SIMATS SCHOOL OF ENGINEERING

CHENNAI

CAPSTONE PROJECT REPORT

Course Name: CSA1674-Data Warehousing and Data Mining for Search Engines

Submitted in the partial fulfillment for the award of the degree of Bachelor of Engineering

IN

Computer science engineering

Submitted by

B. Nirmal kumar.

192211118

Under the Supervision of

Dr.Porkodi V

Abstract— Today, people are more used to online shopping and payment due to which frauds related to online payment is on the rise. The main modes of online payment are credits cards, debit cards, net banking. The fraud related to credit cards is increasing nowadays because people prefer using them as an easy mode of payment. In this paper, we have implemented a mechanism to detect credit card fraud using neural networks using Neuroph IDE. This paper presents the various parameters that were considered while training and testing the neural network. Also, we have presented the results and conclusions of our experiment**.**

Keywords— Credit card frauds, Neural Networks, Multilayer Feed forward network, Back propagation, Neuroph

INTRODUCTION

The rate of credit card frauds is increasing at a very higher rate because attackers are becoming more and more sophisticated and are well equipped. This paper presents a paradigm to detect credit card frauds using artificial neural networks. We present a model using Neuroph IDE which provides an environment to work with neural networks. An artificial neural network is similar to a biological neuron which accepts many inputs, processes it and gives us a single output. The neural network needs to be trained continuously with a set of inputs. The network is made to learn so that when a new data is fed into it, it can properly classify it based on the learning it acquired. The learning can be said to be accurate and efficient if it gives the expected output for the test data.

TYPES OF CREDIT CARD FRAUDS

There are three classes of frauds namely card related, merchant related and internet frauds. Some of them are listed below:

A. Card Related Frauds

1) Application Frauds*:* This type of fraud occurs when the fraudster manipulates the application by gaining entry to a customer's personal information and opens a fraudulent account in his name.

2) Stolen Card*:* This type of fraud occurs when the fraudster simply steals a customer's card. In this case, the customer might feel he has lost his card, but actually his card might have been acquired by an attacker.

B. Merchant Related Frauds

1) Merchant Collusion*:* This type is done when a merchant purposely passes on his customer's personal information to fraudsters.

2) Triangulation: Here, the fraudster creates a fake website and operates from there. Many discounts are given to the customer on this website due to which users are attracted to such websites. They purchase items and in the course enter their personal information. This information is obtained by fraudsters and they use it to perform illegitimate transactions.

C. Internet Frauds

1) False Merchant Sites*:* In this type, the website asks the customers to enter their personal details if they wish to access the content of the site. In this way, fraudsters collect many credit card number which they use later for performing fraudulent transaction.

Some of the latest types of credit card frauds are as follows:

1) Keystroke Loggers: "Keystroke logger" is a spyware which infects a user's computer unknown to him. This spyware tracks all the details typed by the user and gives this information to the fraudster who thus obtains all the personal details.

2) Cell Phone Camera Scam: When a customer is paying his bills, a fraudster may be roaming somewhere near him. The customer may be under the assumption that the attacker is busy chatting on his phone, but actually he is taking digital images of the customer's details such as card number, expiry date, etc. This type of fraud is possible because of powerful cameras used these days.

III. Artificial Neural Network

Humans have a tendency of learning new concepts or ideas with the aid of an example. Rather than understanding algorithms step by step, the human brain has an inclination to grasp new things by analysing and studying examples. After the invention of computers and the tremendous advancements in the digital world, attempts are being made to simulate this human approach of learning. A lot of efforts are being put in creating an artificial neuron which works similar to a biological neuron. A network of such artificial neurons is termed as an artificial neural network.

Perceptron is one of the basic models of artificial neural networks. It contains two layers; namely the input layer and the output layer. The function of the input layer is to obtain data and send the same to the output layer where it is gathered. In such supervised networks, training plays a vital role. The training enables the network to generalise. It helps the network in understanding how to observe features and classify the input based on them. Training also makes the network capable of predicting classes for future inputs. Various functions and threshold values are implemented in this process of classification. An important aspect of training is when to stop it. It is very important to understand when to assume that the training is complete. Various parameters can be considered to cease the training, like, maximum number of iterations in the training, upper and lower bounds on the percentage of error and many more.

One of the striking features of the Perceptron model is its ability measure the convergence to an optimal solution. In other words, it can find out the number of finite iterations required to reach the optimal solution.

Initial weights are usually chosen at random. The rate of convergence is directly proportional to the rate of learning.

Multilayer perceptron is one of the most significant models of artificial neural networks. It is a feed-forward supervised type of neural network. Unlike the Perceptron model, the multilayer perceptron has a hidden layer and can deliver outputs with more than two classes. One of the most important aspects of multilayer perceptron is designing the hidden layer. The hidden layer should be complex enough; that is, the hidden layersshould contain sufficient neurons to understand the input features and generate three different classes of outputs. Lesser the number of hidden layers better is the working and output.

Back-propagation is used in Multilayer Perceptron to optimise the weights. The neural network processes a group of known pairs of input-output. Input whose correct output is known is fed to the network. The network processes the input using the layers in it and generates the output. This output is compared to the correct, already known output and the error is calculated. This is the total error. Every neuron in the network has contributed to this error. This error is divided and assigned to the various neurons present in the output layer. Then the error is back propagated; that is, it is further broken down and assigned to neurons of the previous layers. This is continued till the first layer. After this backpropagation every neuron understands it role in the final error. Based on the error, every neuron changes its weight to nullify or mitigate the same. Due to these changes, the network learns and generates better results.

EXPERIMENTAL DESCRIPTION

Credit card fraud have been prevalent for a very long time and several algorithms have been devised using a variety of classification mechanisms, like Bayesian classification, Hidden Markov model (HMM) etc. However, in the HMM model the inputs play a very critical role in classifying the transactions into legit and fraudulent and even though the Bayesian network is robust, it may not be as accurate. Hence we need a system can intelligently learn about the nature of transactions by studying them. Neural networks seems like an obvious solution to the problem. We have made use of neural networks by providing it a set of training data for learning and some test data to assess its accuracy in classification. Neuroph is the platform used for our experiment.

Neuroph is lightweight object oriented Java Neural Network Framework. This Neuroph is an open source Java Neural Network Framework which helps in developing artificial neural network architectures. Neuroph consists of the Java library and a GUI neural network editor called Neuroph Studio. It is considered to be easy to learn and use as it consists of small number of basic classes that correspond to basic Neural Network concepts, and an editor with an intuitive GUI.

Training a neural network using Neuroph studio is very simple. The 5 steps for training a Neural Network using
Neuroph studio,

- Firstly, we begin by creating a new Neuroph Project in the Neuroph studio.
- Then, we creating a Perceptron network.
- Next step includes creating a new training set.
- Once everything is set, we begin training the network giving out all the parameters like learning rate, momentum, etc.
- And then we finally test the trained network.

A. Input – Dataset

The sample dataset that we are working on was acquired from a data mining blog. This dataset contains the summary of the transactions of 20000 active credit card holders past six months. The input fields include card holder ID, total purchase amount, monthly average balance, total number of purchase transactions and 14 different purchase categories like accessories, appliances, books, apparel, fitness, entertainment, etc. The dataset contains the details of a how much a particular card holder spends on what type of goods. It also shows how much amount was spent in a month on an average giving the average number of transactions performed to do so. It simply gives the details of the cardholders' transactions without stating whether the transactions were legitimate or fraudulent. As for a given cardholder the dataset clearly states his spending areas with a particular number of transaction giving out his average monthly spending and average monthly balance. Based on the various different permutations and combinations considering all the possible areas we decided the legitimacy of the transactions. Meaning taking into account the spending areas, number of transactions, average monthly balance and total purchase amount we decided whether the transaction was to be considered to be fraudulent or not. For example, if a card holder's total purchases amount to $3860 in 42 transactions, of which $400 spent on accessories, $50 on appliances, $1000 on apparel, $800 on entertainment, $600 on food, $260 on health, and $750 on telecommunication services. This shows that the spending area of this particular card holder is across various categories and moderate level using as good as 42 transactions. Hence can be categorized as a legitimate transaction. In another example wherein the total purchases amounts to around $2000 and the entire amount is spent only on gas at gas stations that too in a mere 2-3 transactions. Transactions like these that seem to be nonsensical spending $2000 only on a single area with low number of transactions are categorized as fraudulent ones. On the basis of such hypothesis currently 300 transactions are categorized either as fraudulent or genuine. We propose to expand this dataset further to contain the entire 20000 records. But presently continue with the 300 records in hand. The software used for the implementation is Java Neuroph. This allows us to partition the input dataset into two. One for the training purpose and the other for testing. The partition ratio can either be decided randomly or be given a desired fraction.

B. Processing the given input

The neural network has to now be designed for training. We have chosen 19 neurons in the input layer, 15 in the hidden layer and 3 in the output layer. Multilayer feed forward network using sigmoid transfer function with backpropagation of error along with momentum has been put to use. Now, given the input the neural network needs to be provided some additional information to enable it to learn the transactions more efficiently. The essential parameters include: max error, learning rate, momentum. Max error is maximum acceptable error in classifying a transaction. Learning rate manages the change in weight for every neuron and the bias. Momentum ensures that the system does not converge at a local minimum. Higher the momentum, faster will the system converge, leaving the network unstable. If the momentum is too low, the system is bound to converge at a local minimum. Hence, it is key to choose the right momentum while we train the network. We may or may not control the number of iterations. Even though we can limit the iterations, it is recommended to let the network iterate till it is completely thorough. For training our network, we have taken the max error = 0.03, learning rate = 0.9 and momentum of 0.2.

After training the network we provided it a test set. The results obtained are shown in Fig. 2 and Table 1. As we can observe, the classification is very accurate and within the limit of maximum error. Since we did not limit the number of iterations, the network trained itself in the number of iterations it required, in this case 1111. The network stops training once the mean square error consistently comes out to be less than the max error and approaches zero. After training, the mean square error = 0.02374.

RESULTS

TABLE 1
DESIRED & ACTUAL RESULTS

Transaction No.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Customer Id	10,373	10,353	10,391	10,313	10,312	10,368	10,362	10,393

Average Balance	1,148.5904	201.5127	4,096.4744	145.6962	114.0714	1,843.096	2,259.273	143.3401
Tenure	12	12	12	9	12	12	12	12
Number of Transactions	0	12.5978	126.6986	0.9326	7.7676	13.5736	24.4721	25.487
Accessories	0	75.6258	380.0059	0	0	96.6091	389.1568	211.7663
Appliances	0	0	26.2484	0	0	0	2,277.1027	108.0086
Culture	0	0	0	0	0	0	0	0
Gas	0	0	0	0	386.1504	353.5243	0	0
Books	0	0	0	0	0	0	0	0
Apparel	0	126.105	1,021.1132	0	0	48.2637	0	244.2701
Fitness	0	0	0	0	0	0	0	0
Education	0	0	0	0	0	0	0	0
Entertainment	0	0	0	1,129.1896	0	0	0	140.9639
Food	0	0	3,057.39	0	0	310.9706	0	0
Health	0	0	156.5937	0	0	0	0	0
Garden	0	0	25.845	0	0	0	0	0
Tele Communications	0	1,159.8323	0	0	0	0	0	0
Travel	0	0	91.9916	0	725.7461	212.8545	0	0
Expense	0	1,361.563	4,759.1878	1,129.1896	1,111.8965	1,022.2223	2,666.2596	705.009

Transaction No.		**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Desired Output	Yes	1	0	0	0	0	1	0	0
	No	0	1	0	1	1	0	1	1
	May Be	0	0	1	0	0	0	0	0
Actual Output	Yes	0.9723	0	0.033	0	0.0249	0.5809	0.0003	0.2465
	No	0.082	1	0.6643	1	0.9862	0.0388	0.9865	0.7675
	May Be	0.0104	0	0.3657	0	0.0083	0.1962	0.06	0.0012

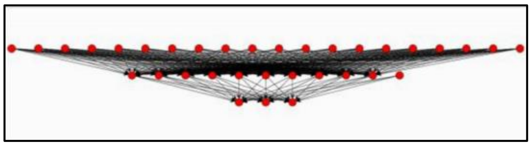

Fig 1. Architecture of the neural network used for the experiment

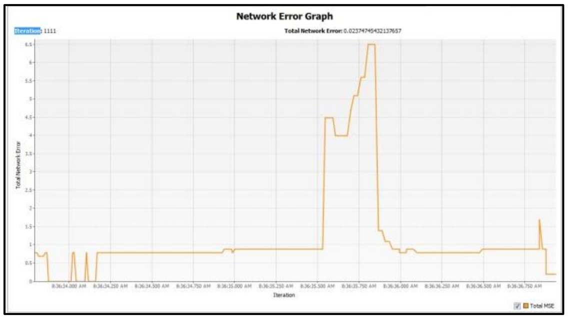

Fig. 2. Total network error graph

CONCLUSION

Credit Card Fraud detection using neural networks is a vast topic. Our research paper focuses on its basic implementation in Neuroph. The results obtained are based on real, though limited data and a neural network having a single hidden layer. The results generated can be further optimised by increasing the number of transactions in the training sets and changing the neural network architecture.

FUTURE SCOPE

The future work of our experiment will include increasing the number of records of the dataset for training the data in order to get accurate results and the network will be able to learn more efficiently with more number of records. We aim to perform the same experiment using other softwares such as JOONE, Weka & RapidMiner. We also aim to compare the results obtained from these softwares to determine the best software that can detect credit card fraud by changing various parameters such as learning rate, momentum, etc. and analyse the same.

ACKNOWLEDGEMENT

I would like to take this opportunity to express my deepest gratitude to all those who have supported us in our research work. I would like to specially acknowledge our Mentor, Prof. Alpa Reshamwala for her valuable guidance and for helping us in writing this paper. A special thanks to our Computer Science Department for giving us an opportunity to do our final year project in the research field. Lastly, I would like to thank our Head of Department, Prof. Dhirendra Mishra for his motivation and support.

REFERENCES

[1] Tej Paul Bhatla, Vikram Prabhu & Amit Dua, "Understanding credit card frauds", Cards Business Review#2003–01, June 2003.

[2] Raghavendra Patidar, Lokesh Sharma, "Credit Card Fraud Detection Using Neural Network", International Journal of Soft Computing and Engineering (IJSCE), ISSN: 2231-2307, Volume-1, Issue-NCAI2011, June 2011, pp. 32-38.

[3] V. Bhusari, S. Patil, "Study of Hidden Markov Model in Credit Card Fraudulent Detection", International Journal of Computer Applications (0975-8887),Volume 20-No.5, April 2011, pp. 33-36.

[4] Artificial Neural Networks:http://www.psych.utoronto.ca/ users/reingold/courses/ai/cache/neural2.html

[5] Neuroph: http://neuroph.sourceforge.net/