Efficient Scalable Multi-Level Classification Scheme for Credit Card Fraud Detection

Dipti D.Patil, Sunita M.Karad, Vijay M.Wadhai, J A Gokhale, Prasad S.Halgaonkar
[†]Assistant Professor, MITCOE, Pune INDIA
[††]Assistant Professor of Computer Engineering, MIT, Pune INDIA
[†††]Professor and Dean of Research, MITSOT, MAE, Pune INDIA
[††††]Assistant Professor, VESIT Chembur, Mumbai INDIA
[†††††]Faculty of Computer Engineering, MITCOE, Pune INDIA

Abstract
The detection of fraudulent transactions in credit card world is an important application of classification techniques. As human behavior is unpredictable classifying any transaction either as fraud or non-fraud is not acceptable. In all of the previous studies, the transactions were classified in only two levels either fraud or legitimate. An approach where the credit card transactions can be classified in various fraud levels depending on different fraudulent situations mined from the historical behavior of the customers is proposed and implemented. To perform the classification, Efficient Scalable Multilevel Classifier (ESMC) algorithm is developed. Our algorithm is scalable to large dataset and results shows better accuracy than the previous algorithm implemented for credit card fraud detection. The concept of multi-dimensional decision tree is introduced to have better scalability in the field of distributed databases.

Index Terms
Distributed data Mining, Decision Tree, Pruning, Intrusion Detection.

I. INTRODUCTION

A secured and trusted inter-banking network [1], [2] for electronic commerce requires high speed verification and authentication mechanisms that allow legitimate users easy access to conduct their business, while thwarting fraudulent transaction attempts by others. Fraudulent electronic transactions are already a significant problem, one that will grow in importance as the number of access points in the nation's financial information system grows.
Financial institutions today typically develop custom fraud detection systems targeted to their own asset bases. Most of these systems employ some machine learning and statistical analysis algorithms to produce pattern-directed inference systems [1]. They use models of anomalous or errant transaction behaviors to forewarn of impending threats.
These algorithms require analysis of large and inherently distributed databases of information about transaction behaviors to produce models of "probably fraudulent" transactions. Recently banks have come to realize that a unified, global approach is required, involving the periodic sharing of information about attacks with each other. Such information sharing is the basis of building a global fraud detection infrastructure where local detection systems propagate attack information to each other, thus preventing intruders from disabling the global financial network.
As credit card transactions continue to grow in number, taking an ever-larger share of the country's banking system and leading to a higher rate of stolen account numbers and subsequent losses by banks [3], improved fraud detection thus has become essential to maintain the viability of the banking system. Large-scale data-mining techniques [4] can improve on the state of the art in commercial practice. Scalable techniques to analyze massive amounts of transaction data that efficiently compute fraud detectors in a timely manner is an important problem, especially for e-commerce. Besides scalability and efficiency, the fraud-detection task exhibits technical problems that include skewed distributions of training data [2], [5] and non-uniform cost per error, both of which have not been widely studied in the knowledge-discovery and data mining community.
The system approach addresses the efficiency and scalability issues in several ways. In this a large data set of labeled transactions (either fraudulent or legitimate) is divided into smaller subsets and mining techniques have been applied to generate classifiers in parallel which will be combined to take global decisions. Section 2 reviews some of the related work carried out on decision tree classification algorithms. Section 3 depicts overall architecture of distributed fraud detection system. Section 4 gives implementation details and the description of new ESMC algorithm. Section 5 shows comparative results of the ESMC and C4.5 algorithms on the developed system. Section 6 concludes the paper and draws direction to future work.

Manuscript received August 5, 2010
Manuscript revised August 20, 2010

II. RELATED WORK

Databases are rich with hidden information that can be used for making intelligent business decisions. Classification and predictions are two forms of data analysis that can be used to extract models describing important data classes or to predict future data trends.

Decision tree generation consists of two phases. First phase is tree construction phase in which in the beginning all the training examples are at the root and then examples are partitioned recursively based on selected attributes. The second phase is tree pruning [6], in which the branches that reflect noise or outliers are identified and removed. Different algorithms for decision tree induction [7] differ from each other in terms of the criterion that is used to evaluate the splits that correspond to tests on different candidate attributes. The choice of the attribute at each node of the decision tree greedily maximizes (or minimizes) the chosen splitting criterion.

The well-known CART (Classification And Regression Trees) [8] and C4.5 (Classifier 4.5) [10] classifiers, for example, grow trees depth-first and repeatedly sort the data at every node of the tree to arrive at the best splits for numeric attributes. SLIQ (Supervised Learning In Quest) [11], on the other hand, replaces this repeated sorting with one-time sort by using separate lists for each attribute. SLIQ uses a data structure called a class list which must remain memory resident at all times. The size of this structure is proportional to the number of input records, and this is what limits the number of input records that SLIQ can handle. SPRINT (Scalable PaRallelizable Induction of decision Trees) [12] shares with SLIQ the advantage of a one-time sort, avoids repetitive sorting like in CART and C4.5. SPRINT doesn't use structure like the class list that grows with the size of input and needs to be memory-resident and provide facility of handling large data sets.

Meta decision trees (MDTs) [13], is a novel approach to combine multiple models. Instead of giving the prediction, this method gives which model to be used for prediction. An algorithm for learning MDTs based C4.5 learning algorithm is presented in [13]. The task of constructing ensembles of classifiers is broken down into two sub-tasks. First to generate diverse set of base level classifiers and once the base level classifiers have been generated then the issue arise to combine their predictions which is handled in [13]. Techniques for combining the predictions obtained from the multiple base-level classifiers can be clustered in three combing frameworks: voting stacked generalization or stacking and cascading. The work presented in [13] focuses on combining the predictions of base-level classifiers induced by applying different learning algorithms to single data set. It adopts the stacking framework, and introduces a notion of meta-decision trees. The difference between ordinary decision trees and MDT is that MDT leaves specify which base-level classifier be used, instead of predicting the class value directly.

Automated credit card fraud detection by means of machine learning is discussed in [14]. In the world of digitization, credit card fraud detection is of great importance to financial institutions. Two machine learning techniques suited for reasoning under uncertainty: artificial neural networks and Bayesian belief networks have been applied to the problem and their significant results on real world financial data are shown in [14]. The paper also discusses the problem of identifying or detecting fraudulent behavior in credit card transaction system.

III. OVERALL ARCHITECTURE

The Overall Architecture of Distributed Fraud Detection System is shown in Fig.1. Data Sites are the local data stores where the local classifiers will be derived. Classification engine acts as a re-combination agent to build a unified global classifier. The detection engine enables one to identify the fraudulent behavior of every input. The detection engine can be based on an online model working in real time, or in an offline manner working on stored, human assisted manner.

The system has two key component technologies: local fraud detection agents that learn how to detect fraud and provide intrusion detection services within a single corporate information system, and a secure integrated system that combines the collective knowledge acquired by individual local agents. The detailed architecture of the system is explained below.

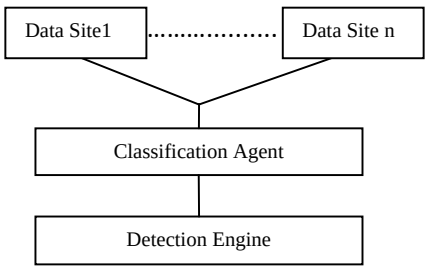

Figure 1: Overall Architecture of Distributed Fraud Detection System

a. Data Cleansing module

The input data given by us to the classifier for learning is in the form of credit card transactions. The transactions used for training are comprised of fields having current transaction data combined with attributes representing some historical information about credit card customer behavior. Due to privacy constraints bank has provided

only summarized information of the credit card database. Rearrangement of information and cleaning of data is done. Data cleaning means only the attributes giving information about fraud situations are picked up. Many of the attributes having continuous values are discretized for the implementation purpose. The ranges of the different attribute values were also decided in our work. The data received from the bank includes card holder's profile, its personal, educational and economical status, and purchase profile giving review of its purchasing behavior within a year.

b. Data Site Architecture

As the system deals with distributed environment, there are different local sites having their on local credit card database of particular branch. The architecture of the data site is depicted in Fig 2. Each data site is deployed with the classification model having decision tree learning and classification algorithm. The classifier is trained with the local database to form the decision rules. These rules are then used for further classification.

c. Global Classification module

There can be various fraudulent situations and may be local to particular branch. So if any outlier behavior takes place at a particular branch it will go undetected due to local classifier having rules related to local database. To overcome this situation global classification module is developed where local rules are integrated to form multidimensional decision tree, which is shown in Fig. 3.

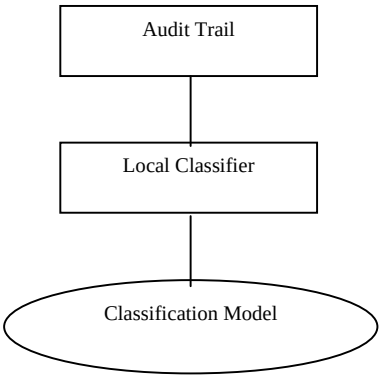

Figure 2: Data Site architecture

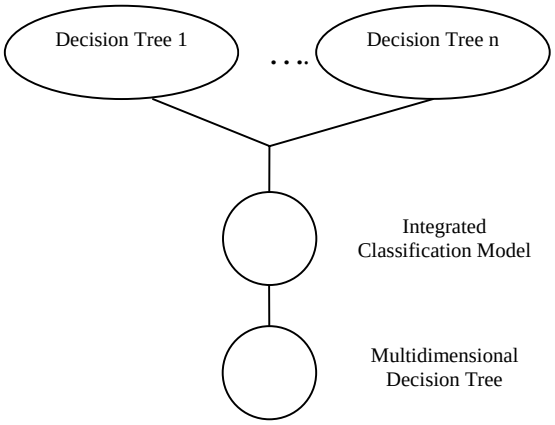

Figure 3: Global Classification Module

d. Detection Engine Architecture

After development of classification model, detection engine (Fig. 4) is used to classify current credit card transaction and returns the class level of the given transaction.

Here, detection engine works as front end where the transactions are given as input, the decision rules formed will be applied to classify the given input behavior in four different levels out of which level1 is normal and rest three are suspicious.

e. Tree Pruning

With decision tree classification algorithm, when decision tree is formed sometimes it happens that it generates some unwanted and meaningless rules as it grows deeper, it is called as over-fitting [15]. This can be avoided by only considering those attributes which will have big contribution in forming the particular rule. This is done by stopping the growth of decision tree at particular level so that the rules formed give better classification.

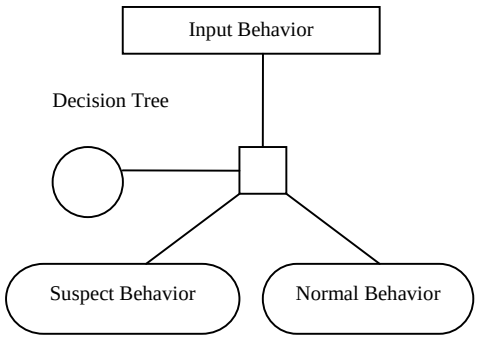

Figure 4: Detection Engine Architecture

Pre-pruning [15], [16] method is applied in the developed application. In this, the growth of decision tree is stopped at particular level and the remaining transactions at that particular split node are assigned with a most frequent class among these transactions.

IV. IMPLEMENTATION

a. Credit Card Database

The credit card database used for training and classification is developed based on the snapshot of the credit card database given by the bank. For security purpose the bank did not allow us to reveal the real data, due to this the database was manually preprocessed from the given information and the overall survey of the credit card world. The credit card transaction table built for learning contains 101580 records.

The transaction table is built based on the current transaction information such as amount, transaction time, transaction location, expiry date entered, card limit, in addition to that some historical information is also combined with these fields like average purchase of previous three months, average purchase of previous twelve months, customer's preferred transaction location and time, limit of number of transactions within a day to trace the customer's normal behavior. The transaction record does not contain customer account number because instead of learning, behavior models of individual customer accounts, overall models that try to differentiate legitimate transactions from fraudulent ones is built. So the model is customer-independent.

b. Types of Fraud

Instead of classifying the given transactions in only two types that is either fraud or non-fraud, in the system implemented transaction gets classified in four different types of class levels (L1, L2, L3, L4) which are decided based on different fraudulent situations traced out from given snapshot of database by bank and survey done on credit card world. The fraudulent situations based on which class levels have been assigned to the transactions.

c. Decision Tree Induction Algorithm

For the local classification Efficient Scalable Multilevel Classifier (ESMC) algorithm is implemented with added features of C4.5 algorithm. Also ESMC is combined with pre-pruning to increase the accuracy of classification.

ESMC Algorithm with pre-pruning

a) Construct the tree in a top-down recursive divide-and
-conquer manner.
b) In the beginning, keep all the training examples at the root.
c) Attributes are considered to be categorical (if continuous-valued, they are discretized in advance).
d) Partition examples recursively based on selected attributes.
e) Select the splitting attribute on the basis of entropy measure.
f) Repeat all the steps until one of the three conditions get satisfied:
   i. All samples for a given node belong to the same class.
   ii. There are no remaining attributes for further partitioning.
   iii. There are no samples left.
   iv. Set prune level is reached.

Entropy Measure

Entropy measure is given by following equation. For a set of record S,

$$\textbf{Entropy } E(S) = -\Sigma\, p_j \log p_j \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (1)$$

Where, $j = 1, 2, \ldots, m$

p_j is the relative frequency of class j in S

Entropy divides S with n records in two sets, S1 with n1 records and S2 with n2 records.

$$E(S1,S2) = \frac{n_1}{n} E(S1) + \frac{n_2}{n} E(S2) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2)$$

In the context of decision trees, if the outcome of a node is to classify the records into two classes, C1 and C2, the outcome can be viewed as message that is being generated and the entropy gives the measure of information for a message to be C1 or C2. If a set of records T is partitioned into a set of disjoint exhaustive classes C1,C2,…., Cn on the basis of a value of the class attribute, then the information needed to identify the class of an element of T is

$$\textbf{Info (T) = Entropy (P)} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3)$$

Where, P is probability distribution of the partition C1, C2, …, Cn.

P is computed based on their relative frequencies, that is,

$$\textbf{P = ((|C1|/|T|, |C2|/|T| , …|Cn|/|T|)} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4)$$

The goal is to lower the Entropy.

d. Classification Algorithm

There are two phases in decision tree classification, first is to generate the decision tree from the given training data and second is actual classification where decision rules of formed decision tree is applied to the transaction having unknown class label to classify it in one of the classes. The algorithm for this classification is given below:

1. For each transaction to be classified, read one by one the decision rule from the Decision table.
2. Match the fields from the transaction with each decision rule. (Fields having blank entries in decision table indicate don't care condition).
3. First try to find out perfect match and fill the Class field of the transaction with the class of matched rule.
4. If perfect match is not found then voting method is applied, where best match is found based on maximum match count and the class field of the transaction is filled with the class of best matched rule.

V. RESULTS

a. Data Sets

The data used in this paper is real world data which is provided by a nationalized bank. As the data contains sensitive information, the database cannot be revealed as per the agreement with the bank. Around 1 lac credit card transactions are used based on the different fraudulent cases. The transactions are then divided into different test sets.

The classifier is trained with different transaction sets and used for the classification of each of these sets. For comparison purpose basic C4.5 algorithm and ESMC algorithm are used for training. As classes of these transactions are already known, the classification accuracy is evaluated by comparing the classified transactions with the original class value of the transactions. Classification measures used for results evaluation are True Positive Rate (TPR), False Positive Rate (FPR), and Accuracy [16].

b. Local Classification Results

From the whole transaction set, some transactions are taken for training and part of it are taken for testing purpose and then this procedure is repeated for the whole transaction database.

The results evaluated with both the training algorithms (C4.5 and ESMC) are listed and compared.

Table 1: Specification of each transaction set

Test Set Name	No. Of Transactions
Test1	10000
Test2	20000
Test3	30000
Test4	40000
Test5	10000
Test6	20000
Test7	30000
Test8	40000
Set 1	50780
Set 2	50780
Main Set	101560

1) Comparison of Accuracy and Classification measures with Main_set as training file is shown:

Figure 5 shows that ESMC gives average 87% of TPR whereas C4.5 gives 82% of TPR.

Figure 6 gives comparative results of False Positive Rate with C4.5 and ESMC algorithm.

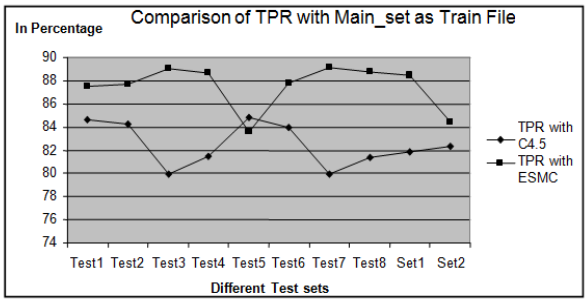

Figure 5: Comparison of True Positive Rate (TPR)

Results show that ESMC gives lower FPR around average 12% than C4.5 which gives average FPR of 30%.

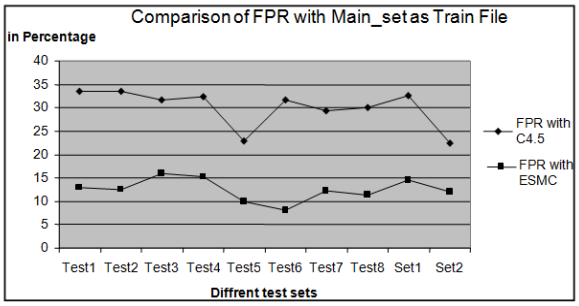

Figure 6: Comparison of False Positive Rate

Figure 7 represents results of overall accuracy evaluation. Overall accuracy of each transaction getting classified to the correct class level is very important for fulfilling the objective of the system. Considering the main set as base classifier and classifying different data sets ESMC gives highest average accuracy of 80% which is much better than C4.5 giving overall accuracy of 62%.

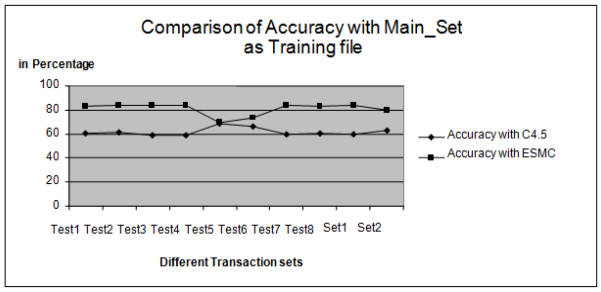

Figure 7: Comparison of Overall Accuracy

2) Level wise comparison of accuracy with training file (main set)

Figure 8 depicts results of evaluated accuracy for class type L1 with C4.5 and ESMC algorithm.

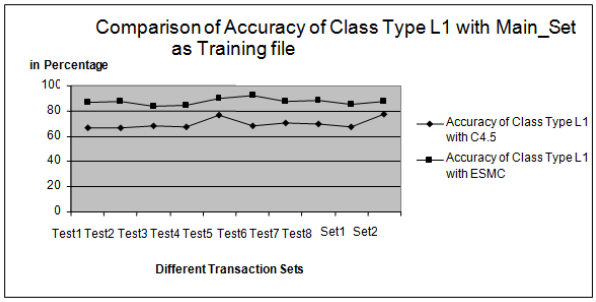

Figure 8: Accuracy Evaluation of Class Type L1

Figure 9 depicts results of evaluated accuracy for class type L2 with C4.5 and ESMC algorithm.

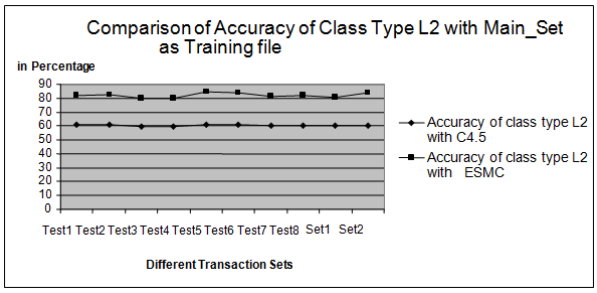

Figure 9: Accuracy evaluation of Class Type L2

Figure 10 shows results of evaluated accuracy for class type L3 with C4.5 and ESMC algorithm.

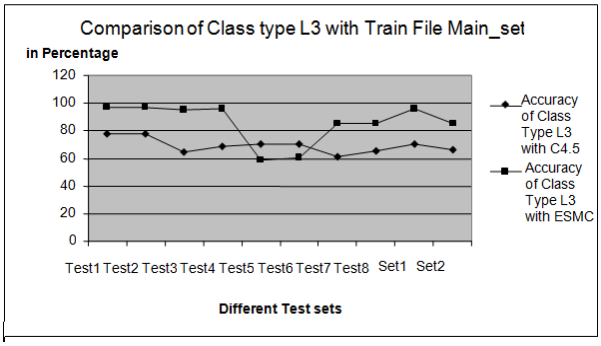

Figure 10: Comparison of Accuracy of Class L3

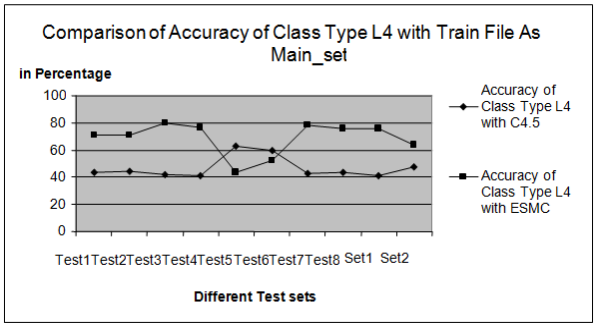

Figure 11: Comparison of Accuracy of Class L4

Figure 11 gives results of evaluated accuracy for class type L4 with C4.5 and ESMC algorithm.

Observations show that ESMC performs well in all the comparisons than C4.5 algorithm. Also accuracy of each class type with ESMC is better than C4.5.

c. Global Classification Results

Global classification is done based on combining rules formed using different training files. The rules are formed by either applying C4.5 algorithm or ESMC algorithm on all the training files.

Table 2: Global Classification Results with Main_set as test file

Measures	Decision table formed using ESMC and C4.5 algorithm	
	Global Decision rules using ESMC	Global Decision rules using C4.5
L1	87.01	30.29
L2	80.75	23.59
L3	80.73	21.57
L4	80.18	17.96
FPR	12.99	69.71
FNR	14.05	05.66
TPR	85.95	94.34
Accuracy	82.63	24.00

Then these global rules are used to test the classification of the whole dataset. Table 2 shows the classification results where ESMC provides very much higher accuracy of 82.63% as compared to C4.5 giving accuracy of just 24%. It has been seen that TPR rate is higher in case of C4.5 but the algorithm is not capable of doing level vise classification.

VI. CONCLUSION

Credit card fraud detection system is one of the applications of ESMC which has been developed. The application is useful for inter-banking where banks can share their fraud detecting rules with each other to overcome the threat of fraud which is spreading widely in world of credit cards. In contrast to previously developed credit card fraud detection systems where transactions were getting classified in only two levels either fraud or non-fraud the system developed can differentiate among different fraudulent situations and classifier transactions in four levels where level wise fraud risk increases.

The performance based on Accuracy and True Positive Rate is compared between basic C4.5 algorithm and newly developed ESMC algorithm. Different transaction sets are formed. The classifier is then trained with these different sets and accuracy is evaluated by classifying sets with these different decision trees. ESMC gives on average 80% accuracy whereas C4.5 algorithm gives on an average 62% accuracy. Fraud catching rate (TPR) of both the classifiers is 85%. False Alarm rate (FPR) of ESMC is 12% and ESMC gives False alarm rate of 30%. ESMC algorithm is decision tree learning algorithm with pruning. Observation shows that at level 4 it gives highest accuracy for different transaction sets of the application. The classification does not consider the customer ID and thus it gives customer independent classification.

Scalability is one of the features provided by the system where database is spread across the network and only decision table (small in size) is used to classify them. Number of decision rules generated by ESMC is much lesser than rules generated by C4.5 algorithm. Many times rules generated by C4.5 are redundant and meaningless. In ESMC algorithm, rules are lesser which directly affects the size of the decision table which reduces the time required to perform classification of large number of transactions. As size of the decision table generated with ESMC algorithm is small, the required network bandwidth while transferring the decision table through agents also reduces.

Levels of fraud can be increased to differentiate various fraud situations. The time behavior of the current system with increased database size has not been studied. Our ongoing work is focused on the effectiveness of the system for real time application.

REFERENCES

[1] Salvatore, Philip et al., “Meta learning agents for fraud and intrusion detection in Financial Information Systems.”, Inv paper Proceedings in International conference of Knowledge Discovery and Data mining, 1996.

[2] S. Stolfo et al., “JAM: Java Agents for Metalearning over Distributed Databases,” Proc. Third Int’l Conf. Knowledge Discovery and Data Mining, AAAI Press, Menlo Park,Calif., 1997, pp. 74–81.

[3] Philip K. Chan, Wei Fan, Andreas L. Prodromidis, and Salvatore J. Stolfo, “Distributed Data Mining in Credit Card Fraud Detection”, Proc.IEEE Int’l Conf. Intelligent Systems, Dec.1999.

[4] Jiawei Han, Micheline Kamber, “Data Mining Concepts and Techniques”, pp. 279-328, 2001.

[5] Salvatore J. Stolfo, David W. Fan, Wenke Lee and Andreas L. Prodromidis, “Credit Card Fraud Detection Using Meta-Learning: Issues and Initial Results”, DARPA, 1999.

[6] Zhang Yong, “Decision Tree’s Pruning Algorithm Based on Deficient Data Sets”, In Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies, 2005.

[7] Sasikiran Kandula, “A Comparative Study Of Classification Algorithms”, Thesis, May 2005.

[8] L. Breiman, J. H. Friedman, R. A. Olshen, and C.J. Stone (1984), “Classification and Regression Trees”, Wadsworth, Belmont, CA, 1984.

[9] J.R. Quinlan, “Induction of Decision Trees, in Machine Learning”, 106-181, 1986.

[10] Manish Mehta, Rakesh Agrawal et al:, “SLIQ- A Fast Scalable Classifier for Data Mining”, In 5th Intl. Conf. on Extending Database Technology, March 1996.

[11] John Shafer, Rakesh Agarwal, e al:, “ SPRINT- A Scalable Parallel Classifier for Data Mining”, Proceedings of 22nd International VLDB conference, 1996.

[12] Todorovski L, Dzeroski S., “Combining Multiple Models with Meta Decision Trees”, In Proceedings of the Fourth European Conference on Principles of Data Mining and Knowledge Discovery. Springer, 2000 and Machine Learning Kluwer Publishers 2003.

[13] Sam Mayes, Karl Tuyls et al:, “Credit card fraud detection using Bayesian and Neural networks.”, Proceedings in International Conference in KDD 2003.

[14] Zhang Yong, “Decision Tree’s Pruning Algorithm Based on Deficient Data Sets”, In Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies, 2005.

[15] Arun Poojari, “Data Mining techniques”, pp 150 -200, 1999.

[16] Szappanos Tibor, Zolotova Iveta, “Distributed Data Mining and Data Warehouse”, ASR '2005 Seminar, Instruments and Control, Ostrava, April 29, 2005.

Dipti D.Patil has received B.E. degree in Computer Engineering from Mumbai University in 2002 and M.E. degree in Computer Engineering from Mumbai University, India in 2008. She has worked as Head & Assistant Professor in Computer Engineering Department in Vidyavardhini's College of Engineering & Technology, Vasai. She is currently working as Assistant Professor in MITCOE, Pune. Her Research interests include Data mining and Business Intelligence.

Sunita M.Karad has received B.E. degree in Computer Engineering from Marathvada University, India in 1992**,** M.E. degree from Pune University in 2007. She is a registered Ph.D. student of Amravati University. She is currently working as Assistant Professor in Computer Engineering department in MIT, Pune. She has more than 10 years of teaching experience and successfully handles administrative work in MIT, Pune. Her research interest includes Data mining, Business Intelligence & Aeronautical space research.

Dr. Vijay M.Wadhai received his B.E. from Nagpur University in 1986, M.E. from Gulbarga University in 1995 and Ph.D. degree from Amravati University in 2007. He has experience of 24 years which includes both academic (17 years) and research (7 years). He has been working as a Dean of Research, MITSOT, MAE, Pune (from 2009) and simultaneously handling the post of Director - Research and Development, Intelligent Radio Frequency (IRF) Group, Pune (from 2009). He is currently guiding 12 students for their PhD work in both Computers and Electronics & Telecommunication area. His research interest includes Deductive Databases, Knowledge Discovery and Data Mining, Data Security, Natural Language processing, Cognitive Radio and Wireless Communication, Spectrum Management, Wireless Sensor Network, ASIC Design - VLSI, Advance Network Design. He is a member of LMISTE, MIETE, MIEEE, MIES and GISFI (Member Convergence Group), India.

J A Gokhale received his B.E. in Electronics and Communications and M.Tech in Computers. He is currently working as a Professor and Head of Computer Engineering Department at VESIT and also a member of the R&D wing of the Institute. He has an overall experience of 25 years in the academics and 6 years in Industry. He has handled various research & Industry Projects and is also a consultant to various IT Organizations in India and abroad. He has also authored several papers at national and International level. His areas of interest include Data Mining, Embedded Systems and Algorithms.

Prasad S.Halgaonkar received his bachelor's degree in Computer Science from Amravati University in 2006 and M.Tech in Computer Science from Walchand College of Engineering, Shivaji University in 2010. He is currently a lecturer in MITCOE, Pune. His current research interest includes Knowledge discovery and Data Mining, deductive databases, Web databases and semi-structured data.