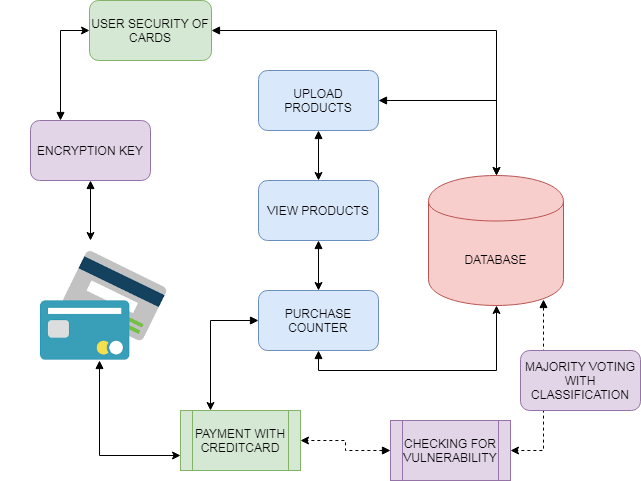
**CREDIT CARD FRAUD DETECTION USING ADABOOST AND MAJORITY VOTING**

**ABSTRACT:**

Credit card fraud is a serious problem in financial services. Billions of dollars are lost due to credit card fraud every year. There is a lack of research studies on analyzing real-world credit card data owing to confidentiality issues. In this paper, machine learning algorithms are used to detect credit card fraud. Standard models are firstly used. Then, hybrid methods which use AdaBoost and majority voting methods are applied. To evaluate the model efficacy, a publicly available credit card data set is used. Then, a real-world credit card data set from a financial institution is analyzed. In addition, noise is added to the data samples to further assess the robustness of the algorithms. The experimental results positively indicate that the majority voting method achieves good accuracy rates in detecting fraud cases in credit cards. In avoiding loss from fraud, two mechanisms can be used: fraud prevention and fraud detection. Fraud prevention is a proactive method, where it stops fraud from happening in the first place. On the other hand, fraud detection is needed when a fraudulent transaction is attempted by a fraudster.

**ARCHITECTURE:**

****

**EXISTING SYSTEM:**

With the rise of e-commerce in the past decade, the use of credit cards has increased dramatically. The number of credit card transactions in 2011 in Malaysia was at about 320 million, and increased in 2015 to about 360 million. Along with the rise of credit card usage, the number of fraud cases has been constantly increased. While numerous authorization techniques have been in place, credit card fraud cases have not hindered effectively. Fraudsters favor the internet as their identity and location are hidden. The rise in credit card fraud has a big impact on the financial industry. The global credit card fraud in 2015 reached to a staggering USD $21.84 billion. Loss from credit card fraud affects the merchants, where they bear all costs, including card issuer fees, charges, and administrative charges. Since the merchants need to bear the loss, some goods are priced higher, or discounts and incentives are reduced. Therefore, it is imperative to reduce the loss, and an effective fraud detection system to reduce or eliminate fraud cases is important. There have been various studies on credit card fraud detection. Machine learning and related methods are most commonly used, which include artificial neural networks, rule-induction techniques, decision trees, logistic regression, and support vector machines. These methods are used either standalone or by combining several methods together to form hybrid models.

**PROPOSED SYSTEM:**

In this paper, a total of twelve machine learning algorithms are used for detecting credit card fraud. The algorithms range from standard neural networks to deep learning models. They are evaluated using both benchmark and real-world credit card data sets. In addition, the AdaBoost and majority voting methods are applied for forming hybrid models. To further evaluate the robustness and reliability of the models, noise is added to the real-world data set. The key contribution of this paper is the evaluation of a variety of machine learning models with a real-world credit card data set for fraud detection. While other researchers have used various methods on publicly available data sets, the data sets used in this paper are extracted from actual credit card transaction information over three months.

**ALGORITHM:**

**NAIVE BAYES ALGORITHM**

A class's prior may be calculated by assuming equiprobable classes (i.e., priors = 1 / (number of classes)), or by calculating an estimate for the class probability from the training set (i.e., (prior for a given class) = (number of samples in the class) / (total number of samples)). To estimate the parameters for a feature's distribution, one must assume a distribution or generate nonparametric models for the features from the training set.[8] The assumptions on distributions of features are called the event model of the Naive Bayes classifier. For discrete features like the ones encountered in document classification (include spam filtering), multinomial and Bernoulli distributions are popular. These assumptions lead to two distinct models, which are often confused. It is a classification technique based on Bayes’ Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as ‘Naive’. Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

**MODULES:**

1. **PURHASE PHASE**

Admin uploads the products and user can view the details. If user wishes to purchase the details then they have to enter details of their own to purchase. The payment is very important phase, in order to protect the details of users’ credit card. The basic process of purchase is done here.

1. **CREDIT CARD SAFETY**

Generally credit cards are required to enter the details in e-commerce websites. The credit card details are contains the card number, expiry date, and verification number through telephone or website is extremely secrecy. The credit card details are given as encrypted format in order to maintain the secrecy. User holds the key to encrypt or decrypt details of credit card.

1. **MAJORITY VOTINGS**

The majority voting does can be collected from the customer’s feedback. From the feedback naïve bayes classifier applied to find the positive or negative or about which product. The main classifications are shown to admin in order to block the details of malfunctioned e commerce websites.

1. **PICTORIAL ANALYSIS**

This is the place where admin have ability to come for particular solution about proposed system. The pictorial representations of collected data are shown in the form of graphs. The different graphs give the best analysis of the system.

**FUTUREWORK:**

For future work, the methods studied in this paper will be extended to online learning models. In addition, other online learning models will be investigated. The use of online learning will enable rapid detection of fraud cases, potentially in real-time. This in turn will help detect and prevent fraudulent transactions before they take place, which will reduce the number of losses incurred every day in the financial sector.

**REQUIREMENT ANALYSIS**

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

**REQUIREMENT SPECIFICATION**

**Functional Requirements**

* Graphical User interface with the User.

**Software Requirements**

For developing the application the following are the Software Requirements:

1. Python
2. Django
3. Mysql
4. Wampserver

**Operating Systems supported**

1. Windows 7
2. Windows XP
3. Windows 8

**Technologies and Languages used to Develop**

1. Python

**Debugger and Emulator**

* Any Browser (Particularly Chrome)

**Hardware Requirements**

For developing the application the following are the Hardware Requirements:

* Processor: Pentium IV or higher
* RAM: 256 MB
* Space on Hard Disk: minimum 512MB

**CONCLUSION:**

A study on credit card fraud detection using machine learning algorithms has been presented in this paper. A number of standard models which include NB, SVM, and DL have been used in the empirical evaluation. A publicly available credit card data set has been used for evaluation using individual (standard) models and hybrid models using AdaBoost and majority voting combination methods. The MCC metric has been adopted as a performance measure; as it takes into account the true and false positive and negative predicted outcomes. The best MCC score is 0.823, achieved using majority voting. A real credit card data set from a financial institution has also been used for evaluation. The same individual and hybrid models have been employed. A perfect MCC score of 1 has been achieved using AdaBoost and majority voting methods. To further evaluate the hybrid models, noise from 10% to 30% has been added into the data samples. The majority voting method has yielded the best MCC score of 0.942 for 30% noise added to the data set. This shows that the majority voting method is stable in performance in the presence of noise.