academicJournals

Vol. 10(1), pp. 5-13, June 2017
DOI: 10.5897/AJMCSR2017.0686
Article Number: 8D3A10364626
ISSN 2006-9731

http://www.academicjournals.org/AJMCSR

African Journal of Mathematics and Computer Science Research

Full Length Research Paper

A multi-algorithm data mining classification approach for bank fraudulent transactions

Oluwafolake Ayano* and Solomon O. Akinola

Department of Computer Science, University of Ibadan, Nigeria.

Received 22 February, 2017; Accepted 19 April, 2017

This paper proposes a multi-algorithm strategy for card fraud detection. Various techniques in data mining have been used to develop fraud detection models; it was however observed that existing works produced outputs with false positives that wrongly classified legitimate transactions as fraudulent in some instances; thereby raising false alarms, mismanaged resources and forfeit customers' trust. This work was therefore designed to develop a hybridized model using an existing technique Density-Based Spatial Clustering of Applications with Noise (DBSCAN) combined with a rule base algorithm to reinforce the accuracy of the existing technique. The DBSCAN algorithm combined with Rule base algorithm gave a better card fraud prediction accuracy over the existing DBSCAN algorithm when used alone.

Key words: Card fraud detection, density-based spatial clustering of applications with noise (DBSCAN), rule base algorithm, data mining.

INTRODUCTION

Card fraud is one of the biggest threats to organizations today. Card fraud is simply defined as unauthorized, deliberate deception to secure unfair or unlawful access to a victim's transaction card in order to defraud him (Salem, 2012).

A fraud detection system usually comes to play when the fraudsters outwit the fraud prevention mechanism and initiate fraudulent transactions. In the business world, the application of data mining technique to fraud detection is of special interest as a result of the great losses companies suffer due to such fraudulent activities. This work describes data mining technique and its application to card fraud detection.

Fraud detection notion is based on data mining techniques and principles. One of such techniques is classification. Although existing works have proved to reduce fraud, many of the transactions labeled as fraudulent are actually legitimate. This mismatch has resulted in huge loss of money, wasting of time that can be used to examine real fraud cases and cause customer dis-satisfaction in the sense that the legitimate transactions are being delayed and customers are bothered with lots of false alarms.

It is quite obvious that multi-algorithm, that is, using different possible combinations, can be a strong combination of soft computing paradigm, this explained why there has been researches and application to many different problem domains. A domain that is conspicuously omitted is the card fraud detection. It is assumed that Density-Based Spatial Clustering of

*Corresponding author. E-mail: flakkydoja@yahoo.com.

Applications with Noise (DBSAN)-Rule Base combination should be able to perform very well. This assumption motivated this research work in order to explore DBSCAN-Rule Base combination to develop a card fraud detector.

This study presented a hybridized model that makes use of an existing algorithm (DBSCAN) to group transactions into several clusters and then enhance the output of the clustering with a rule base algorithm in order to characterize the transactions as fraudulent or otherwise. The proposed model enhances the accuracy of the existing system.

RELATED WORKS

Srivastava et al. (2008) present a credit card fraud detection system using the Hidden Markov Model (HMM). The researchers trained the HMM with the normal pattern of a customer and the incoming transaction is considered as illegitimate if it does not resemble the normal pattern the HMM was trained with. Abdelahlim and Traore (2009) designed a fraud detection system using Decision Tree to solve the problem of application fraud.

Ogwueleka (2011) presented a Credit Card Fraud (CCF) detection model using Neural Network technique. The self-organizing map neural network (SOMNN) technique was applied to solve the problem of carrying out optimal classification of each transaction into its associated group since the output is not pre-determined.

Fraud Miner was proposed by Seeja and Masoumeh (2014). It is a credit card fraud detection model for detecting fraud from highly imbalanced and anonymous credit card transaction dataset. Frequent item set mining was used to handle the class imbalance problem thereby finding legal and illegal transaction patterns for each customer. A matching algorithm is then used to determine the pattern of an incoming transaction whether legal or illegal. The evaluation of Fraud Miner confirmed that it was able to detect fraudulent transaction and improve imbalance classification.

Sevda and Mohammad (2015) developed a model that can detect fraud in financial credit using real data. They used decision tree algorithm and neural network technique. The model clusters clients based on client type. That is, each cluster represents a client type. The model determines an appropriate rule for each cluster using the behaviour of the group members.

Keerthi et al. (2015) proposed a model using Neural Network technique. The self-organizing map neural network (an unsupervised method of AI) was used to cluster credit card transactions using four clusters of low, high, risk and high-risk clusters. If a transaction is legitimate, it was processed immediately. Fraudulent transactions are logged in the database but are not processed.

DBSCAN is an acronym for Density-Based Spatial Clustering of Applications with Noise. It is a density-based spatial clustering algorithm that identifies the dense regions in dataset based on density. Usually, the density of an object say x is measured by the number of objects that are close to x. DBSCAN identifies the core objects that have dense neighbourhoods. It requires two user-defined parameters, which are neighborhood distance *epsilon* (eps) and minimum number of points *minpts*. These parameters are difficult to determine especially when dealing with real world high dimension dataset.

For a given point, the points in the *eps* distance are called neighbours of that point. If the number of neighbouring points of a point is more than *minpts,* this group of points is called a cluster. DBSCAN labels the data points as core points, border points, and outlier (anomalous) points. Core points are those that have at least *minpts* number of points in the *eps* distance. Border points can be defined as points that are not core points, but are the neighbours of core points. Outlier points are those that are neither core points nor border points (Sander et al., 1998; Ajiboye et al., 2015).

These core objects and their neighbourhoods are connected to form group of dense regions called clusters. DBSCAN uses the Euclidean distance metrics to determine which instances belong together in a cluster. There is no need to specify the number of clusters as expected in other techniques like K-means; DBSCAN clusters data automatically, identifies arbitrarily shaped clusters and incorporates a notion of anomaly (Witten et al., 2011; Salganicoff, 1993).

PROPOSED SYSTEM

Data source and nature

A real banking dataset was obtained from a financial institution in Nigeria. The dataset used in this study consists of some card transactions received in a period of six months from July to December 2015.

Data cleaning

In data mining, data cleaning is an important step as it eliminates noisy data and performs data normalization.

The dataset consists of some card transactions received in a period of six months July to December 2015. The dataset consists of 1,356,243 records from 813 cards. The following steps were taken to clean up the dataset.

Cards with less than 3 months transactions were removed as they will not provide enough information for the study. Cards with inactive status were also separated as such cards will only allow inflow but no outflow; therefore, the chances of fraud on such cards are limited. Debit transactions were identified. Transactions in this category include bill payments and purchase transactions. From the dataset, it was discovered that some customers had just one transaction in the period under review; such customers were removed from the dataset as there is no way a pattern can be established from just one transaction. Transactions that did not have complete information were also filtered and ensured that only the transactions that were settled, not reversed and have impacted on the banking host were used. After the data cleaning exercise, 1,000,023 records in the dataset remained useful.

DBSCAN

DBSCAN is the preferred algorithm for this study because it has some special attributes that are suitable for the task.

(1) It has the capability to process very large database
(2) The number of clusters is not predetermined
(3) It can find clusters with subjective shapes.

However, DSCAN has its own limitations, which include its inability to handle temporal data and false positives; hence, the need to use the modified version of DBSCAN that can handle the nature of card transactions. The DBSCAN Algorithm is presented in pseudocode, thus (Source Wikipedia, 2015):

```
DBSCAN(D, eps, MinPts) {
  C = 0
  for each point P in
    dataset D {
    if P is visited
```

```
  continue next point
     mark P as visited
     NeighborPts =

 regionQuery(P, eps)
     if sizeof(NeighborPts)
       < MinPts
       mark P as NOISE
     else {
       C = next cluster
       expandCluster(P,
       NeighborPts, C, eps,
       MinPts)
     }
   }
}

expandCluster(P, NeighborPts,
   C, eps, MinPts) {
   add P to cluster C
   for each point P' in
      NeighborPts {
      if P' is not visited {
        mark P' as visited
        NeighborPts' =
 regionQuery(P', eps)
 if sizeof(NeighborPts') >=
    MinPts
    NeighborPts = NeighborPts
    joined with NeighborPts'
     }
 if P' is not yet member of any cluster add P' to cluster C
   }
}

regionQuery(P, eps)
   return all points within
   P's eps-neighborhood
   (including P)
```

The rule base algorithm

In many real world applications, data contains uncertainty as a result of various causes which could include measurement and decision error. Since uncertainty is commonplace, there is need to develop algorithm to handle such occurrences. A rule base classifier is a technique for classifying records using a collection of "IF …THEN…" rules. The IF part of the rule is referred to as the Rule Antecedent/Precondition. It is made up of one or more tests that are logically ANDed and the THEN part is called Rule/Consequent and it consists of class prediction. The rule algorithm has the rule extraction and rule pruning (Nobel, 2015). This work is focused on building a novel rule base classification algorithm. A way of generating rules was proposed and same was applied to real financial institution's card transactions. Set of rules that demonstrate the relationship between the features of our dataset and the class label was extracted. One rule set can have multiple rules defined, that is, Rs = {Ri…..Rn}. The rule is pruned by removing conjunct, which will increase the accuracy of the rules on the pruning set.

The rule base algorithm is a set of rules put together to further prune the result of the DBSCAN to overcome the challenges raised on it, such that the algorithm was further strengthened and adapted for use. A new epsilon (eps) was introduced to the DBSCAN classifier to measure the temporal properties. Therefore, eps1 was used to measure the closeness of the transaction amount while eps2 measures the time elapsed between the transactions. To achieve this, the transactions were sorted first by keeping the temporal properties and then the spatial properties.

The new model solves the problem of false positives by passing the output of DBSCAN Classifier through the Rule Base Algorithm. The rule base algorithm traversed all the clusters applying the rules set to each element before it safely concludes that the transaction is actually legitimate or fraudulent. The rule base algorithm involves three main rules:

Rule 1: Transaction amount

Algorithm was developed for the transaction amount, the customer spending behavioural pattern was studied and the merchants' patronages were investigated. Maximum transaction amount was retrieved for a period of three months from the database for the customer and 200% of the maximum amount was computed. It is expected that a customer can still perform up to 200% of her maximum transactions. The outlier was checked to confirm if the transaction was above 200% more than the total outflow in the last three months.

Rule 2: Location

The location of the transaction was built into the logic of the rule base algorithm such that it verifies the customer country code with the transaction's country code. If the two are not the same, then it checks the time zone of the current transaction with the last transaction.

Rule 3: Channel

There are various channels of payments which include POS, ATM or WEB. If the channel of payment is either POS or ATM, it checks to confirm if the card had been reported stolen. If the channel is WEB, It checks if the billing address is different from the shipping address.

The Rule-Based Algorithm as developed for the research is expressed as follows in pseudocode:

```
RULEBASE(D- Database, Amt- Incoming transaction amount, t-
time of the transaction, loc- location, c-channel, P)
Output 0 – legitimate, 1 – fraudulent
  {
Begin
              channelRule(c){
                if c in
   (approvedChannels)then {
                    if status =
      'Active'then  {
                      locationRule(loc,t)
                    }
                    else
                            mark P as NOISE
                            Output = 1

                }
               }

locationRule(loc, t){
  K= 0;
  R = 0;
  if D(countryCode) <> loc then{
R = timediff(lastTransaction, IncomingTransaction);
K = ZoneTimeDiff
   (D(countryCode, loc);
```

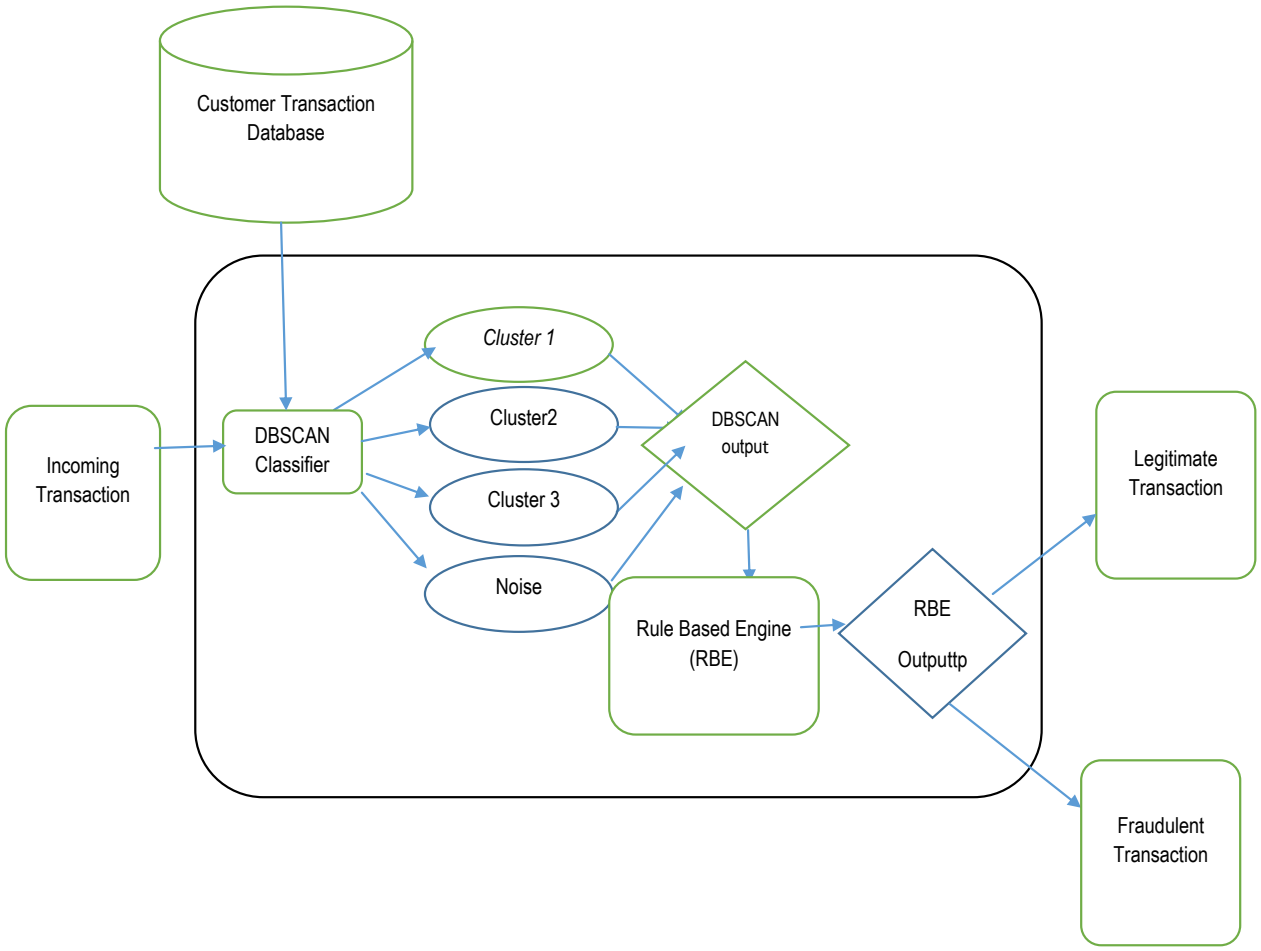

Figure 1. Architecture of the hybridised model.

```
If R > K then {
                amountRule(amt, t)
   }
 else

mark P as NOISE
   Output = 1

 }
}

amountRule(amt, t){
  K= 0;
  B = 0;
  Dateadd(dd, -90, t)
  K =  max D(amount) within
    Dateadd(dd, -90, t)
  B = (200 / 100) * K;
  If amt < K then {
      Add P to cluster
      Output = 0

 }
    else
      mark P as NOISE
      Output = 1
```

Architecture of the proposed system

The proposed model is a hybridized technique that combines DBSCAN classifier with rule-base algorithm to determine fraudulent transaction dynamically and reduce classification mismatch. Figure 1 shows the proposed fraud detection model.

An incoming transaction is fetched into the DBSCAN clustering system which also retrieves previous transactions for the customer for a period of three months from the database using the account number of the customer as a retrieval argument. The transactions for the customer are mined into different clusters using the Epsilon and minimum point defined. The classifiers look for a cluster closest to the new transaction and put it there. Otherwise, the new transaction is considered a noise. The output of the DBSCAN classifier is passed to the Rule Base Engine which further prunes the transaction using the rule defined in previously for processing. This is to ensure that the transaction is correctly labeled and improves the decision accuracy.

Implementation

The implementation was done on a PC with Windows Operating system. The hardware consists of 1.86 GHz Pentium Dual Intel Processor and a memory capacity of 2GB.

The implementation was done using Microsoft Visual Studio 2010

Table 1. DBSCAN result breakdown.

Dataset	Number of card	Number of transactions	Fraudulent transactions	TP	FP	FN
A	1	500	12	6	6	2
B	1	413	10	3	7	2
C	1	302	5	2	3	1
D	3	1215	17	8	9	5
E	213	10,000	30	8	22	7
F	304	10,500	24	6	18	9
G	320	20,000	14	7	7	10

Table 2. DBSCAN_Rule base result breakdown.

Dataset	Number of cards	Number of transactions	Fraudulent transactions	TP	FP	FN
A	1	500	8	6	2	1
B	1	413	5	3	2	1
C	1	302	2	2	0	0
D	3	1215	12	8	4	3
E	213	10,000	22	8	14	6
F	304	10,500	13	6	7	5
G	320	20,000	10	7	3	6

Table 3. Precision results.

Dataset	A	B	C	D	E	F	G
Dbscan	0.500	0.300	0.400	0.471	0.267	0.250	0.500
Dbscan-Rule base	0.750	0.600	1.000	0.667	0.364	0.462	0.700
Improvement (%)	50.00	100.00	150.00	41.67	36.36	84.62	40.00

with C# programming language. The data was warehoused in SQL Server 2008 Management studio.

Data sets

Due to the massive size of the original dataset, the dataset was broken into a number of smaller subsets in order to test the model. To test our model, seven datasets were prepared labeled A to G. The first 3 subsets of data labeled dataset A, B and C, respectively, contain transactions on one card. These subsets contained a mix of both legitimate and illegitimate transactions. These subsets were used to test the model for single customer cases to evaluate the model's performance from the specific transaction behaviour of a single customer.

Dataset D combines the datasets A, B and C into a single dataset. This is the smallest multiple card dataset. Datasets E, F and G contain 10000 transactions, each from several cards selected randomly within the period under review for the purpose of the test.

RESULTS

The performance of the proposed system was evaluated using Precision, Recall, F-Measure and Kappa Statistics. Precision, Recall and F-Measure were calculated using the result of a confusion matrix.

(a) Precision = TP/(TP +FP)
(b) Recall = TP/(TP +FN)
(c) F-Measure = 2 × [(Precision × Recall) / (Precision + Recall)]

Table 1 presents the breakdown of the DBSCAN Model results. It presents the number of transactions that are True Positives (TP), False Positives (FP) and False Negatives (FN), while Table 2 presents the breakdown of the DBSCAN-Rule Base Model results.

Comparison of the classifiers using precision

Precision measures the number of true positives divided by the number of true positives and false positives. In other words, precision is the measure of a classifier exactness. Table 3 presents the precision values of the DBSCAN and the combined DBSCAN-Rule based classifiers. It was observed that the DBSCAN has lower

Figure 2. Comparing DBSCAN and DBSCAN-RULE BASE using precision.

Table 4. Recall results

Dataset	A	B	C	D	E	F	G
Dbscan	0.750	0.600	0.667	0.615	0.533	0.400	0.412
Dbscan- Rule base	0.857	0.750	1.000	0.727	0.571	0.545	0.609
Improvement	14.29	25.00	50.00	18.18	7.14	36.36	30.77

precision values than the DBSCAN-Rule based classifier. A low precision indicates large number of false positives. It is therefore inferred that DBSCAN classifier has more non-fraudulent transactions labeled as fraudulent.

With the DBSCAN classifier combined with the rule base (DBSCAN-Rule base classifier), the number of false positives was reduced as seen in Figure 2. The percentage improvement was also presented in Table 3. Therefore, the DBSCAN-Rule base performed better in term of precision. The mean percentage improvement is 71.81%.

Comparison of the classifiers using recall

Recall measures the number of true positives divided by the number of true positives and the number of false negatives. In essence, recall can be thought of as a measure of the classifier completeness. A low recall indicates many false negatives. Table 4 and Figure 3 show the recall values of the DBSCAN and DBCAN-Rule base classifiers.

Comparison of the classifiers using F-Measure

The F-Measure indicates the balance between the recall and precision values. Table 5 shows the F-Measure values of the DBSCAN and the DBSCAN-Rule based classifiers. Figure 4 also compares the values of the two classifiers.

Comparison of the classifiers using Kappa statistics

Kappa statistics represent the extent to which the data collected correctly represents the variables measured. From Table 6 and Figure 5, it was observed that values

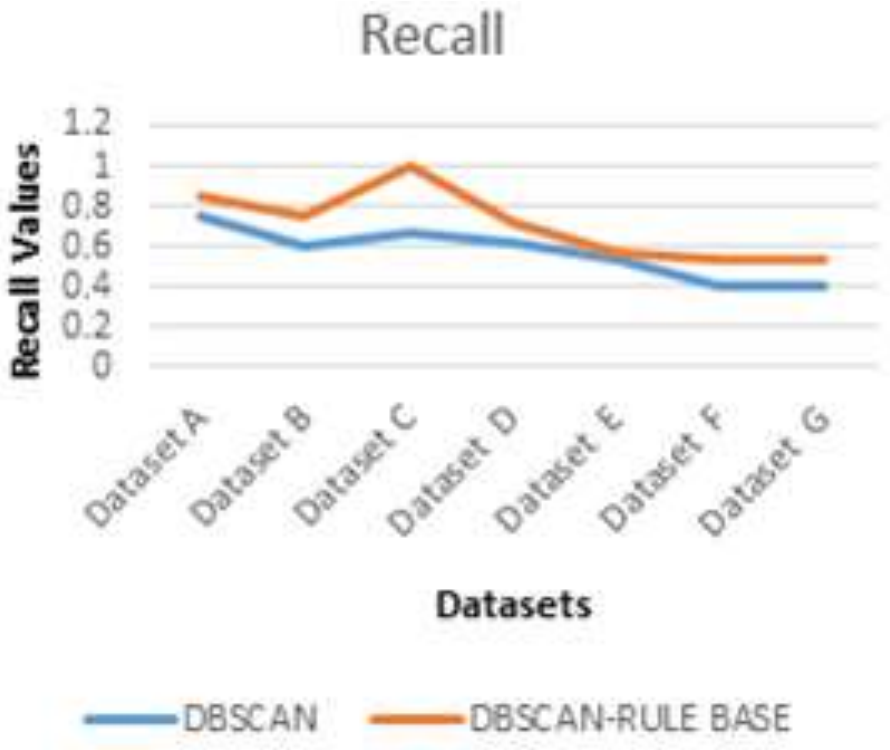

Figure 3. Comparing DBSCAN and DBSCAN-RULE BASE using recall.

Table 5. F-Measure results.

Dataset	A	B	C	D	E	F	G
Dbscan	0.600	0.400	0.500	0.533	0.356	0.308	0.452
Dbscan-Rule base	0.800	0.667	1.000	0.696	0.444	0.500	0.609
Improvement	33.33%	66.67%	100.00%	30.43%	24.99%	62.50%	34.78%

for DBSCAN-Rule based is closer to the Kappa statistics best value of 1 than the DBSCAN values in all instances which shows that the DBSCAN-Rule base has almost perfect agreement.

DISCUSSION

The best model was selected based on the comparisons and the research goal. The research aimed at detecting fraudulent transactions using multi-algorithm techniques to achieve higher accuracy. Therefore, the model needed to keep the number of TN very high and the FP rate as low as possible. Not much attention is paid to the FN as predicting failure (in this case, legitimate transactions) instead of success (fraudulent transactions) would do less harm to financial institutions.

With this in mind, the DBSCAN-Rule Base classifier is selected as the best predictive model for this study. It had higher classification Accuracy, Recall, Precision and F-Measure values in addition to these, its receiver operating characteristic curve (ROC) area which indicates the trade-off between TP Rate and FP Rate was also the best in comparison with the DBSCAN. Also, the number of FP in the DBSCAN-Rule Base model indicated in the confusion matrix was lower than the DBSCAN classifier.

The results obtained using the proposed DBSCAN-Rule Base model show that the hybridized model performed better than the single DBSCAN model as the number of transaction mismatches got reduced drastically. The result shows that the hybridized model has the tendency to perform better than a single model as it combines the strengths of the models used to come up with a better result. This is in line with researchers who undertook investigations into multi-algorithm models. Stolfo et al. (1997) opined that using multi-algorithm achieve higher accuracy over single algorithm. The results from the experiments showed great success in the implementation of a meta-learning classifier in the

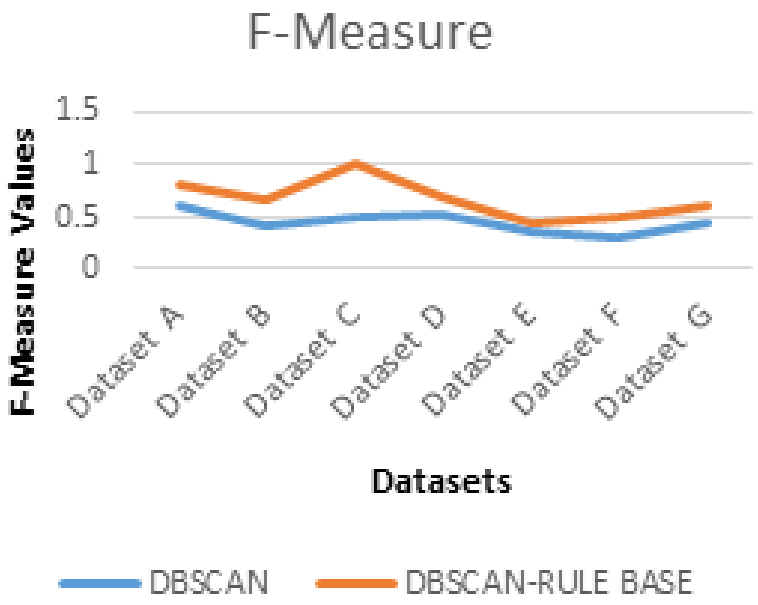

Figure 4. Comparing DBSCAN and DBSCAN-RULE BASE using F-Measure.

Table 6. Kappa statistics.

Dataset	A	B	C	D	E	F	G
Dbscan	0.592	0.390	0.494	0.528	0.354	0.306	0.451
Dbscan- Rule base	0.797	0.663	1.000	0.693	0.443	0.499	0.608
Improvement (%)	34.58	69.94	102.54	31.30	25.18	62.96	34.85

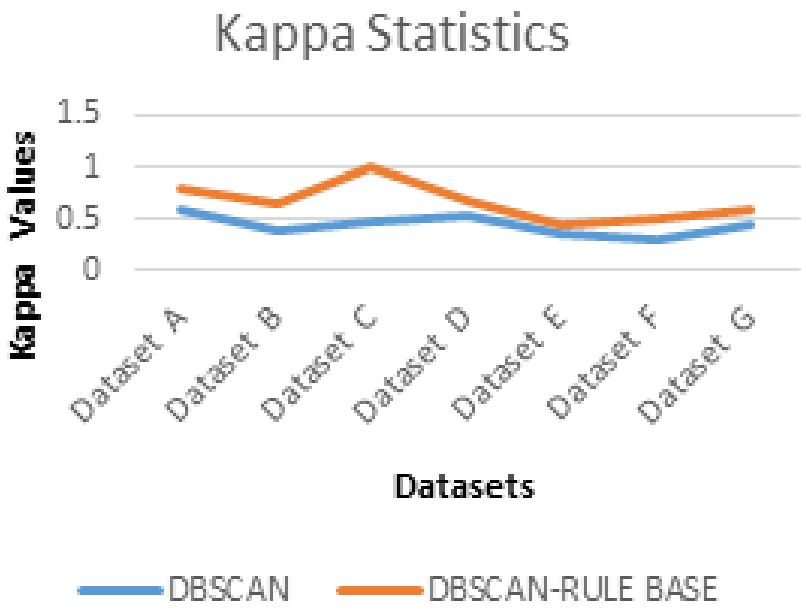

Figure 5. Comparing DBSCAN and DBSCAN-RULE BASE using Kappa statistics.

detection of credit card fraud. Saravanan and Babu (2013) combined meta-classifier

with Neural Network. The meta-classifier acts as a filter. The meta-classifier uses the predictions of different base classifiers to determine the final prediction of a transaction. This study claimed that no single learning algorithm can uniformly outperform other algorithms over all datasets.

Conclusion

The combined effect of DBSCAN and Rule base data mining prediction algorithms on detection of card fraudulent transactions in a is presented. The combined algorithms were demonstrated to be more effective in detecting or predicting card frauds than the single use of DBSCAN algorithm alone.

This research fills a gap in the current body of literature. Fraud card detection has not been tried with a combination of DBSCAN and RULE BASE before.

This research has made some basic discoveries and contributions to the field. To provide more conclusive and wider evidence of the usefulness of Multi algorithm in credit card fraud detection and eventually designing a functioning knowledge base system based on the findings, more research efforts are required.

CONFLICT OF INTERESTS

The authors have not declared any conflict of interest.

REFERENCES

Abdelahlim A, Traore I (2009). Identity application fraud detection using web mining and rule-based decision tree. Int. J. Netw. Comput. Secur. 1(1):31-44.

Ajiboye AR, Abdul-Hadi J, Akintola AG, Ameen AO (2015). Anomaly Detection in Dataset for Improved Model Accuracy Using DBSCAN Clustering Algorithm. Afr. J. Comp ICTs. 8(1):39-46.

Keerthi A, Remya MS, Nitha L. (2015). Detection of Credit Card Fraud using SOM Neural Network. www.researchgate.net.

Nobel F (2015). Data Mining Rule Based Classification Available at: www.tutorialspoint.com/data_mining/dm_rbc.htm.

Ogwueleka FN (2011). Data Mining Application in Credit Card Fraud Detection System. School of Engineering. Taylor's University. J. Eng. Sci. Technol. 6(3):311-322.

Salem SM (2012). An Overview of Research on Auditor's Responsibility to Detect Fraud on Financial Statements. J. Glob. Bus. Manag. 8(2):218-229.

Salganicoff M (1993). Density adaptive learning and forgetting. In Proceeding of the Tenth International Conference on Machine Learning. 276-283 Amherst, MA. Morgan Kaufmann

Sander J, Ester M, Kriegel HP, Xu X (1998). Density-based Clustering in Spatial Databases: The Algorithm DBSCAN and its Applications. Data Min. Knowl. Discov. 2(2):169-194.

Saravanan SK, Babu Suresh GNK (2013). An Analysis of Fraud Detection from Credit Card using Web Data Mining Techniques. Intl. J. Advanc. Res. Data Mining Cloud Comput. 1(1).

Seeja KR, Masoumeh Z (2014). Fraud Miner. A Novel Credit Card Fraud Detection Model Based on Frequent Item set Mining. The Sci. World J. Article ID 252797, 10p.

Sevda S, Mohammad AB (2015). The Study of Fraud Detection in Financial and Credit Institutions with Real Data. J. Comput. Sci. Eng. 5(2):30-36

Srivastava A, Kundu A, Sural S, Majumdar AK (2008). Credit card fraud detection using hidden Markov model. IEEE Trans. Depend. Secure Comp. 5(1):37-48.

Stolfo SJ, Fan DW, Lee W, Prodromidis A, Chan PK, (1997). Credit card fraud detection using meta-learning: Issues and initial results. Proceedings of AAAI-97 Workshop on AI Approaches to Fraud Detection and Risk Management, AAAI Press, Menlo Park, California pp. 83-90.

Wikipedia (2015). Support Vector Machines. https://en.wikipedia.org/wiki/Support_vector_machine.

Witten IH, Frank E, Hall MA (2011). Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann, San Francisco, Calif, USA, 3rd edition.