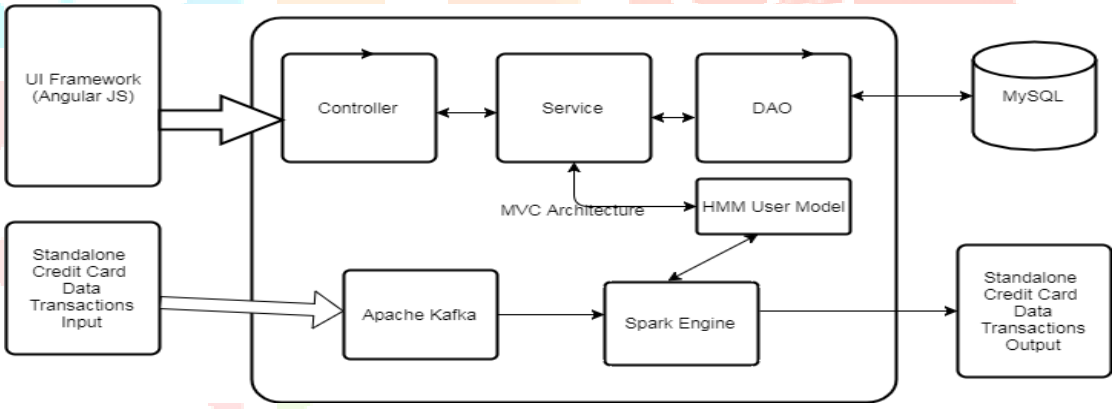

Fig 1: System Architecture

AngularJS: It is a JavaScript structure made by Google for building complex customer side applications. angular's executioner highlight is 'orders' that enable you to expand HTML by making labels and traits. angular tasks have a to some degree unexpected structure in comparison to other JavaScript MVC systems, yet it can be very particular and simple to keep up once you comprehend the structure.

Controller: AngularJS application predominantly depends on controllers to control the stream of information in the application. A controller is characterized utilizing ng-controller mandate. A controller is a JavaScript question containing traits/properties and capacities. Every controller acknowledges $scope as a parameter which alludes to the application/module that controller is to control.

Services: AngularJS supports the idea of "Separation of Concerns" using the architecture of services. These are the JavaScript functions which are responsible to perform a specific tasks only. This makes them an individual entity which can be, maintained and tested. Controllers can be called on the basis of requirement. Services are normally injected using dependency injection mechanism of AngularJS.

DAO: DAO stands for data access object ,it used to separate data persistence logic in a separate layer. All points of interest of capacity are avoided whatever remains of the application. Consequently, conceivable changes to the diligence instrument can be

actualized by simply adjusting one DAO execution while whatever remains of the application isn't influenced. DAOs go about as a middle person between the application and the database. They move information forward and backward amongst items and database records.

MYSQL: MySQL is the most well known Open Source Relational SQL Database Management System. MySQL is a standout amongst other RDBMS being utilized for creating different electronic programming applications.

APACHE KAFKA: Kafka is intended for appropriated high throughput frameworks. Kafka tends to work exceptionally well as a substitution for a more customary message merchant. In contrast with other informing frameworks, Kafka has better throughput, worked in apportioning, replication and inborn adaptation to internal failure, which makes it a solid match for extensive scale message handling applications.

APACHE KAFKA - INTEGRATION WITH SPARK: Spark Streaming API empowers versatile, high-throughput, blame tolerant stream handling of live information streams. Information can be ingested from numerous sources like Kafka, Flume, Twitter, and so on., and can be handled utilizing complex calculations, for example, abnormal state capacities like guide, diminish, join and window. At long last, handled information can be pushed out to filesystems, databases, and live dash-sheets. Strong Distributed Datasets (RDD) is a crucial information structure of Spark. It is a permanent disseminated gathering of items. Each dataset in RDD is isolated into legitimate segments, which might be registered on various hubs of the bunch.

Combination with Spark: Kafka is a potential informing and combination stage for Spark gushing. Kafka go about as the focal center point for continuous surges of information and are prepared utilizing complex calculations in Spark Streaming. Once the information is prepared, Spark Streaming could be distributing comes about into yet another Kafka point or store in HDFS, databases or dashboards. The accompanying graph delineates the applied stream.

IV. WORKFLOW

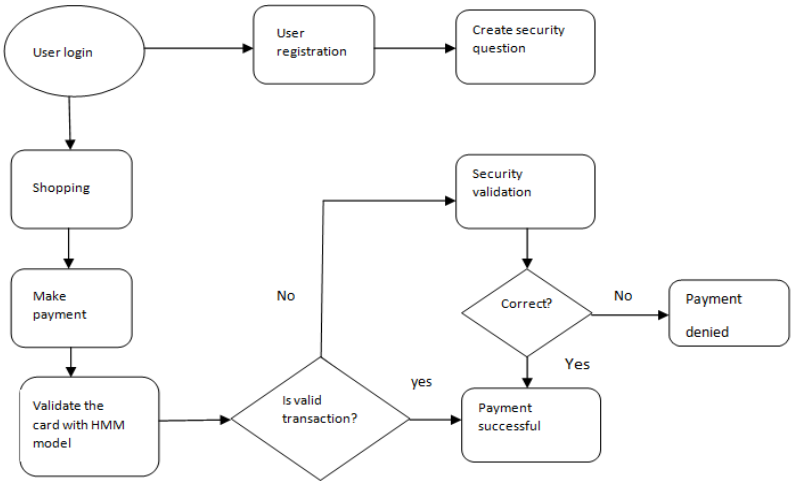

Fig 2: workflow of credit card fraud detection

The above diagram shows the workflow of the credit card fraud detection system. First a user has to log in, whenever a user comes to the website for the first time, he has to register. While registering the user has to create security questions which only he will know. Again the user has to login to the webpage to continue shopping. Then the user can enter the items to the card and can proceed to the checkout. The user has to enter the card details like card number, cardholder's name, expiry date, CVV number to make the payment and the system will validate the credit card data with HMM model.

The HMM model will compare the transaction with the previous transaction history of the user. In this approach, the location, transaction type and IP address or Mac address are compared with the historical pattern. If anyone of the attributes differs from the previous historical pattern, the transaction is considered to be a suspicious transaction. The IP address and Mac address are clubbed together if anyone matches with previous transactions, it will be considered as genuine.

Whenever a suspicious transaction will be made by the user, the user has to validate the security question which he would have made during the transaction. If the user enters the correct answer the transaction will be considered as genuine and the payment will be successful otherwise, the transaction will be considered as fraud and the user will be able to do the transaction, the transaction will be denied.

On the successful transaction, the transaction details will be stored in the database which is on the cloud, next time when similar kind of transaction is made, the user will not have to validate the security question, he can directly make the payment.

V.EXPERIMENTS AND RESULTS

Login screen:

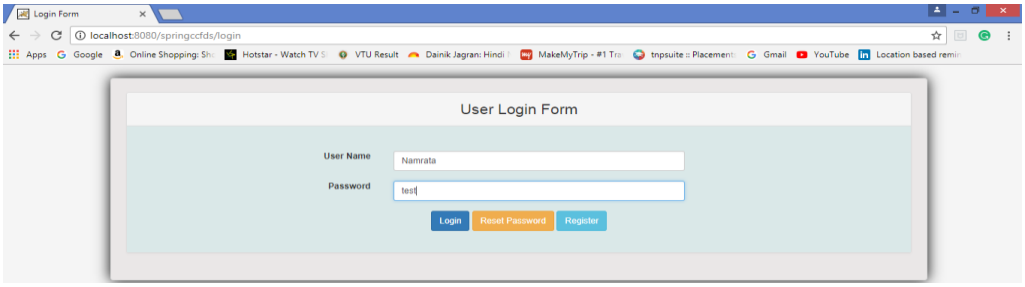

Fig 3: login screen

Shopping Cart Details Screen:

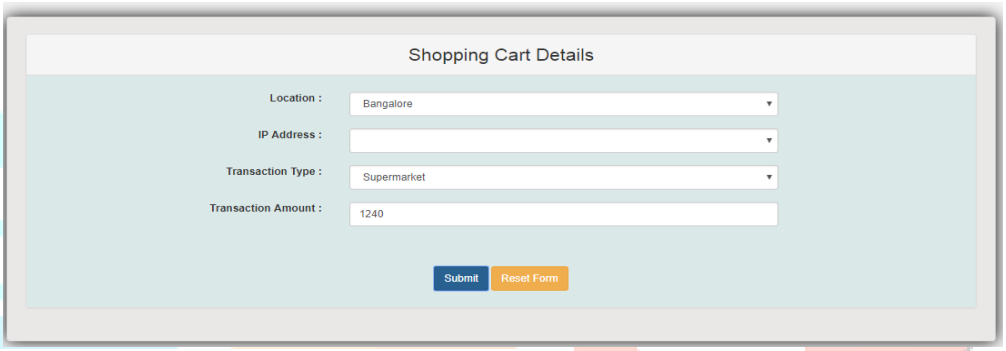

Fig 4: shopping cart details screen

Credit Card Entry Details Screen:

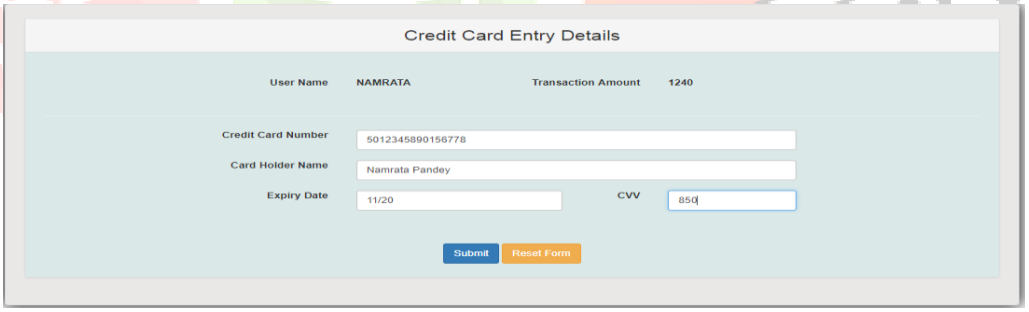

Fig 5: credit card entry detail screen

Validate Security Questions:

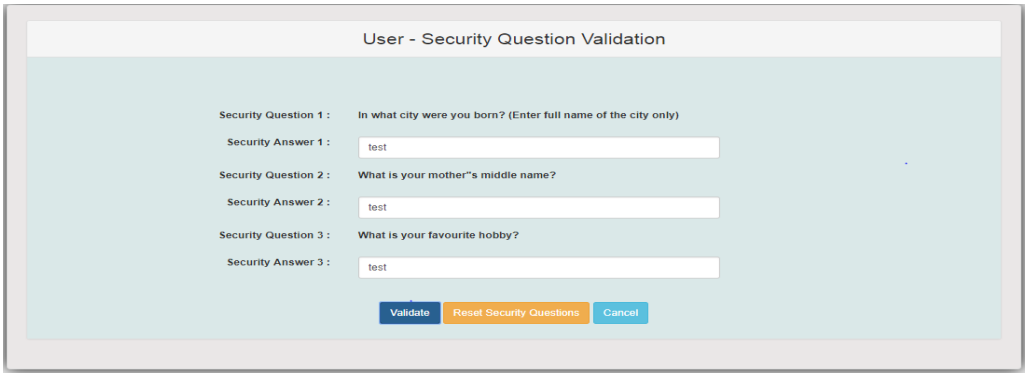

Fig 6:Security question validation

Transaction Completed Successfully:

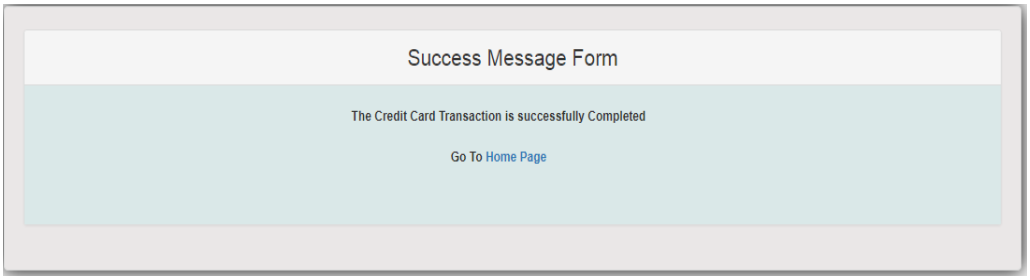

Fig 7: successful transaction

These are the results obtained with real data sets.

VI. CONCLUSION

An exceedingly proficient and exact credit card fraud detection framework is the need of great importance as a many transactions are being done each day. Thus a expansive measure of research is being done in this area furthermore, various systems are proposed conquered credit card extortion. The Fraud Detection System is additionally versatile for taking care of tremendous volumes of transactions. The HMM based credit card fraud detection framework isn't taking long time what's more, having complex procedure to perform misrepresentation check like the existing framework and it gives preferable and quick come about over existing framework. The Hidden Markov Model makes the handling of recognition simple and tries to expel the many-sided quality. In this paper, big data technologies like Spark, Kafka and zookeeper are being used for storing and processing of large amount of historical transactions of the users. Our result shows the effectiveness and correctness of the proposed system over a broad deviation of data.

REFERENCES

[1] Akaash Vishal Hazarika, G JagadeeshSai Raghu Ram, Eeti Jain, “Performance comparison of Hadoop and spark engine”, IEEE International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), pp.671-774, 2017.

[2] VladyslavTaran, Oleg Alienin, SergiiStirenko, Yuri Gordienko, A. Rojbi, “Performance evaluation of distributed computing environments with Hadoop and Spark frameworks”, IEEE International Young Scientists Forum on Applied Physics and Engineering (YSF), pp. 80-83, 2017.

[3] YassirSamadi, MostaphaZbakh, Claude Tadonki, “Comparative study between Hadoop and Spark based on Hibench benchmarks” IEEE 2nd International Conference on Cloud Computing Technologies and Applications (CloudTech), pp.267-275, 2016.

[4] AryaShriharіRamani, R. Sasikala, “Collaborative filtering algorithm using spark and MapReduce”, IEEE International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), pp. 1-7, 2017.

[5] AnkushVerma, AshikHussainMansuri, Neelesh Jain, “Big data management processing with HadoopMapReduce and spark technology: A comparison”, IEEE Symposium on Colossal Data Analysis and Networking (CDAN), pp. 1-4, 2016.

[6] You Dai, Jin Yan, Xiaoxin Tang, Han Zhao and MinyiGuo, “Online Credit Card Fraud Detection: A Hybrid Framework with Big Data Technologies”, IEEE Trustcom/BigDataSE/ISPA, pp. 1644-1651, 2016.

[7] Y. Kltr M. U. alayan, “cardholder behavior model for detecting credit card fraud", IEEE 9th International Conference on Application of Information and Communication Technologies (AICT), pp. 148-152, 2015.

[8] Mohammad Reza HaratiNik, Mahdi Akrami; ShahramKhadivi, Mahdi Shajari, “FUZZGY: A hybrid model for credit card fraud detection”, IEEE 6th International Symposium on Telecommunications (IST), pp. 1088-1093, 2012.

[9] Divya. Iyer, ArtiMohanpurkar; SnehaJanardhan; DhanashreeRathod; AmrutaSardeshmukh, “Credit card fraud detection using Hidden Markov Model”, IEEE World Congress on Information and Communication Technologies, pp. 1062-1066, 2011.

[10] NehaBharill, ArunaTiwari, AayushiMalviya, “Fuzzy Based Clustering Algorithms to Handle Big Data with Implementation on Apache Spark”, IEEE Second International Conference on Big Data Computing Service and Applications (BigDataService), pp. 95-104, 2016.

[11] Ashphak Khan; Tejpal Singh; Amit Sinhal2012, "Implement credit card fraudulent detection system using observation probabilistic in hidden Markov model", Nirma University International Conference on Engineering" (NUiCONE) pp: 1 – 6, 2012.

[12]V. Bhusari; S. Patil, "Study of Hidden Markov Model in credit card fraudulent detection" 2016 World Conference on Futuristic Trends in Research and Innovation for Social Welfare (Startup Conclave) pp: 1 – 4, 2016.

[13] "Credit card fraud detection using Hidden Markov Model"2011 World Congress on Information and Communication TechnologiesYear: 2011 pp: 1062 – 1066

[14]Divya. Iyer; ArtiMohanpurkar; SnehaJanardhan; DhanashreeRathod; AmrutaSardeshmukh "Credit Card Fraud Detection Using Hidden Markov IEEE Transactions on Dependable and Secure Computing, Volume: 5, Issue: 1 pp: 37 – 48, 2008

[15] fraud" YiğitKültür; Mehmet UfukÇağlayan "A novel cardholder behavior model for detecting credit card 2015 9th International Conference on Application of Information and Communication Technologies (AICT) pp: 148 – 152, 2015

[16] Changjun Jiang; Jiahui Song; Guanjun Liu; LutaoZheng; Wenjing Luan ,"Credit Card Fraud Detection: A Novel Approach Using Aggregation Strategy and Feedback Mechanism" IEEE Internet of Things Journal, (Early Access) pp: 1 – 1, 2018

[17] ZahraKazemi; HoumanZarrabi, "Using deep networks for fraud detection in the credit card transactions 2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI)pp: 0630 – 0633, 2017

[18]KuldeepRandhawa; Chu Kiong Loo; ManjeevanSeera; CheePeng Lim; Asoke K. Nandi "Credit Card Fraud Detection Using AdaBoost and Majority Voting" IEEE, Volume: 6 pp: 14277 – 14284, 2018

[19] FabrizioCarcillo; Yann-Aël Le Borgne; Olivier Caelen; GianlucaBontempi "An Assessment of Streaming Active Learning Strategies for Real-Life Credit Card Fraud Detection" 2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA) Pages: 631 – 639, 2017

[20]SumanArora; Dharminder Kumar " Selection of optimal credit card fraud detection models using a coefficient sum approach" 2017 International Conference on Computing, Communication and Automation (ICCCA),pp: 482 – 487, 2017

[21]S Md. S Askari; Md. Anwar Hussain " Credit card fraud detection using fuzzy ID3" 2017 International Conference on Computing, Communication and Automation (ICCCA) pp: 446 – 452, 2017

[22]Luis Vergara; Addisson Salazar; JordiBelda; Gonzalo Safont; Santiago Moral; Sergio Iglesias "Signal processing on graphs for improving automatic credit card frauddetection" 2017 International Carnahan Conference on Security Technology (ICCST) pp: 1 – 6, 2017

[23]John O. Awoyemi; Adebayo O. Adetunmbi; Samuel A. Oluwadare "Credit card fraud detection using machine learning techniques: A comparative analysis" 2017 International Conference on Computing Networking and Informatics (ICCNI)pp: 1 – 9, 2017.

[24]Andrea Dal Pozzolo; GiacomoBoracchi; Olivier Caelen; CesareAlippi; GianlucaBontempi "Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy" IEEE Transactions on Neural Networks and Learning Systems (Early Access) pp: 1 – 14, 2018.

[25]Nathan Fogal; Stephen Adams; Donald E. Brown; Peter A. Beling Mary Frances Zeager; Aksheetha Sridhar; "Adversarial learning in credit card fraud detection" 2017 Systems and Information Engineering Design Symposium (SIEDS) pp: 112 – 116, 2017.