International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 04 Issue: 07 | July -2020 ISSN: 2582-3930

COMPARATIVE STUDY ON CREDIT CARD FRAUD DETECTION

[1]Dr. D.Durga Prasad, [2]Shaziya Mohammad, [3]A.Someswari, [4]A.Meghana, [5] N.Prathyusha

[1]Professor, Dept of CSE, PSCMRCET, Vijayawada, A.P, India

[2,3,4,5]Department of Computer Science & Engineering, Potti Sriramulu Chalavadi Mallikharjuna Rao College of Engineering & Technology, Vijayawada, India.

Abstract:
Credit card fraud is a very serious problem in financial services. Billions of dollars are lost every year due to the fraudulent credit card transactions. Almost every organization is experiencing this economic crime these days. Moreover, the development of new technologies provide additional ways in which criminals may commit fraud. The use of credit cards is prevalent in modern day society. While this might be convenient for almost everyone, on the flip-side fraudulent transactions are on the rise as well. So, implementation of efficient fraud detection systems has become imperative to minimize the losses. Credit card fraud detection can be implemented by using various techniques. We need to understand which techniques give accurate results in order to develop efficient credit card fraud detection model. The project "comparative study on credit card fraud detection" will analyse the performance of various machine learning algorithms like Decision Tree, Random Forest and Support Vector Machine. The performance of these algorithms is evaluated based on accuracy, error rate, sensitivity, specificity and precision.

Keywords: Credit Card, Credit Card fraud, Fraud detection, Machine Learning techniques, Random Forest, Support Vector Machine, Decision Tree.

1. INTRODUCTION

1.1 Brief Overview of the Project

In today's world, we are on the express train to a cashless society. There has been a phenomenal growth in the number of credit card transactions. While this might be convenient for almost everyone, on the flip-side fraudulent transactions are on the rise as well. So, implementation of efficient fraud detection systems has become imperative to minimize the losses. Credit card fraud detection can be implemented by using various techniques. We need to understand which techniques give accurate results in order to develop an efficient model. The project "comparative study on credit card fraud detection" will help us in analysing various machine learning algorithms like Decision Tree, Random Forest and Support Vector Machine. The performance of these algorithms is evaluated based on accuracy, error rate, sensitivity, specificity and precision.

1.1.1 Scope

This project is capable of providing most of the essential features required to detect fraudulent and legitimate transactions. As technology changes, it becomes difficult to track the behaviour and pattern of fraudulent transactions. With the upsurge of machine learning, artificial intelligence and other relevant fields of information technology, it becomes feasible to automate the process and to save some of the effective amount of time and labour that is put into detecting credit card fraudulent activities.

1.1.2 Purpose

The purpose of this project is to identify the algorithm that is best suited for the implementation of effective credit card fraud detection system. To fulfil this purpose, we perform a comparison study to evaluate the performance of various machine learning techniques considering different aspects such as accuracy, error rate, sensitivity, specificity and precision.

1.2 Problem Statement

Credit card fraud is a very serious problem in financial services. Billions of dollars are lost every year due to the fraudulent credit card transactions. Almost every organization is experiencing this economic crime these days. Moreover, the development of new technologies provides additional ways in which criminals might commit fraud. The use of credit cards is prevalent in modern day society and credit card fraud has been growing every year. Financial losses affect not only merchants and banks, but also individuals who use the credit cards. Fraud may also affect the reputation and image of a merchant causing non-financial losses that, though difficult to quantify in the short term, may become visible in the long period. For example, if a cardholder is victim of fraud with a certain company, he may no longer trust their business and choose a contender. Therefore, there is definitely an urge to solve the problem of credit card fraud detection by identifying the efficient techniques for effective implementation.

2. LITERATURE REVIEW

2.1 Related Work

[1] Y. Sahin, S. Bulkan, and E. Duman, "A cost-sensitive decision tree approach for fraud detection," Expert Systems with Applications, vol. 40, no. 15, pp. 5916–5923, 2013.

In this paper, a new cost-sensitive decision tree approach which minimizes the sum of misclassification costs while selecting the splitting attribute at each non-terminal node is developed and the performance of this approach is compared with the well-known traditional classification models on a real world credit card data set. In this approach, misclassification costs are taken as varying. The results show that this cost-sensitive decision tree algorithm outperforms the existing well-known methods on the given problem set with respect to the well-known performance metrics such as accuracy and true positive rate, but also a newly defined cost-sensitive metric specific to credit card fraud detection domain. Accordingly, financial losses due to fraudulent transactions can be decreased more by the implementation of this approach in fraud detection systems.

[2] A. O. Adewumi and A. A. Akinyelu, "A survey of machinelearning and nature-inspired based credit card fraud detection techniques," International Journal of System Assurance Engineering and Management, vol. 8, pp. 937–953, 2017. This paper presents a review of improved credit card fraud detection techniques. Precisely, this paper focused on recent Machine Learning based and Nature Inspired based credit card fraud detection techniques proposed in literature. This paper provides a picture of recent trend in credit card fraud detection. Moreover, this review outlines some limitations and contributions of existing credit card fraud detection techniques and also provides necessary background information for researchers in this domain. Additionally, this review serves as a guide and stepping stone for financial institutions and individuals seeking for new and effective credit card fraud detection techniques.[3] J. T. Quah, and M. Sriganesh, "Real-time credit card fraud detection using computational intelligence," Expert Systems with Applications, vol. 35, no. 4, pp. 1721–1732, 2008.

In this paper, the techniques used for credit scoring are summarized and classified and the new method—ensemble learning model is introduced. This article also discusses some problems in current study. It points out that changing the focus from static credit scoring to dynamic behavioural scoring and maximizing revenue by decreasing the Type I and Type II error are two issues in current study. It also suggested that more complex model cannot always been applied to actual situation. Therefore, how to use the assessment models widely and improve the prediction accuracy is the main task for future research.

[4] S. Bhattacharyya, S. Jha, K. Tharakunnel, and J. C., "Data mining for credit card fraud: A comparative study," Decision Support Systems, vol. 50, no. 3, pp. 602–613, 2011.

This paper evaluates two advanced data mining approaches, support vector machines and random forests, together with the wellknown logistic regression, as part of an attempt to better detect (and thus control and prosecute) credit card fraud. The study is based on real-life data of

transactions from an international credit card operation.

[5] N. S. Halvaiee and M. K. Akbari, "A novel model for credit card fraud detection using Artificial Immune Systems," Applied Soft Computing, vol. 24, pp. 40–49, 2014.

This paper uses Hadoop architecture to provide a fraud detection technique based on the streaming nature of the data. Naïve Bayes is used as the detection technique and Spark framework is used for implementing the algorithm. Experiments show that the proposed framework exhibits excellent detection levels on the fast streaming data.

[6] P. Ravisankar, V. Ravi, G. R. Rao, and I. Bose, "Detection of financial statement fraud and feature selection using data mining techniques," Decision Support Systems, vol. 50, no. 2, pp. 491– 500, 2011.

This paper uses data mining techniques such as Multilayer Feed Forward Neural Network (MLFF), Support Vector Machines (SVM), Genetic Programming (GP), Group Method of Data Handling (GMDH), Logistic Regression (LR), and Probabilistic Neural Network (PNN) to identify companies that resort to financial statement fraud. Each of these techniques is tested on a dataset involving 202 Chinese companies and compared with and without feature selection. PNN outperformed all the techniques without feature selection, and GP and PNN outperformed others with feature selection and with marginally equal accuracies.

2.2 Existing System

In the existing System, a research about a case study involving credit card fraud is considered. Here, data normalization is applied before Cluster Analysis. The results obtained from the use of Cluster Analysis and Artificial Neural Networks on fraud detection have shown that by clustering attributes, neuronal inputs can be minimized and promising results can be obtained using normalized data where the data should be MLP trained. This research was based on unsupervised learning. Significance of this paper was to find new methods for fraud detection and also to increase the accuracy of the results. The data set for this paper is based on real life transactional data by a large European company. The personal details in data were kept confidential. Accuracy of an algorithm is around 50%. Significance of this paper was to find an algorithm which reduces the cost measure. The result obtained was by 23% and the algorithm they found was Bayes minimum risk.

Disadvantages

1. Here, a new collative comparison measure that reasonably represents the gains and losses due to fraud detection is proposed.
2. A cost sensitive method which is based on Bayes minimum risk is presented using the proposed cost measure.

2.3 Proposed System

In the proposed system, we apply various machine learning techniques such as Support Vector Machine, Decision Trees and Random Forest for classification of the credit card dataset. Random Forest has an advantage over Decision Trees and Support Vector Machine as it corrects the habit of over fitting to its training set. A subset of the training set is sampled randomly to train each individual tree and then a decision tree is built. Each node then splits on a feature selected from a random subset of the full feature set. Even for large datasets with many features and data instances, training is extremely fast in Random Forest because each tree is trained independently from others. The Random Forest algorithm has been found to provide a good estimation of the generalization error and to be resistant to over fitting.

Advantages

1. Random Forest ranks the importance of variables in a classification or regression problem in a natural way.
2. The 'amount' feature is the transaction amount and the 'class' feature is the target class for the binary classification and it takes the value '1' for positive case (fraud) and '0' for negative case (no fraud).

2.4 Objective of the Study

The objective is to perform a comparison study in order to analyse the performance of various machine learning techniques such as Random Forest, Support Vector Machine and Decision Trees. These techniques will be applied on the publicly available credit card dataset and their

performance is evaluated based on accuracy, error rate, sensitivity, specificity and precision. This analysis helps us in identifying the technique that gives most accurate results. This study helps in implementing an effective credit card fraud detection system using appropriate and efficient machine learning algorithms.

3. SYSTEM ANALYSIS

3.1 System Study

3.1.1 Feasibility Study

It is wise to think about the feasibility of any problem we take on. Feasibility is the study of impact that occurs in an organization by the development of a system. The impact can be either positive or negative. When the positive dominates the negative, the system is considered feasible. Here, the feasibility study can be performed in two different ways such as technical feasibility and operational feasibility.

1. Technical Feasibility: It is one of the important phases of system development. All the necessary technical requirements such as software facilities and other important resources needed for the development as well as maintenance of the software must be identifiedand must be available when needed. For this project, we are utilizing the resources that are already available.

2. Operational Feasibility: This project is beneficial if and only if it meets all the operating requirements. The operational feasibility study determines whether the new system could be used or not once it is developed and implemented. This project satisfies all the requirements and produces the results as required. Hence, operational feasibility is assured.

3.2 Requirement Analysis

System Requirements Specification (SRS) formally specifies the system-level requirements of a single system or an application. The System Requirements Specification identifies, defines and clarifies the requirements, that when satisfied through development meets the operational/functional need identified in the Project Concept Proposal, Project Business Case, and Project Charter. Approval of this document constitutes agreement that the developed system satisfying these requirements will be accepted.

3.2.1 Functional Requirements

Functional requirements specify which output file should be produced from the given file they describe the relationship between the input and output of the system, for each functional requirement a detailed description of all data inputs and their source and the range of valid inputs are must be specified. In software engineering, a functional requirement defines a function of a software system or its component. A function is described as a set of inputs, the behaviour, and outputs. In requirements engineering, it specifies the particular results of a system. A requirements analyst generates use cases after gathering and validating a set of functional requirements. The Functional Requirements Specification documents the operations and activities that a system must be able to perform.

3.2.2 Non Functional Requirements

Usability: This section includes all the requirements that effect usability. The proposed project will be very easy for the user to understand as the results generated will be very clear.

Reliability: The proposed scheme is very reliable as the code that has been used to implement it is Python.

Performance: The performance of the machine learning techniques used in the proposed project will be evaluated based on various factors as follows.

In the following formulae, TP = True Positives, TN = True Negatives, FP = False Positives and TN = False Negatives.

1.Accuracy: Accuracy (ACC) is calculated as the number of all correct predictions divided by the total number of the dataset. The best accuracy is 1.0, whereas the worst is 0.0. It can also be calculated by 1 – Error_Rate.

Accuracy = (TP+TN) / (TP+TN+FP+FN)

2. Error Rate: Error rate (ERR) is calculated as the number of all incorrect predictions divided by the total number of the dataset. The best error rate is 0.0, whereas the worst is 1.0.

Error Rate = (FP+FN) / (TP+TN+FP+FN)

3. Sensitivity: Sensitivity (SN) is calculated as the number of correct positive predictions

divided by the total number of positives. It is also called recall or true positive rate. The best sensitivity is 1.0, whereas the worst is 0.0.

Sensitivity = TP / (TP+FN)

4. Specificity: Specificity (SP) is calculated as the number of correct negative predictions divided by the total number of negatives. It is also called true negative rate. The best specificity is 1.0, whereas the worst is 0.0.

Specificity = TN / (TN+FP)

5. Precision: Precision (PREC) is calculated as the number of correct positive predictions divided by the total number of positive predictions. It is also called positive predictive value. The best precision is 1.0, whereas the worst is 0.0.

Precision = TP / (TP+FP)

Supportability: The proposed project uses Python language for comparing various algorithms. This makes our project cross platform compatible.

3.3 System Requirement Specification

System Requirements Specification (SRS) specifies all the system-level requirements of a single system or an application.The System Requirements Specification identifies, defines and clarifies the requirements, that when satisfied through development meet the operational/functional needs identified in the Project Concept Proposal, Project Business Case, and Project Charter.

3.3.1 Software Requirements

- Python 3.5.4
- Operating System - Windows 10 (32 or 64 bit)

3.3.2 Hardware Requirements

Minimal hardware requirements are as follows.

- Processor - Intel CORE i5
- RAM - 4 GB
- Hard Disk – 1 TB

3.4 Process Model

Data Pre-processing

Three common data pre-processing techniques are; Formatting: The data you have selected may not be in a format that is suitable for you to work with. The data may be in a relational database and you would like it in a flat file or the data may be in a proprietary file format and you would like it in a relational database or a text file.Cleaning: It is the removal or fixing of missing data. There may be data instances that are incomplete and do not carry the data you believe you need to address the problem. These instances may need to be removed. Additionally, there may be sensitive information in some of the attributes and these attributes may need to be removed from the data entirely.Sampling: There may be far more selected data available than you need to work with. More data can result in much longer running times for algorithms and larger computational and memory requirements. You can take a smaller representative sample of the selected data that may be much faster for exploring and prototyping solutions before considering the whole dataset.

Feature Extraction

Feature extraction is an attribute reduction process.Unlike feature selection, which ranks the existing attributes according to their predictive significance, feature extraction actually transforms the attributes.The transformed attributes, or features, are linear combinations of the original attributes. Finally, our models are trained using Classifier algorithms. We use classify module on Natural Language Toolkit library on Python. We use the gathered labelled dataset. The rest of our labelled data will be used to evaluate the models. Some machine learning algorithms were used to classify pre-processed data. The chosen classifiers were Random forest. These algorithms are very popular in text classification tasks.

Evaluation Model

Agile SDLC (Software Development Life Cycle) is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like;

- Planning
- Requirement Analysis
- Design
- Coding
- Unit Testing
- Acceptance Testing

At the end of the iteration, a working product is displayed to the customer and important stakeholders.

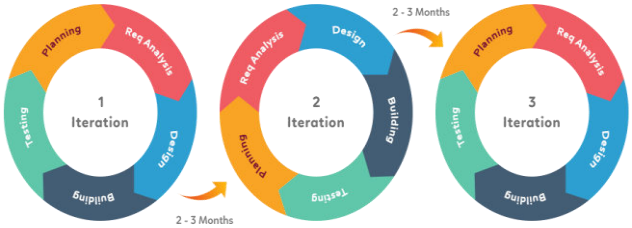

Fig-3.4.1 Agile Model Working

Advantages of Agile Model

- Realistic approach to software development.
- Promotes teamwork and cross training.
- Functionality can be developed rapidly and demonstrated.
- Resource requirements are minimum.
- Suitable for fixed or changing requirements
- Delivers early partial working solutions.
- Good model for environments that change steadily.
- Minimal rules, documentation can be easily employed.
- Enables concurrent development and delivery within an overall planned context.
- Little or no planning required.
- Easy to manage.
- Gives flexibility to developers.

This project uses Agile Model for its implementation. The reason for choosing Agile Model is that its working is similar to the implementation of the project. As Agile is an iterative and incremental model, we can consider the implementation of each algorithm as an iteration. In each iteration, an algorithm is implemented and values of various factors such as accuracy, error rate, sensitivity, specificity and precision. Once all these values are calculated for all the algorithms, we can say which algorithm is effective for the implementation of effective credit card fraud detection system. As this process is similar to the working of Agile Model, we have chosen this instead of all other evaluation models.

4. SYSTEM DESIGN

4.1 About System Design

System design is the process of designing the elements of a system such as the architecture, modules, components, the different interfaces of those components and the data that goes through that system.

The purpose of the System Design process is to provide sufficient detailed data and information about the system and its system elements to enable the implementation consistent with architectural entities as defined in models and views of the system architecture.

Elements of a System

Architecture: This is the conceptual model that defines the structure, behaviour and other views of a system. We can use flowcharts illustrate the architecture.

Modules: These are components that handle one specific task in a system. A combination of the modules makes up the system.

Components: This provides a particular function or group of related functions. They are made up of modules.

Interfaces: This is the shared boundary across which the components of the system exchange information and relate.

Data: This is the management of the information and data flow.

Major Tasks Performed During the System Design Process

A. Initialize design definition

Plan and identify the technologies that will compose and implement the system's elements and its physical interfaces.

Determine which technologies and system elements have a risk to become obsolete, or evolve during the operation stage of the system. Plan their potential replacement. Document the design definition strategy, including the need and requirements of any enabling systems, products, or services to perform the design.

B. Establish design characteristics

Define the design characteristics relating to the architectural characteristics and check that they are implementable.

Define the interfaces that were not defined by the System Architecture process or that need to be refined as the design details evolve. Define and

document the design characteristics of each system element.

C. Assess alternatives for obtaining system elements

Assess the design options

Select the most appropriate alternatives.

If the decision is made to develop the system element, rest of the design definition process and the implementation process are used. If the decision is to buy or reuse a system element, the acquisition process may be used to obtain the system element.

D. Manage the design

Capture and maintain the rationale for all selections among alternatives and decisions for the design, architecture characteristics.

Assess and control the evolution of the design characteristics.

4.2 System Architecture

First, the credit card transactions dataset is taken from the source.Then, cleaning and validation is performed on the dataset which includes removal of redundancy, filling empty spaces in columns, converting necessary variable into factors or classes then data is divided into two parts, one is training dataset and the other is a test dataset. Now the original sample is randomly partitioned into test and train dataset.

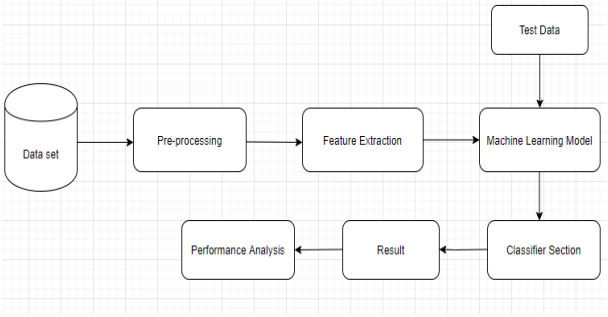

Fig-4.2.1 System Architecture

4.3 Module Description

Based on System analysis, we are proposing and comparing three algorithms. Those are;

a. Decision Tree Classifier
b. Random Forest
c. Support Vector Machine

a. Decision Tree Classifier

Decision Tree is an algorithm that uses a tree like graph or model of decisions and their possible outcomes to predict the final decision, this algorithm uses conditional control statement.It is an algorithm for approaching discrete-valued target functions, in which decision tree is denoted by a learned function. For inductive learning these types of algorithms are very famous and have been successfully applied to abroad range of tasks.Decision rules determine the outcome of the content of leaf node. In general rules have the form of 'If condition 1 and condition 2 but not condition 3 then outcome'. Decision tree helps to determine the worst, best and expected values for different scenarios, simplified to understand and interpret and allows addition of new possible scenarios. The two main entities of a tree are decision nodes, where the data is split and leaves, where we got outcome. The example of a binary tree for predicting whether a person is fit or unfit providing information like age, eating habits and exercise habits, is given

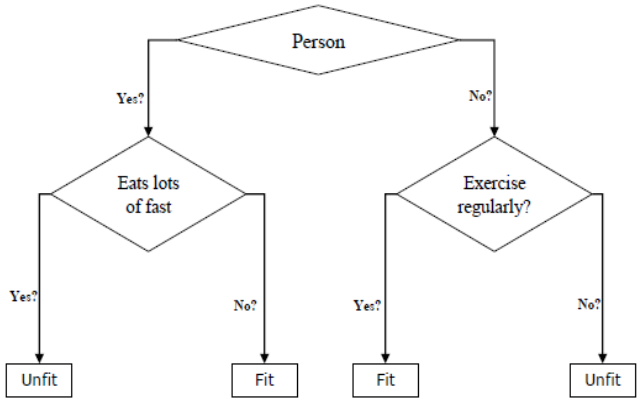

Fig-4.3.1 Classification Decision Tree Example

In the above decision tree, the questions are decision nodes and final outcomes are leaves.

b. Random Forest

Random Forest is an algorithm for classification and regression. It is actually a collection of decision tree classifiers. Random forest has advantage over decision tree as it corrects the habit of overfitting to their training set. A subset of the training set is sampled randomly so to train each individual tree and then a decision tree is built, each node then splits on a feature selected from a random subset of the full feature set. Even for large data sets with many features and data instances training is extremely fast in random forest and because each tree is trained independently of the others. The Random Forest algorithm has been found to provide a good

estimate of the generalization error and to be resistant to overfitting.

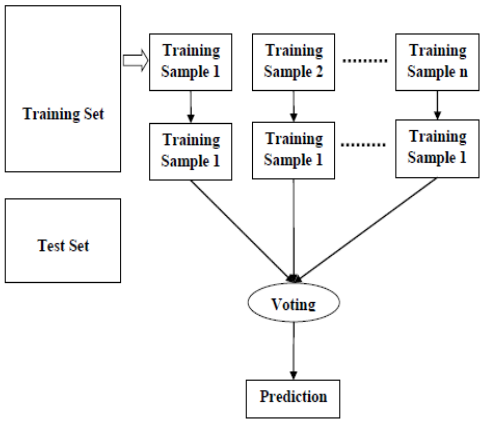

Fig-4.3.2 Working of Random Forest

c. Support Vector Machine

SVM is a one of the popular machine learning algorithm for regression, classification. It is a supervised learning algorithm that analyses data used for classification and regression. SVM modelling involves two steps, firstly to train a data set and to obtain a model & then, to use this model to predict information of a testing data set. A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane where SVM model represents thetraining data points as points in space and then mapping is done so that the points which are of different classes are divided by a gap that is as wide as possible. Mapping is done in the same space for new data points and then predicted on which side of the gap they fall.

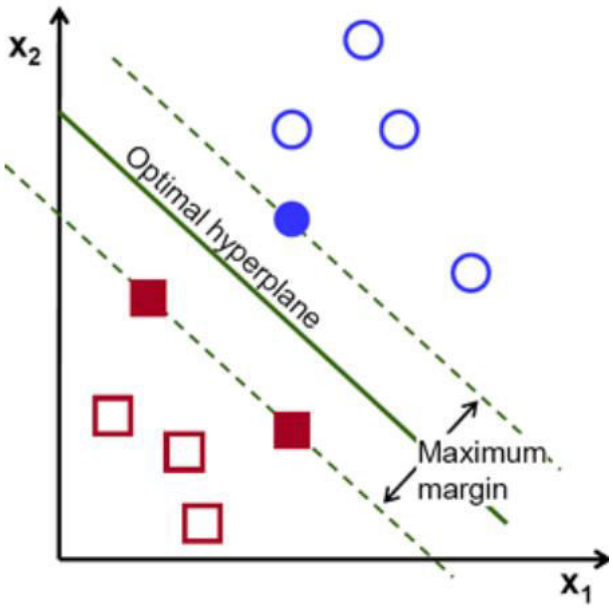

Fig-4.3.3 SVM Model Graph

In SVM algorithm, plotting is done as each data item is taken as a point in n-dimensional space where n is number of features, with the value of each feature being the value of a particular coordinate. Then, classification is performed by locating the hyper-plane that separates the two classes very well.

5. SYSTEM IMPLEMENTATION

5.1 About System Implementation

Implementation is the most crucial stage in achieving a successful system and giving confidence to the user that the new system is workable and effective. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover and an evaluation of changeover methods apart from planning.Two major tasks of preparing the implementation are education and training of the users and testing of the system. The more complex the system is implemented, the more involved will be the system analysis and design effort required just for implementation. The implementation phase comprises of several activities.The required hardware and software acquisition is carried out. The system may require some software to be developed. For this Programs are written and tested. The user then changes over to his new fully tested system and the old system is discontinued.

5.2 Comparison Study: Decision Tree Classifier, Random Forest Classifier and Support Vector Classifier

We applied different classification techniques like Support Vector Machine, Decision Tree and Random Forest and compare the performance of these techniques. The performance is valuated based on accuracy, error rate, sensitivity, specificity and precision.

Techniques / Performance	Decision Tree Classifier	Random Forest Classifier	Support Vector Classifier
Accuracy	99.922 %	99.950 %	99.931 %
Error Rate	0.077	0.049	0.068

Specificity	76.0 %	92.929 %	93.827 %
Sensitivity	99.964 %	99.960 %	99.938 %
Precision	0.76	0.929	0.938

Table – 5.2.1 Comparison Study Results

The above table shows the values of accuracy, error rate, specificity, sensitivity and precision for all the three machine learning techniques like Decision Tree Classifier, Random Forest Classifier and Support Vector Classifier. This table helps us in identifying which algorithm is best suitable for Credit Card Fraud Detection System.

6. CONCLUSION AND FUTURE SCOPE

Conclusion

The proposed scheme analyses various machine learning algorithms based on accuracy, error rate, specificity, sensitivity and precision. Based on the results obtained, we compare different machine learning classification techniques like Decision Trees, Random Forest and Support Vector Machine. Accuracy is one of the important factors based on which we can identify the efficient algorithm for the implementation of any system. The accuracies for the algorithms used in this project are; Decision Tree Classifier: 99.922%, Random Forest Classifier: 99.950% and Support Vector Classifier: 99.931%. If we observe the results of all the implemented algorithms, the accuracies are almost similar. Random Forest Classifier gives highest accuracy when compared with the others. We can observe that Random Forest gives more accurate results. Therefore, it is the most efficient machine learning algorithm for the implementation of effective Credit Card Fraud Detection System.

Future Scope

In the current project, we have compared only three different algorithms. In future, we would like to conduct a comparison study using more machine learning algorithms other than Decision Tree, Random Forest and Support Vector Machine. We would like to use Naïve Bayes Classifier, Neural Networks, Boosted Trees, Logistic Regression and k-Nearest Neighbor algorithms for future study.

7. REFERENCES

[1] Y. Sahin, S. Bulkan, and E. Duman, “A cost-sensitive decision tree approach for fraud detection,” Expert Systems with Applications, vol. 40, no. 15, pp. 5916–5923, 2013.

[2] A. O. Adewumi and A. A. Akinyelu, “A survey of machinelearning and nature-inspired based credit card fraud detection techniques,” International Journal of System Assurance Engineering and Management, vol. 8, pp. 937–953, 2017.

[3] J. T. Quah, and M. Sriganesh, “Real-time credit card fraud detection using computational intelligence,” Expert Systems with Applications, vol. 35, no. 4, pp. 1721–1732, 2008.

[4] S. Bhattacharyya, S. Jha, K. Tharakunnel, and J. C., “Data mining for credit card fraud: A comparative study,” Decision Support Systems, vol. 50, no. 3, pp. 602–613, 2011.

[5] N. S. Halvaiee and M. K. Akbari, “A novel model for credit card fraud detection using Artificial Immune Systems,”Applied Soft Computing, vol. 24, pp. 40–49, 2014.

[6] P. Ravisankar, V. Ravi, G. R. Rao, and I. Bose, “Detection of financial statement fraud and feature selection using data mining techniques,” Decision Support Systems, vol. 50, no. 2, pp. 491– 500, 2011.