INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

MODERN METHODOLOGY IN CREDIT CARD FRAUD DETECTION

R. LAKSHMI NARAYANAN

Assistant Professor, Dept of ECE, Apollo Engineering College, Chennai, Tamilnadu, India.

Accepted Date: 22/11/2014; Published Date: 01/12/2014

Abstract: Usage of credit card has been increasing dramatically in recent years as this is being the latest and easier mode of fund transfer, online purchase and direct purchase. In the meanwhile credit card frauds are also increasing tremendously. In this paper, sequence of operations in credit card transaction is modelled using a Hidden Markov Model (HMM) and this will help us in detecting the frauds in credit card transaction. Initially HMM is trained with a normal spending pattern of a credit card holder. Based on the training, HMM checks for whether the credit card transaction is authentic or not. If the transaction made is not found to be authentic, then based on the training provided to HMM, it detects the transaction and is considered to be fraudulent. This also ensures all the genuine transactions are discarded / rejected.

Keywords: Credit Card, Detection

Corresponding Author: MR. R. LAXMI NARAYANAN

Access Online On:

www.ijpret.com

How to Cite This Article:

R Laxmi Narayanan, IJPRET, 2014; Volume 3 (4): 200-209

PAPER-QR CODE

INTRODUCTION

Credit card frauds have been ever growing today. According to A C Nielsen study conducted in 2005, one-tenth of the world's population is shopping online. Germany and Great Britain have the largest number of online shoppers, and credit card is the most popular mode of payment (59 percent). About 350 million transactions per year were reportedly carried out by Barclaycard, the largest credit card company in the United Kingdom, toward the end of the last century. Retailers like Wal-Mart typically handle much larger number of credit card transactions including online and regular purchases. As the number of credit card users rises world-wide, the opportunities for attackers to steal credit card details and, subsequently, commit fraud are also increasing. The total credit card fraud in the United States itself is reported to be $2.7 billion in 2005 and estimated to be $3.0 billion in 2006, out of which $1.6 billion and $1.7 billion, respectively, are the estimates of online fraud.

Credit-card- based purchases can be categorized into two types: 1) physical card and 2) virtual card. In a physical-card based. purchase, the Card holder presents his card physically to a merchant for making a payment. To carry out fraudulent transactions in this kind of purchase, an attacker has to steal the credit card. If the cardholder does not realize the loss of card, it can lead to a substantial financial loss to the credit card company. In \the second kind of purchase, only some important information about a card (card number, expiration date, secure code) is required to make the payment. Such purchases are normally done on the Internet or over the telephone. To commit fraud in these types of purchases, a fraudster simply needs to know the card details. Most of the time, the genuine cardholder is not aware that someone else has seen or stolen his card information. The only way to detect this kind of fraud is to analyze the spending patterns on every card and to figure out any inconsistency with respect to the "usual" spending patterns. Fraud detection based on the analysis of existing purchase data of cardholder is a promising way to reduce the rate of successful credit card frauds. Since humans tend to exhibit specific behaviouristic profiles, every cardholder can be represented by a set of patterns containing information about the typical purchase category, the time since the last purchase, the amount of money spent, etc.

2. HMM BACK GROUND

An HMM is a double embedded stochastic process with two hierarchy levels. It can be used to model much more complicated stochastic processes as compared to a traditional Markov model. An HMM has a finite set of states governed by a set of transition probabilities. In a

particular state, an outcome or observation can be generated according to an associated probability distribution. It is only the outcome and not the state that is visible to an external observer.

An HMM can be characterized by the following:

N is the number of states in the model. We denote the set of states S = {S1;,S2, . . . SN}, where Si , i = 1,2….,N is an individual state. The state at time instant t is denoted by qt.

M is the number of distinct observation symbols per state. The observation symbols correspond to the physical output of the system being modeled. We denote the set of symbols V = {V1, V2, . . . VM},

where Vi, i = {1, 2, ...M} is an individual symbol.

The state transition probability matrix A = [aij], where

aij = P(qt+ 1 = Sj \ qt =Si), 1 ≤ i ≤ N , 1 ≤ j ≤ N ;

For the general case where any state j can be reached from any other state i in a single step, we have

aij > 0 for all i , j. Also , ∑ aij = 1, 1 ≤ i ≤ N.

The observation symbol probability matrix B =[bj(k)]

Where

bj(k) = P(Vk\Sj), 1 ≤ j ≤ N , 1 ≤ k ≤ M and ∑bj(k) = 1, 1 ≤ j ≤ N.

The initial state probability vector π = [π i], where

πi = P(q1 = Si), 1 ≤ i ≤ N, such that ∑ πi = 1. 6. The observation sequence O =

O1;O2;O3; . . .OR, where each observation Ot is one of the symbols from V, and R is the number of observations in the sequence.

It is evident that a complete specification of an HMM requires the estimation of two model parameters, N and M, and three probability distributions A, B, and π. We use the notation λ = (A,B,π) to indicate the complete set of parameters of the model, where A, B implicitly include N and M.

An observation sequence O, as mentioned above, can be generated by many possible state sequences. Consider one such particular sequence

Q = q1, q2 , ... qR. where q1 is the initial state.

The probability that O is generated from this state sequence is given by

$$1 (O\backslash Q, \lambda) = \prod P(Ot\backslash qt, \lambda)$$

7. ROLE OF HMM FOR CREDIT CARD FRAUD DETECTION

An FDS runs at a credit card issuing bank. Each incoming transaction is submitted to the FDS for verification. FDS receives the card details and the value of purchase to verify whether the transaction is genuine or not. The types of goods that are bought in that transaction are not known to the FDS. It tries to find any anomaly in the transaction based on the spending profile of the cardholder, shipping address, and billing address, etc. If the FDS confirms the transaction to be malicious, it raises an alarm, and the issuing bank declines the transaction. The concerned cardholder may then be contacted and alerted about the possibility that the card is compromised. In this section, we explain how HMM can be used for credit card fraud detection.

[3] SPENDING PROFILE OF CARD HOLDERS

The spending profile of a cardholder suggests his normal spending behavior. Cardholders can be broadly categorized into three groups based on their spending habits, namely, high-spending (hs) group, medium- spending (ms) group, and low-spending (ls) group. Card holders, who belong to the hs group, normally use their credit cards for buying high priced items. Similar definition applies to the other two categories also.

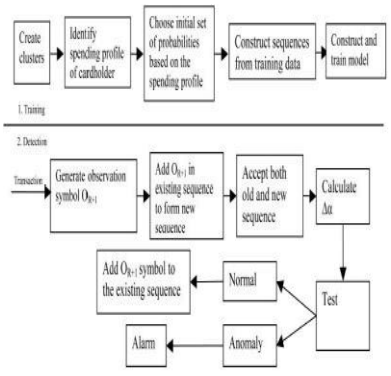

Fig. 2 Process flow of the proposed Fraud Detection System

TABLE. 1. Spending profile of cardholder

Transaction No.	1	2	3	4	5	6	7	8	9	10
Dollar Amount	40	25	15	5	10	25	15	20	10	80

Spending profiles of cardholders are determined at the end of the clustering step. Let pi be the percentage of total number of transactions of the cardholder that belong to cluster with mean ci. Then, the spending profile (SP) of the cardholder u is determined as follows:

SP(u) = arg max(pi) i

Thus, spending profile denotes the cluster number to which most of the transactions of the cardholder belong. In the example, the spending profile of the cardholder is 2, that is ,m and, hence, the cardholder belongs to the ms group.

5. FRAUD DETECTION

After the HMM parameters are learned, we take the symbols from a cardholder's training data and form an initial sequence of symbols. Let O1, O2,……OR be one such sequence of length R.

This recorded sequence is formed from the cardholder's transactions up to time t. We input this sequence to the HMM and compute the probability of acceptance by the HMM. Let the probability be α1 which can be written as follows:

α1 = P (O1,O2,……OR\λ)

Let OR+1 be the symbol generated by a new transaction at time t+1. To form another sequence of length R, we drop O1 and append OR+1 in that sequence, generating O2,O3, . . .OR,OR+1 as the new sequence. We input this new sequence to the HMM and calculate the probability of acceptance by the HMM. Let the new probability be α2

α2 = P (O2,O3,O4…… O R+1\λ),

Δα = α1 - α2

If Δα > 0, it means that the new sequence is accepted by the HMM with low probability, and it could be a fraud. The newly added transaction is determined to be fraudulent if the percentage change in the probability is above a threshold, that is,

Δα/α1 ≥ Threshold

The threshold value can be learned empirically, as will be discussed. If ORþ1 is malicious, the issuing bank does not approve the transaction, and the FDS discards the symbol. Otherwise, ORþ1 is added in the sequence permanently, and the new sequence is used as the base sequence for determining the validity of the next transaction. The reason for including new non malicious symbols in the sequence is to capture the changing spending behaviour of a cardholder. Fig 2 shows the complete process flow of the proposed FDS. As shown in the figure, the FDS is divided into two parts—one is the training module, and the other is detection. Training phase is performed offline, whereas detection is an online process.

6. RESULTS

Testing credit card FDSs using real data set is a difficult task. Banks do not, in general, agree to share their data with researchers. There is also no benchmark data set available for experimentation. We have, therefore, performed large-scale simulation studies to test the efficacy of the system. A simulator is used to generate a mix of genuine and fraudulent transactions. The number of fraudulent transactions in a given length of mixed transactions is normally distributed with a user specified _ (mean) and _ (standard deviation), taking cardholder's spending behaviour into account. _ specifies the average number of fraudulent transactions in a given transaction mix. In a typical scenario, an issuing bank, and hence, its FDS receives a large number of genuine transactions sparingly intermixed with fraudulent transactions. The genuine transactions are generated according to the cardholders' profiles. The cardholders are classified into three categories as mentioned before—the low, medium, and hs groups. We have studied the effects of spending group and the percentage of transactions that belong to the low-, medium-, and high-price-range clusters. We use standard metrics—True Positive (TP) and FP, as well as TP-FP spread and Accuracy metrics, as proposed in [7] to measure the effectiveness of the system. TP represents the fraction of fraudulent transactions correctly identified as fraudulent, whereas FP is the fraction of genuine transactions identified as fraudulent. Most of the design choices for a FDS that result in higher values of TP, also cause FP to increase. To meaningfully capture the performance of such a system, the difference between TP and FP, often called the TP-FP spread, is used as a metric.

Accuracy represents the fraction of total number of transactions (both genuine and fraudulent) that have been detected correctly. It can be expressed as follows:

$$\text{Accuracy} = \frac{\text{No: of good trans: detected as good + No: of Bad trans: detected as bad}}{\text{Total No: of transactions}}$$

We first carried out a set of experiments to determine the correct combination of HMM design parameters, namely, the number of states, the sequence length, and the threshold value. Once these parameters were decided, we performed comparative study with another FDS.

6.1 CHOICE OF DESIGN PARAMETERS

Since there are three parameters in an HMM, we need to vary one at a time keeping the other two fixed, thus generating a large number of possible combinations. For choosing the design parameters, we generate transaction sequences using 95 percent low value, 3 percent medium value, and 2 percent high value transactions. The reason for using this mix is that it represents a profile that strongly resembles a ls customer profile. We also consider the μ and values to be 1.0 and 0.5, respectively. This is chosen so that, on the average, there will be 1 fraudulent transaction in any incoming sequence with some scope for variation. After the parameter values are fixed, we will see, how the system performs as we vary the profile and the mix of fraudulent transactions.

For parameter selection, the sequence length is varied from 5 to 25 in steps of 5. The threshold values considered are 30 percent, 50 percent, 70 percent, and 90 percent. The number of states is varied from 5 to 10 in steps of 1. We consider both TP and FP for deciding the optimum parameter values. Thus, there are a total of 120 (5X4X6) possible combinations of parameters. The number of simulation runs required for obtaining results with a given confidence interval (CI) was derived as follows

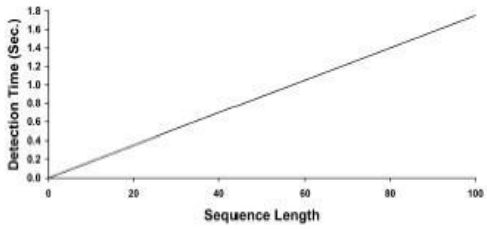

Fig. 3 Detection time versus length of sequence.

An initial set simulation runs each with100samples, was carried out to estimate the mean and standard deviation of both TP and FP for a fixed sequence length, number of states, and threshold value. Mean TP was found to be an order of magnitude higher than mean FP. Standard deviation of TP was 0.1 and that for FP was 0.005. We set the target 95 percent CI for TP and FP, respectively, as 2.5 percent around their mean values. Using Student's t-distribution, the minimum number of simulation runs required for obtaining desired CI for TP was derived as 83 and that for FP as 23. Based on these observations, we set the number of simulation runs for all the experiments to be 100. The results obtained were within the desired CI, as mentioned above.

Since it is not convenient to present the detailed results for each of the 120 combinations, we show summarized results. We show the results for each value of sequence length averaged over all the six states. Similarly, we present results for each value of the number of states averaged over all the five sequence lengths. In the highest value of TP, as well as the lowest value of FP, has been highlighted for each row. It is seen that FP shows a clear trend of decreasing with higher threshold and smaller sequence lengths. However, the number of states does not have a strong influence either on TP or on FP. It is seen that TP is high for sequence length 15 in 75 percent of the cases. Also, fraud detection time increases linearly with the sequence length, as shown in Fig. 3. The results have been plotted for a Java implementation on a 1.8 GHz Pentium IV processor machine. Hence, we choose 15 as the length of observation symbol sequence for optimum performance. Once sequence length is decided, it is seen that the threshold could be set to either 30 percent or 50 percent. Although TP is higher for threshold = 30%, FP is also higher. To minimize FP, we choose threshold = 50%. After choosing sequence length and threshold, we have to choose the number of states. Since there is no clear indication from the above summary information, we take a look at the detailed data for TP when threshold =50%.

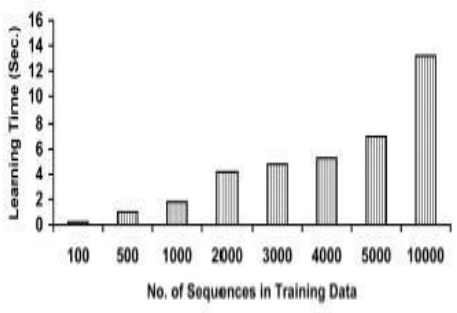

Fig. 4 Model learning time versus number of sequences in training data.

It is seen that for sequence length = 15, the highest value of TP occurs for no: of states = 10 and, hence, it would be a good choice for our design. We have also analyzed the time taken by the training phase, which is performed offline for each cardholder's HMM. Fig. 4 shows the plot of model learning time against the number of sequences in the training data. As the size of training data increases, learning time increases, especially beyond 100. We therefore, use 100 sequences for training the HMM. Although done offline, the model learning time has a strong impact on the scalability of the system. Since an HMM is trained for each cardholder, it is imperative that the training time is kept as low as possible especially when an issuing bank is meant to handle millions of cardholders with many new cards being issued everyday. The online processing time of about 200 ms on a 1.8 GHz Pentium IV machine also shows that the system will be able to handle a large number of concurrent operations and, hence, is scalable.

Thus, our design parameter setting is given as follows:

1. number of hidden states N = 10,

2. length of observation sequence R = 15,

3. Threshold value = 50%, and

4. Number of sequences for training = 100.

7. CONCLUSION

In this paper, we have proposed an application of HMM in credit card fraud detection. The different steps in credit card transaction processing are represented as the underlying stochastic process of an HMM. We have used the ranges of transaction amount as the observation symbols, whereas the types of item have been considered to be states of the HMM. We have suggested a method for finding the spending profile of cardholders, as well as application of this knowledge in deciding the value of observation symbols and initial estimate of the model parameters. It has also been explained how the HMM can detect whether an incoming transaction is fraudulent or not. Experimental results show the performance and effectiveness of our system and demonstrate the usefulness of learning the spending profile of the cardholders. Comparative studies reveal that the Accuracy of the system is close to 80 percent over a wide variation in the input data. The system is also scalable for handling large volumes of transactions.

REFERENCE

1. S. Ghosh and D.L. Reilly, "Credit Card Fraud Detection with a Neural-Network," Proc. 27th Hawaii Int'l Conf.System Sciences: Information Systems: Decision Support And Knowledge-Based Systems, vol. 3, pp. 621-630, 1994.

2. M. Syeda, Y.Q. Zhang, and Y. Pan, "Parallel Granular Networks for Fast Credit Card Fraud Detection," Proc. IEEE Int'l Conf. Fuzzy Systems, pp. 572-577, 2002.

3. S. J. Stolfo, D.W. Fan, W. Lee, A.L. Prodromidis, and P.K. Chan, "Credit Card Fraud Detection Using Meta-Learning: Issues and Initial Results," Proc. AAAI Workshop AI Methods in Fraud and Risk Management, pp. 83-90, 1997.

4. S.J. Stolfo, D.W. Fan, W. Lee, A. Prodromidis, and P.K. Chan, "Cost-Based Modeling for Fraud and Intrusion Detection: Results from the JAM