SUST Journal of Engineering and Computer Science (JECS), Vol. 20, No. 1, 2019

A Fraud-Detection Fuzzy Logic Based System for the Sudanese Financial Sector

Saeed Khalil Saeed [1] and Hani Hagras [2]
Sudan University of Science and Technology, College of Postgraduate Studies, Department Computer Science [1]
University of Essex, U. K. The Computational Intelligence Centre, School of Computer Science and Electronic Engineering [2]

saeed_kl@hotmail.com

Received: 9/05/2019
Accepted: 10/06/2019

ABSTRACT–Financial fraud considered as a global issue that faces the financial sector and economy; as a result, many financial institutions loose hundreds of millions of dollars annually due to fraud. In Sudan, there are difficulties of getting real data from banks and the unavailability of systems which explain the reasons of suspicious transaction. Hence, there is a need for transparent techniques which can automatically detect fraud with high accuracy and identify its causes and common patterns. Some of the Artificial Intelligence (AI) techniques provide good predictive models, nevertheless they are considered as black-box models which are not easy to understand and analyze. In this paper, we developed a novel intelligent type-2 Fuzzy Logic Systems (FLSs) which can detect fraud in debit cards using real world dataset extracted from financial institutions in Sudan. FLSs provide white-box transparent models which employ linguistic labels and IF-Then rules which could be easily analyzed, interpreted and augmented by the fraud experts. The proposed type-2 FLS system learnt its fuzzy sets parameters from data using Fuzzy C-means (FCM) clustering as well as learning the FLS rules from data. The proposed system has the potential to result in highly accurate automatic fraud-detection for the Sudanese financial institutions and banking sectors.

Keywords: *Type-2 fuzzy logic system, Fuzzy C-means, fraud detection, online payments, debit cards.*

المستخلص-يعتبر الاحتيال المالي قضية عالمية تواجة القطاع المالي والاقتصادي, نتيجة لذلك العديد من المؤسسات المالية تفقد مئات الملايين من الدولارات سنويا , وفي السودان هناك صعوبات في الحصول على بيانات حقيقية من البنوك وعدم توفر أنظمة تشرح أسباب المعاملات المشبوهة. وبالتالي ، هناك حاجة إلى تقنيات شفافة يمكنها اكتشاف الاحتيال تلقائيًا بدقة عالية وتحديد أسبابه وأنماطه الشائعة. توفر بعض تقنيات الذكاء الاصطناعي نماذج تنبؤية جيدة ، ومع ذلك ، فهي تعتبر نماذج الصندوق الأسود التي ليست من السهل فهمها وتحليلها. في هذا البحث ، قمنا بتطوير أنظمة ذكية مبتكرة من النوع الثاني **Fuzzy Logic Systems (FLSs)** يمكنها اكتشاف الاحتيال في بطاقات السحب باستخدام مجموعة بيانات حقيقية ومستخرجة من المؤسسات المالية في السودان. توفر **FLSs** نموذج الصندوق الابيض الشفاف و تستخدم علامات لغوية وقواعد **IF–Then** التي يمكن تحليلها بسهولة وتفسيرها وتوسيعها من قبل الخبراء في مجال الاحتيال. تعلم نظام **FLS** من النوع الثاني المقترح مجموعة المعاملات الضبابية من البيانات باستخدام مجموعات **Fuzzy C–mean (FCM)** وكذلك تعلمت قواعد **FLS** من البيانات. يتمتع النظام المقترح بإمكانية الكشف التلقائي عن الاحتيال بشكل دقيق للغاية وذلك للمؤسسات المالية والقطاعات المصرفية السودانية.

INTRODUCTION

Nowadays, mobile payments, online shopping, ATMs and e-commerce have become essential components of our daily lives. However, the financial institutions suffer every day from new fraud patterns which use many techniques, modes and types. The concept of fraud in financial systems includes many types of illegal activities such as falsification of documents, fraudulent loans, fraudulent accounts, online banking fraud, phishing, scamming, credit card fraud and etc.

The fraud crimes cost financial institutions millions of dollars annually which affects the institution financial situation and the customers' confidence. Globally, the estimation of losses made by fraud indicates that fraud costs considerable amounts that are increasing significantly each year. In UK financial fraud losses across payment cards, online transactions and cheques totaled £618 million in 2016 [1] while the online banking fraud losses increased by

9% in 2016 compared to 2015 [1]. In USA the total fraud amount costed about $15 billion [2]. Figure (1) shows some statistics of financial fraud in global market and financial institutions in 2014-2016 this statistic reflects the huge losses and the percentage increases in many countries [3][4]

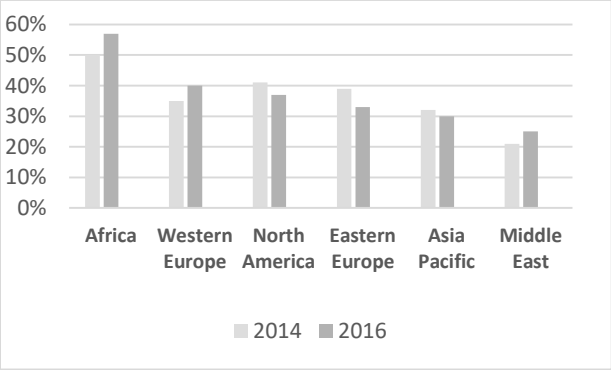

Figure 1 .Financial Fraud in Global Market and Financial Institutions In 2014-2016 [3][4] [5].

In Sudan there are many online banking services such as electricity purchase, custom payments, bill payments and E15 payments. Most of these services are available in ATMs, Point of Sale (PoS) and mobile banking. The Central Bank of Sudan (CBoS) has started E-government project to increase the use of non-cash payments.

After the USA lifted economic sanctions against Sudan on telecommunications and other technology sectors. This allowed banks to make global transactions and motivate the e-commerce besides the use of VISA and MASTER cards. Accordingly, Sudanese banks might face different kinds of fraud cases and should be ready for this global openness. In particular, debit card fraud.

Debit card fraud is to withdraw money from ATMs, PoS or make online payment without owner permission. This includes illegal use of card, card information, Personal Identification Number (PIN) or Internet Personal Identification Number (iPIN), without the owner approval, which is forbidden by law.

Debit/credit card fraud is very complicated process since the legitimate and fraudulent transactions are similar and it is difficult to differentiate between them as the fraud style is not always same and it is an everlasting challenge. Hence, there is a need for novel techniques which can automatically detect fraud and most importantly help to understand fraud common patterns.

Some of the Artificial Intelligence (AI) techniques like support vector machines and neural networks provide good predictive models, nevertheless they are considered as black-box models which are not easy to understand, analyze and augment with human experience and most importantly help to understand fraud common patterns. In this paper, we developed a novel intelligent type-2 Fuzzy Logic Systems (FLSs) which can detect fraud in debit cards using real world dataset extracted from financial institutions in Sudan. The proposed system has the potential to result in highly accurate automatic fraud-detection for the Sudanese financial institutions and banking sectors. Most importantly the proposed system resulted in identifying the common patterns for fraud (which is not possible via black-box model) which can help to design counter measures to stop these fraud patterns from source.

The aim of this paper is to develop novel white box AI technique for financial fraud detection in Sudanese banks starting by focusing on debit card fraud which can be later generalized to other kind of fraud.

The paper is organized as follows, the following section presents a brief overview on fraud detection systems. This will be followed by brief overview on type-1 and type-2 FLSs, and followed by the proposed fraud-detection type-2 FLS for the Sudanese financial sector. This will then be followed by experiments and results. then conclude by presenting the paper conclusions and future work.

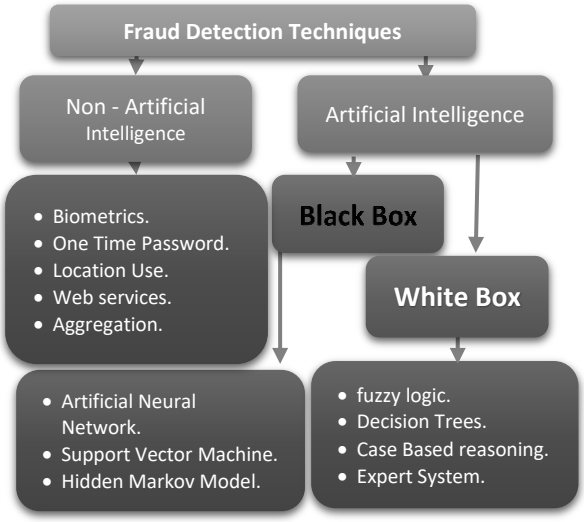

Figure 2. The categorization of fraud detection techniques.

A Brief Overview on Fraud Detection Systems

Fraud detection in financial systems is a very hot research topic which has been studied by many researchers from both academic and industrial fields. Many fraud detection techniques have been successfully applied which could be broadly categorized into Non – AI and AI based as shown in Figure (2).

- **Non - AI Techniques in Debit Card Fraud Detection:**

Several non - AI techniques have been introduced, and most of these techniques are not complicated and easy to understand, but in contrast they depend on other equipment which means these techniques involve hardware or require more resources, this section discusses some of these techniques.

In 2001, Takhar [6] invented credit card fraud elimination system by using the verification of the ID of a credit card's user using fingerprint to prevent the unauthorized use. In [7], they used One Time Password (OTP) to prevent phishing attack that could compromise credit card holders, this technique proposed the user receiving a new password in each transaction by SMS or via alternate email address. This new password is valid for one transaction then the application will receive encrypted token from the web server, as a result the transaction can be authenticated successfully.

Location based credit card fraud prevention [8] was invented in 2009, this technique uses the location of a PoS or ATM and the location of customer's mobile which must be same. The location can be selected by using Global Positioning system (GPS), Observed time difference (OTD), Time of arrival (TOA), Time difference of arrival (TDOA), Received signal strength (RSS) etc.

However, the above mentioned techniques are impractical solutions in online transactions, particularly when the customer goes abroad.

A web services-based collaborative scheme applies shared web services where banks share their information about fraud patterns[9]. Transaction Aggregation Strategy [10] aggregates historical transactions to capture spending pattern for each transaction then identify the fraudulent transaction by using Average to recognize fraudulent transactions and some authors used AI techniques like random forests [11] as classification method. The aggregation strategy depends on customer behavior and thus has low accuracy.

- **AI Techniques in Credit Card Fraud Detection (Black Box and White Box Algorithms):**

The AI algorithms can be categorized as Black Box algorithms (BB) or White Box algorithms (WB) which are considered more transparent[12], consequently, it is easy to understand and analyze. Figure (3) shows that the white box algorithm can be more clear to the user.

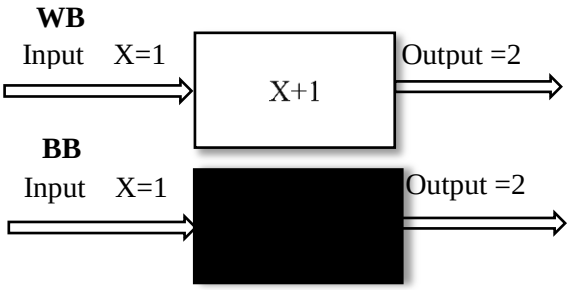

Figure 3. White Box & Black Box Algorithms.

A Brief Overview on Black Box Models for Fraud Detection:

Artificial Neural Network (ANN) used in credit card fraud detection [13].

There are two types of ANN learning method [14], [15], [16]:

Supervised (using Labeled data in training such as fraudulent or legitimate transactions).

Unsupervised (using behavior method such as normal or fraudsters behavior and no need for historical data).

Supervised and Unsupervised.

In [17] , they proposed CARDWATCH (supervised method) based on a NN with three layers , The idea of this technique is to train the neural network with the historical data (spending patterns, transaction time ...) of a specific customer and let the NN detect anomalies using pattern recognition as database mining tool.

Parallel Granular Neural Networks (GNN) [18] aim to speed up knowledge discovery and data mining, GNN it is a kind of Fuzzy Neural Network based on Knowledge Discovery (FNNKD) which

uses parallel processing to train parallel fuzzy neural network to produce fuzzy rules which can be used in prediction and then in fraud detection [18].

ANN can be trained by different attributes such as [19]:

- Geolocation: this by using computer IP address to recognize the location.
- Email address, Shipping address or phone number.
- Regular products and services, neural network also trained by regular customer's purchases type and services type for the recent years.

A simulated annealing algorithm is used to learn the neural network by initializing random weights then evaluate the results if the result is not appropriate it repeats with new random solutions until reaching good result to train NN [20], [21].

ANN is able to learn from the historical experiments. However, NNs require long training and testing time and it is very difficult to understand the cause of capturing the fraudulent transaction which can result in high false positive rate.

Support Vector Machines (SVMs) shown in Figure (4) were introduced by Cortes and Vapnik in 1995 [22]. In [23] they used SVM in fraud detection. SVM has a good performance but it is a complex classification algorithm and thus lacks transparency.

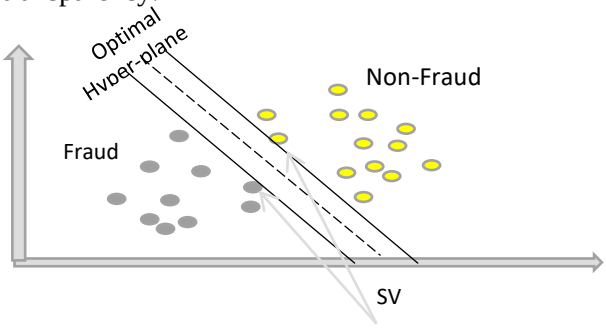

Figure 4. Support Vector Machine.

Hidden Markov Model is a statistical tool containing a finite set of states controlled by a set of transition probabilities [24][25]. At first HMM is trained by the normal behavior of a normal customer such as a spending pattern. When the trained HMM receives unaccepted customer transaction or receives any transaction which has anomaly with high probability, it considered it as fraudulent transaction [24][26]. In [27], they used HMM with three spending profiles of the card holder (Low (0, $100), Medium ($100, $500) , High ($500,and more)) then they examine incoming transaction against spending profiles of customers if it is rejected then it would not be genuine transaction. HMM is fast in detection process but produces high false alarm and cannot detect new kinds of fraud. So if the thief used a high-classified spending card he might not be captured.

A Brief Overview on White Box Models for Fraud Detection:

Case-Based Reasoning (CBR) is a model for solving problems that generates solutions from previous similar cases and reuse them in new problem cases. This approach is continued learning, since a new knowledge is taken each time a problem has been solved and making it directly available for upcoming problems [28]. CBR algorithm contain several neighborhood-based and probabilistic algorithms such as Case retrieval (Nearest neighbor matching)[29].

K-NN algorithm is a clustering supervised learning algorithm, it classifies any received transaction by calculating of nearest point to new received transaction, if the nearest neighbor is fraudulent, then the transaction marked as a fraudulent and vice versa[30]. Accordingly KNN is expanding the number of neighbors but fixed number in nearest neighbor [29] and it is fast in detection process but sometimes offers suboptimal results.

Expert Systems (ES) obtain knowledge from a human expert and store it in a rule-based system such as IF-THEN rules. In [31], they presented an expert rule based model to detect the fraudulent usage of card before the fraud transaction has been reported by the cardholder, exactly within the authorization process, the goal of this approach is just to receive genuine transactions.

In [32] , they proposed FUZZGY using fuzzy expert system, using Fuzzy rules rather than crisp rules and calculate anomaly degree for each customer. Finally, FUZZGY compute the suspicious degree of new transactions compared by customer behavior. It is Easy in ES to modify the Knowledge base and add new rules, but it is poor in handling unexpected data or data lost.

Decision tree (DT) is a data mining technique which is used to solve complicated problems [33][34]. DT Pruning in C4.5[35], remove some branches to simplify and understand

the decision tree by converting the big tree to small tress. This method targeted to improve the accuracy and speed of classification by pruning sub trees from the decision tree, there are many DT pruning techniques that use statistical measures to eliminate the least dependable branches, and some of these techniques are:

- Reduced Error Pruning.
- Cost-Sensitive Decision Tree Pruning.
- Pessimistic Error Pruning.
- Optimal Pruning.

Decision trees is easy to understand and easy to implement (explainable) and capable of dealing with noisy data plus has a good flexibility and powerful in classification however require maintenance regularly to check the new leafs.

Fuzzy logic is used in credit card fraud detection. In [36] , they proposed Fuzzy Evolutionary Detection technique, This technique describes the use of genetic programming (GP) & fuzzy expert system to develop fuzzy logic rules capable of categorizing credit card transactions into two groups "suspicious" and "non-suspicious". Fuzzy association rules [37] extracts a best set of fuzzy rules from a data set containing genuine and fraudulent transactions and uses these results with incoming transaction .

Fuzzy logic is explainable and has a good knowledge representation plus is maintainable due to the transparency. However, type-1 fuzzy logic cannot deal with uncertainty.

Brief Overview on Type-1 and Type-2 Fuzzy Logic Systems

This section discusses an introduction to Type-1 and Type-2 Fuzzy Logic Systems (FLSs).

Fuzzy Logic introduced by Lotfi A. Zadeh in the 1965, FLS tries to mimic the way of human thinking, which is approximate and imprecise way such as linguistic human concepts (Cold, Hot, Tall and Short) they are not precise [38][39] . The traditional logical systems use Boolean logic or crisp sets and they have sharp boundaries between custom set as shown in Figure 5. Shows crisp sets and illustrates sharp boundaries. Figure 6. Shows the young fuzzy sets and the smooth transition between the sets.

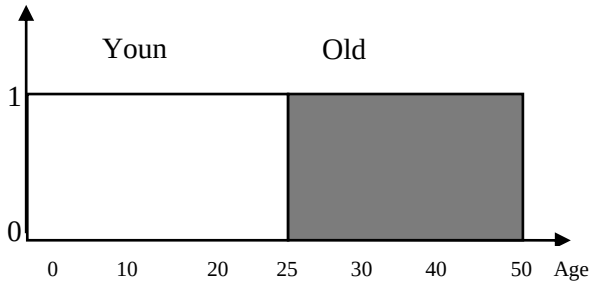

Figure 5. Crisp sets.

Form Figure 6. apparently can realize the similarity between the human thinking and the mathematical expression, therefore the human always tries to describe the young person from the first sight by saying "he/she is a young man" we are not saying "he/she is 22 years old". Consequently, we are sometimes not precise, in fuzzy set it is possible to calculate the membership or the degree to which an item is a member.

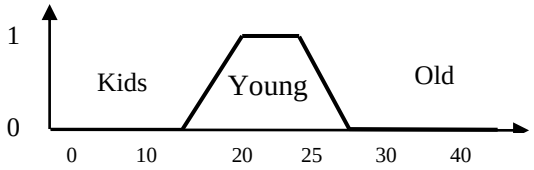

Figure 6. Fuzzy Set.

Figure 7. Shows the Structure of type-1 fuzzy logic controller which consist from four main parts.

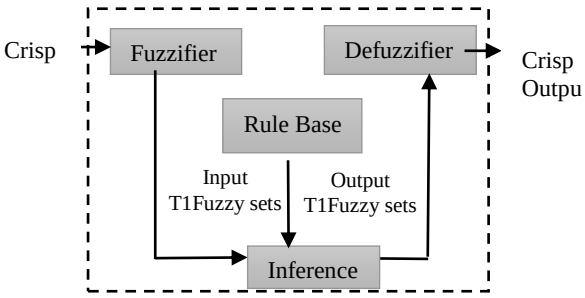

Figure 7. The Structure of T1FLS.

The fuzzification role is to convert each crisp input or measurements to fuzzy values. Rule Base or set of IF-Then rules are the core of a FLS, rules can be extracted from numerical data or can be designed by experts, these rules are fired by using inference mechanisms which play very essential role by receiving a fuzzy input sets from the fuzzifier and produces a fuzzy output sets to the defuzzifier in addition it selects the corresponding rules from the rule-based to be triggered to produce fuzzy output sets. Defuzzification

produces crisp outputs from the fuzzy sets that appear at the output of a fuzzy inference machine.

Type-2 Fuzzy Logic Systems:

Type-2 fuzzy sets are useful in circumstances where it is difficult to determine the precise membership function for a fuzzy set [40][41], [42], [43].

Type 1 fuzzy sets cannot handle the high level of linguistic and numerical uncertainties. Type-2 fuzzy sets (shown in Figure 8) has a three dimensional membership function and a Footprint of Uncertainty (FOU) located between the lower membership and the upper membership functions which provide extra degrees of freedom to better handle and model higher degrees of uncertainties.

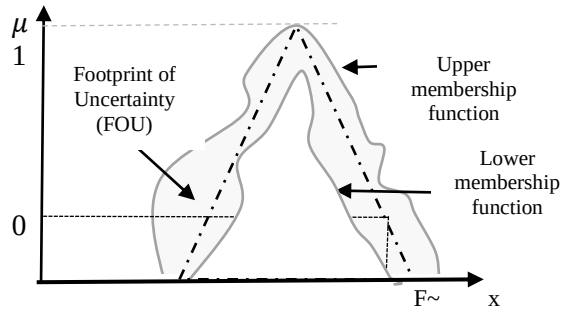

Figure 8. Membership of a type-2 fuzzy set [43]

Many researches proved that using interval type-2 fuzzy sets to characterize the inputs or/and outputs of FLS has many features and advantages when compared to the type-1 fuzzy sets. Since the additional degrees of freedom provided by the FOU allows a type-2 FLS to produce outputs that cannot be achieved by type-1 FLSs with the identical number of membership functions[42].

Figure 9. Shows the structure of type-2 FLS, the main difference is a type reducer; the type reducer and defuzzifier will perform the type-reduction and defuzzification to get an output crisp value from the output type-2 fuzzy set.

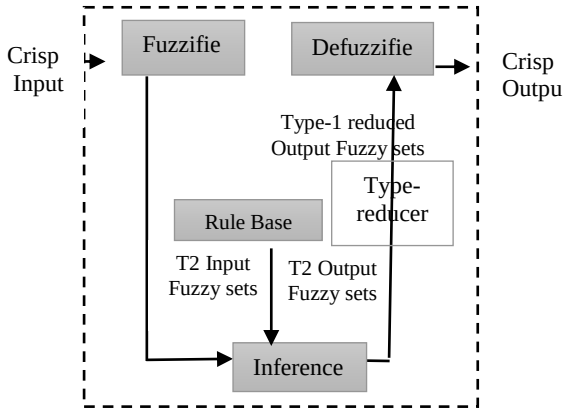

Figure 9. The Structure of T2FLS [40] [42][43]

The proposed Fraud Detection Type-2 Fuzzy Logic Based System for the Sudanese Financial Sector:

This section explains in details the main components of the proposed technique, and clarifies how the modeling phase and prediction phase work. Figure 10. shows how the proposed Type-2 Fuzzy Logic System (T2FLS) for fraud detection works where it starts with the training phase of extracting the rules from the dataset then handling these rules by calculating the weighted scaled dominance, which is a new approach used to resolve the conflicting rules when the data is highly imbalanced because the majority class of non-fraud transaction is much greater than fraudulent transactions, this is called training phase.

Figure 10. shows, the proposed FLS has two main components the first is the training phase and the second is the prediction phase, the training phase works according to the following steps detailed in the following subsections.

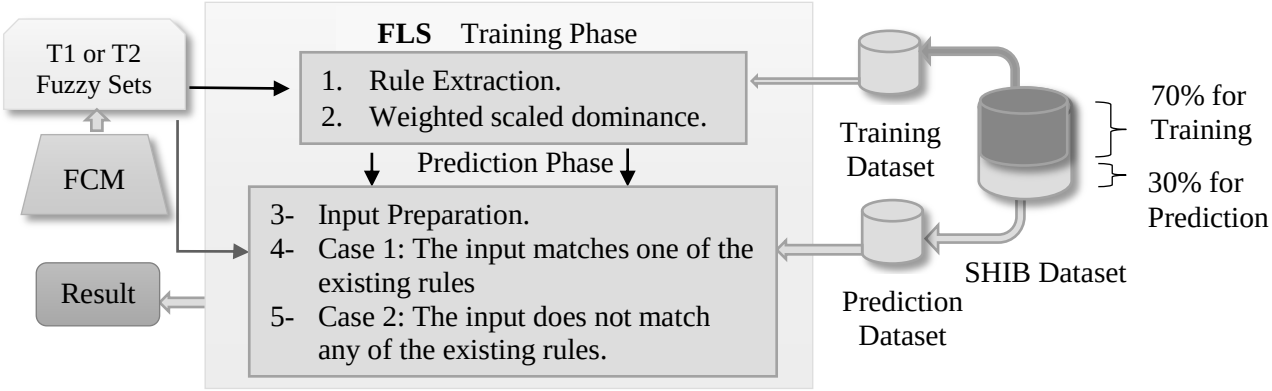

Figure 10. Type-2 Fuzzy Logic Based System for Fraud Detection for the Sudanese Financial Sector

Generation of T1 & T2 Fuzzy Set From Data:
To generate the fuzzy set, Fuzzy c-means (FCM) clustering algorithm was used which allows one piece of numerical data to belong to many clusters with different membership values. This algorithm developed by Dunn in 1973 and improved by Jim Bezdek in 1981 [44] is widely used in pattern recognition. It is based on minimization of the following objective function [44]:

$$J_m = \sum_{i=1}^{n}\sum_{j=1}^{c} \mu_{ij}^{m}\|x_i - v_i\|^2, 1 \le m < \infty \quad (1)$$

Where *m* is the weighting exponent and usually set to 2, μ_{ij}^{m} membership values ranging from [0,1], x_i is the *i*th of d-dimensional measured data and v_j their *d-dimension* of the cluster centers, v_j can be updated and summarized by the following equations:

$$v_i = \frac{\sum_{i=1}^{n}\mu_{ij}^{m}x_i}{\sum_{i=1}^{n}\mu_{ij}^{m}}, j = (1,2,\dots C) \quad (2)$$

Then update μ_{ij}^{m} Membership with v_i by using:

$$\mu_{ij} = \left(\sum_{k=1}^{c}\left(\frac{\|x_i - v_i\|}{\|x_i - v_k\|}\right)^{\frac{2}{m-1}}\right)^{-1}, j = (1,2,\dots C), i = (1,2,\dots N) \quad (3)$$

This iteration will stop when $\| \mu^{(k)} - \mu^{(k-1)} \| < \varepsilon$; otherwise return to equation (2), where ε is a termination criterion between 0 and 1, whereas k are the iteration steps.

Step A: Raw Rule Extraction: The rule extraction approach used by type-2 is based on [45][50],from dataset each input–output pair ($x^{(t)}$, $C^{(t)}$), *t = 1, . . . T* (where T is the total number of training dataset records available for the training phase) for each antecedent, calculate the upper and lower membership values ($\bar{\mu}_{A_s^q}$, $\underline{\mu}_{A_s^q}$) .

Each input fuzzy set *q = 1, . . . K* (where *K* is the total number of fuzzy sets representing the input pattern s, where *s = 1 . . . n*). Extract all rules combining the matched fuzzy sets A_s^q (i.e. either $\bar{\mu}_{A_s^q} > 0$ or $\underline{\mu}_{A_s^q} > 0$) for all *s = 1 . . . n.*

Therefore, the rules represented by ($x^{(t)}$, $C^{(t)}$) will have different antecedents and the equivalent consequent class *C(t)*. once the instance crosses many sets then one input can generate more than one rule, consequently each of the generated rules by ($x^{(t)}$, $C^{(t)}$) can be written as follows:

$$R^j: If\ x_1 is\ \tilde{A}_1^{qjt}\ and \dots and\ x_n\ is\ \tilde{A}_n^{qjt}\ then\ class\ C_t, t = 1,2 \dots T \quad (4)$$

then calculate the firing strength F^t for each extracted rule, the benefit of the firing strength is to determine and measure the strength of the points $x^{(t)}$. F^t is defined in terms of the lower and upper bounds of the firing strength ($\overline{f^{(t)}}$, $\underline{f^{(t)}}$) which can be expressed as follows:

$$\underline{f^{jt}}(x^{(t)}) = \underline{\mu_{A_1^{qjt}}}(x_1) * \dots * \underline{\mu_{A_n^{qjt}}}(x_n) \quad (5)$$

$$\overline{f^{jt}}(x^{(t)}) = \overline{\mu_{A_1^{qjt}}}(x_1) * \dots * \overline{\mu_{A_n^{qjt}}}(x_n) \quad (6)$$

Step (A) is repeated for all the *t* input data points from *1* to *T* to obtain extracted rules in the form of Equation (4).

In general, the financial data is usually highly imbalanced especially in fraud applications where fraudulent transactions represent the minority class. Hence, an approach were used to handle imbalanced data by trying to give minority classes good chance when competing with the majority class. This approach called "weighted scaled dominance" [45] which is based on the weighted confidence measure introduced by [47]. To calculate the scaled dominance for each given rule having a consequent Class C_j, the firing strength of this rule was divided by the summation of the firing strengths of all the rules which had C_j as the consequent class. the firing strength was scaled by scaling the upper and lower bounds of the computed firing strengths, which can be calculated as follows:

$$\overline{fs^{jt}} = \frac{\overline{f^{jt}}}{\Sigma_{J\epsilon\ class}\overline{f^{jt}}} \quad (7)$$

$$\underline{fs^{jt}} = \frac{\underline{f^{jt}}}{\Sigma_{J\epsilon\ class}\underline{f^{jt}}} \quad (8)$$

This facilities handling the imbalance of data towards a given class.

Step B: Weighted Scaled Dominance:
To calculate the weighted scaled dominance [45], the scaled confidence and scaled support must be computed by grouping the rules that have the same antecedents and conflicting classes, this is very significant to resolve this conflict.

For given *m* conflicting rules with the same

antecedents and conflicting classes, the definition of the scaled confidence ($\tilde{A}_q \rightarrow C_q$) which has upper bound $\bar{c}$ and lower bound $\underline{c}$,that class C_q is the consequent class for the antecedents $\tilde{A}_q$ can be written as follows:

$$\bar{c}(\tilde{A}_q \rightarrow C_q) = \frac{\Sigma_{xs \in class C_q} \overline{f^{jt} x(s)}}{\sum_{j=1}^{m} \overline{f^{jt} x(s)}} \quad (9)$$

$$\underline{c}(\tilde{A}_q \rightarrow C_q) = \frac{\Sigma_{xs \in class C_q} \underline{f^{jt} x(s)}}{\sum_{j=1}^{m} \underline{f^{jt} x(s)}} \quad (10)$$

The confidence can be viewed as a numerical approximation of the conditional probability [49], the scaled confidence can be viewed as measuring the validity of $\tilde{A}_q \rightarrow C_q$, whereas the support can be viewed as measuring the coverage of training patterns by $\tilde{A}_q \rightarrow C_q$. The scaled support (defined by its upper bound $\bar{s}$ and lower bound $\underline{s}$, it is scaled as it includes the scaled firing strengths mentioned in the previous step) is written as follows:

$$\bar{s}(\tilde{A}_q \rightarrow C_q) = \frac{\Sigma_{xs \in class C_q} \overline{f^{jt} x(s)}}{m} \quad (11)$$

$$\underline{s}(\tilde{A}_q \rightarrow C_q) = \frac{\Sigma_{xs \in class C_q} \underline{f^{jt} x(s)}}{m} \quad (12)$$

The scaled dominance, (defined by its upper bound $\bar{d}$ and lower bound $\underline{d}$) can now be computed by multiplying the scaled support and scaled confidence of the rule and can be written as follows:

$$\bar{d}(\tilde{A}_q \rightarrow C_q) = \bar{s}(\tilde{A}_q \rightarrow C_q) * \bar{c}(\tilde{A}_q \rightarrow C_q) \quad (13)$$

$$\underline{d}(\tilde{A}_q \rightarrow C_q) = \underline{s}(\tilde{A}_q \rightarrow C_q) * \underline{c}(\tilde{A}_q \rightarrow C_q) \quad (14)$$

To calculate the (weighted scaled dominance) (which is defined by its upper bound $\overline{wd}$ and lower bound $\underline{wd}$) multiply the scaled dominance (mentioned in the previous step) by the average dominance d_{ave} (defined in terms of $\overline{d_{ave}}$ and $\underline{d_{ave}}$) over fuzzy rules with the same antecedent $\tilde{A}_q$ but different consequent classes C_q which are calculated as follows:

$$\overline{wd}(\tilde{A}_q \rightarrow C_q) = \bar{d}(\tilde{A}_q \rightarrow C_q) * \overline{d_{ave}} \quad (15)$$

$$\underline{wd}(\tilde{A}_q \rightarrow C_q) = \underline{d}(\tilde{A}_q \rightarrow C_q) * \underline{d_{ave}} \quad (16)$$

Consequently, to resolve the conflict in the rules, replace these rules by one rule having the same antecedents and the consequent class which will be equivalent to the rule that results the highest average value in (weighted scaled dominance) average value = $\frac{\overline{wd} + \underline{wd}}{2}$.

The Prediction Phase:

Once an input pattern is entered from the prediction dataset to the produced model, instantly calculate the upper and lower membership values ($\bar{\mu}_{A_s^q}$, $\underline{\mu}_{A_s^q}$) . Two prospects might take place: the first case is when the input $x^{(p)}$ matches any of the X rules in the produced model, in this prospect follow the process illustrated by case 1 below. If $x^{(p)}$ does not match any of the existing X rules, follow the process illustrated by case 2.

In [45] [48], the produced model generated only the rule with the highest firing strength, however all rules were generated that are produced by the given input patterns, consequently this allows covering a larger area in the decision space.

Case 1 - The Input Matches One of the Existing Rules:

In this situation the incoming input $x^{(p)}$ matches any of the existing X rules, compute the firing strength of the matched rules as calculated before in training phase in Equations (5) and (6), this will result in $\overline{f^j}(x^{(p)})$, $\underline{f^j}(x^{(p)})$. In this case, the predicted class will be identified by calculating a vote for each class which could be as follows:

$$\bar{z}Class_h(x^{(p)}) = \frac{\Sigma_{x \in h} \overline{f^j}(x^{(p)}) * \overline{wd}(\tilde{A}_q \rightarrow C_q)}{max_{j \in h} \overline{f^j}((x^{(p)}) * \overline{wd}(\tilde{A}_q \rightarrow C_q))} \quad (17)$$

$$\underline{z}Class_h(x^{(p)}) = \frac{\Sigma_{x \in h} \underline{f^j}(x^{(p)}) * \underline{wd}(\tilde{A}_q \rightarrow C_q)}{max_{j \in h} \underline{f^j}(x^{(p)}) * \underline{wd}(\tilde{A}_q \rightarrow C_q))} \quad (18)$$

The above equations taking the summation of the product of the upper and lower firing strengths and the weighted scaled dominance (which is calculated previously in training phase) divided by the maximum of the product of the upper and lower firing strengths and the weighted scaled dominance (which is already calculated in training phase) correspondingly among the "K" rules selected for each class. The total vote strength is then computed as follows:

$$zClass_h = \frac{\bar{z}Class_h(x^{(p)}) + \underline{z}Class_h(x^{(p)})}{2} \quad (19)$$

From the incoming input vector $x^{(p)}$ the predicted class will be the class with the highest $zClass_h$.

Case 2 - The Input Does Not Match Any of The Existing Rules:

The output class must be determined for the input, in the case of incoming input vector $x^{(p)}$ does not match any of the existing X rules, once an input pattern is entered from the prediction dataset to the produced model, as mentioned later calculate the upper and lower membership values ($\bar{\mu}_{A_s^q}$, $\underline{\mu}_{A_s^q}$) for each inputs, and once the input matches many sets then one input can generate more than one instance, and each rule will have an associated a firing strength but not an output class. The next step is to find the closest rule in the rule base for each rule in *MR* ($x^{(p)}$), where *MR* ($x^{(p)}$) is the set of rules obtained by combining the matched fuzzy sets. To do this, compute the similarity (or distance) versus each of the fuzzy rules produced by $x^{(p)}$ and each of the *X* rules stored in the rule base in the generated model. When "*k*" is the number of rules created from the input $x^{(p)}$. furthermore the linguistic labels (i.e. *Low, Medium, High*, etc) that fit $x^{(p)}$ be written as $v_{inputr} = (v_{input1r}, v_{input2r}, \ldots, v_{inputnr})$ where r is the index of the *r*-th rule generated from the input as mentioned later. Let the linguistic labels matching to a given rule in the rule base be $v_j = (v_{j1}, v_{j2}, \ldots, v_{jn})$. Each of these linguistic labels could be converted into a number, where $V_1 \ldots V_n$ represents the number of linguistic labels representing each variable. Therefore, the similarity will be calculated by finding the distance between the two vectors as follows:

$$Similarity_{\ input\, r \leftrightarrow j} = \left(\left(1 - \left|\frac{vinput1r - vj1}{v1}\right|\right) * \left(1 - \left|\frac{vinput2r - vj2}{v2}\right|\right) * \ldots * \left(1 - \left|\frac{vinputnr - vjn}{vn}\right|\right)\right) \quad (20)$$

At this stage each rule in the rule base will have a similarity associated with the *r*-th rule generated form the input. For each rule in *MR* ($x^{(p)}$) the most similar rule in the rule base, and by using above equation we can determine the output class. There will be "*k*" rules selected to decide for the $x^{(p)}$ input the output class (where "*k*" is the most similar rules to the *k* rules in *MR* ($x^{(p)}$)). Finally the predicted class will be determined as a vote for each class as same as mentioned in Equations (17) and (18) then the total vote strength can be computed as Equations (19) by taking the highest $zClass_h$.

Experiments:

Our study acquired payment data for one year (2016), from an Alshamal Islamic bank (SHIB) Khartoum – Sudan, which contains multi type of transactions such as: ATMs, POS, mobile Banking and internet Banking etc. The dataset contains 803,386 rows with 107 fraud transaction hence the dataset is highly unbalanced, the positive class (frauds) rate is 0.0133% of all transactions. Unfortunately, due to confidentiality issues, we cannot provide all features and more background information about the data such as Cardholder info: Names, Addresses, Card Number, Account number, Mobile etc. But 17 important features were used for this research:

1. **Transaction Time**: when transaction held (early morning, morning, day, mid of day, night or mid night).
2. **Transaction Amount**: the amount of transaction (very small amount, small amount, mid amount, large amount or very large amount).
3. **Gender**: male or female.
4. **Branch**: branch of the card holder.
5. **Reference**: reference of transaction.
6. **Occupation**: job of the card holder.
7. **Education**: education of the customer (none, basic, high school, B.Sc., PhD…).
8. **Account Type**: saving, current, investment or employee account.
9. **Marital Status**: single, married, divorced or widowed.
10. **Week Day**: beginning, mid or end of Week.
11. **Day of Month**: beginning of month, mid of month or end of month
12. **Age**: young, middle, old and very old.
13. **Transaction Type**: mobile application, internet transaction, ATM/PoS transaction,
14. **Service Type**: cash, bill payment, E15, NEC, or mobile top-up.
15. **Bank Terminal**: where transaction held.
16. **City**: city of terminal where transaction held.
17. **Class**: fraud or non-Fraud.

We have performed this experiment starting by data collection and we designed type-1 fuzzy sets with equally space sets. We then used FCM to extract the type-1 fuzzy sets from data, and to

improve our result we used type-2 fuzzy set while varying the FOU to 10%, 20% and 30%.

We randomly divided the dataset to 70% for training and 30% for prediction stage as shown in figure 10.

Designing Fuzzy Sets Using FCM:

We have used the FCM algorithm to realize the type-1 fuzzy sets where Figure 11. shows an example of the shapes the age fuzzy sets generated by FCM.

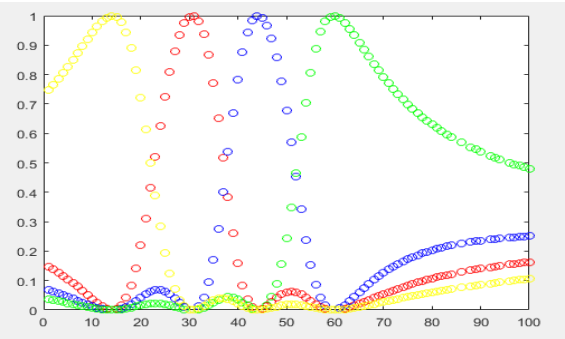

Figure 11. Type-1 Fuzzy Set Generated by FCM for Age.

We approximated the shapes shown in Figure (11) to generate convex normal type-1 fuzzy sets as shown in Figure 12.

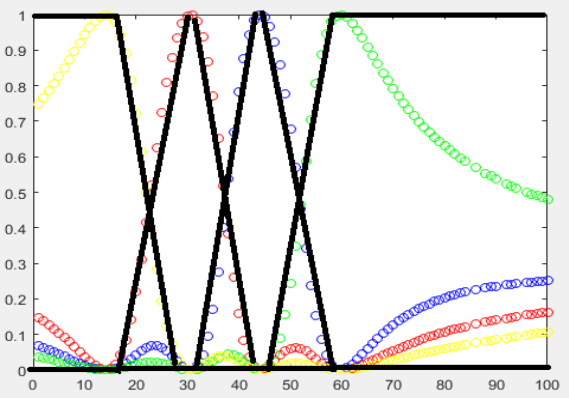

Figure 12. Generated Convex Normal Type-1 Fuzzy Sets from the FCM Results in Figure 11.

Calculate AVG Recall using Confusion Matrix:

In order to evaluate and measure the results of the proposed technique a confusion matrix was used, which is a table that is often used to describe the performance of a classification. It contains information about actual and predicted classifications done by a classification system, in our work confusion matrix was used for a binary classifier (Fraud Transaction or Non-Fraud Transaction). Table 1. shows a Confusion matrix for a binary classifier which is a table with two rows and two columns that reports four possible results [46]:

- **False Positives (FP):** We predicted positive, and they do negatives.
- **True Positives (TP):** These are cases in which we predicted positives (they Non-Fraud), and they do positives.
- **True Negatives (TN):** We predicted Negatives, and they do negatives.
- **False Negatives (FN):** We predicted negative, but they actually positive.

Table 1. Confusion Matrix for a Binary Classifier.

		Actual class	
Predicted class		Actual Positive	Actual Negative
	Positive Prediction	TP	FP
	Negative Prediction	FN	TN

In binary classification a Recall also known as sensitivity or true positive rate which is defined as the fraction of positive cases that were correctly identified by [46], as follows:

$$\text{Recall Positive rate} = \frac{\text{TP}}{\text{TP} + \text{FN}} \qquad (21)$$

Recall is calculated on the positive class and negative class by the formula:

$$\text{Recall Negative rate} = \frac{\text{TN}}{\text{TN} + \text{FP}} \qquad (22)$$

Consequently, the average recall is:

$$\text{AVG } \textit{Recall} \text{ rate} = \frac{\text{Recall positive} + \text{Recall Negative}}{2} \qquad (23)$$

Results:

Our target is not to increase the accuracy in prediction but to explain why the transaction is fraudulent, nevertheless we achieved a good result. We designed a universal Type-1 & Type-2 fuzzy logic system by using JAVA programming language, this application can be configured with any dataset and any number of fuzzy set, the application has the ability to draw the fuzzy set as shown in Figure 13. it illustrates type-2 fuzzy set for Day of Month with three sets (Beginning of month, Mid of month and End of month)

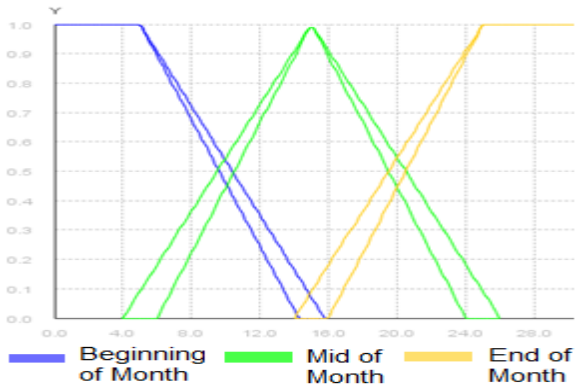

Figure 13. Type-2 Fuzzy Set for Day of Month Generated from Fuzzy Logic System.

Firstly, we computed the results for Type-1 fuzzy logic with equally space sets for prediction data and training data, then we used type-1 fuzzy sets that generated by FCM and we calculated the results for prediction data and training data again, the results for both was similar which is 84% for prediction data, and 99% for training data which is good results (as shown in table 2 & 3).

Then we used type-2 fuzzy set with equal incremental in FOU for each fuzzy sets as 10%, 20% and 30% for prediction data.

Table 2. AVG Recall Rates for Training Data in Type-1 & Type-2 FLC using FCM.

Type	Recall Positive	Recall Negative	AVG Recall Rate
T1 (Equally Space)	99%	98%	98.5%
T1 (FCM)	99%	98%	98.5%
Type-2 FLC using FCM			
10%	99%	97%	98%
20%	99%	98%	98.5%
30%	99%	98%	98.5%

Table 3. AVG Recall Rates for Predicting Data in Type-1 & Type-2 FLC using FCM.

Type	Recall Positive	Recall Negative	AVG Recall Rate
T1 (Equally Space)	99%	69%	84%
T1 (FCM)	99%	70%	84.5%
Type-2 FLC using FCM			
10%	99%	66%	82.5%
20%	99%	72%	85.5%
30%	99%	72%	85.5%

Finally, we have compared all the results and we selected the best AVG recall, hence 30% of incremental in FOU can result 85.5% in prediction data besides 99% in learning data.

Discussions

The most significant matter in our model each result can be read with clear justification for example below is one rule taken from the rule base consequently the experts or the employees can easily read it:

" IF BRANCH is BurjBranch and AMOUNT is Larg and GENDER is Female and OCCUPATION is Student and EDUCATION is High School and MARITAL_STATUS is Single and ACC_TYPE is isSavingAcc and TIME_TR is mid night and DAY_OF_MONTH is Mid and WEEKDAY is MidWeek and AGE is young and CITY is Khartoum and BANK is SSOD and TR_TYPE is ATM-PoS and SERVICE is Cash-PoS Then Fraud".

From the rule above we can infer there is someone used a female student card's and took large amount at mid of month and the time was night, and generally the students use their cards with small amount at morning or day time.

In this work, we have executed several experiments to improve the results, we started out into type-1 FLS with equal space fuzzy sets were results slightly improved with type-1 FCM fuzzy sets and then type-2 was much better and that we tried to adjust the foot print of uncertainty (FOU) in interval type-2 to find the best percentage of FOU, we conducted from varying FOU (10%, 20% and 30%) the best results, which are in 20% and 30% of FOU, this illustrate the amount of uncertainty was increased to cover all possibilities when we increased the FOU.

The developed system generated new rules were not generated before in type-2 with 10% incremental in FCM, this rules increased the AVG recall rate, for example:

"IF BRANCH is MadaniBranch and AMOUNT is VerySmall and GENDER is Male and OCCUPATION is Worker and EDUCATION is Basic and MARITAL_STATUS is Married and ACC_TYPE is isSavingAcc and TIME_TR is Day and DAY_OF_MONTH is Mid and WEEKDAY is EndWeek

and AGE is Old and CITY is Khartoum and BANK is SHIB and TR_TYPE is Sudani-SMS and SERVICE is SUDANI TopUp Then Fraud"

From this rule someone tries to take a very small amount from the old worker man, the fraudulent person repeated this transaction three/four times per day by using worker man mobile's, it does not make sense to make Mobile Balance Transfer many times per day, the developed system predicted this transaction as fraudulent transaction, such as this rules were generated from the developed system which are very significant and allowing the financial sector in Sudan to track fraud patterns as well as these rules are very simple and explainable which can be easily read, as we realized from the above examples and from the learning phase the developed system can generate rules that can deal with Sudanese society and the stop any unacceptable behaviors, and any financial institution simply explain how the fraud can take a place by using these transparent rules.

Conclusions and Future Work:

In this paper, we developed an intelligent type-2 fuzzy logic systems which can detect fraud in Sudanese bank starting by debit cards and using real world dataset.

The electronic-payment (e-payment) environment in Sudan is different from the others, because there are amount ceilings for some type of transactions and we use a debit cards not credit cards which is mean we have different environment so the proposed solution was upgraded to universal solution hence it can work in any environment but still we need to assessment the results by using different datasets.

We have shown how the proposed system can learn from Alshamal Islamic bank data sets of the input type-1 with equal space fuzzy sets and type-2 fuzzy sets using FCM and the effect of increasing the FOU, and we used data mining measure called weighted scaled dominance to handle imbalanced data, and We have presented how the prediction phase works in case of the input matches one of the existing rules and in case of the input does not match any of the existing rules which was called similarity measure.

We used the confusion matrix to calculate the results by taking the highest AVG recall which was type-2 with FCM with 20% and 30% increment in FOU, which was better than using type-1 FLS, hence 30% of incremental in FOU resulted 85.5% in prediction data besides 99% in learning data, so this proved the proposed whit box type-2 system better than type-1 FLS.

we have presented clear, transparent and simple models and some examples of rules were generated by the proposed system and explained the simplicity of the rules and these rules can help to identify fraud patterns and helping to stop fraud from happening in the financial sector in Sudan.

Hence, for our future work, we will aim to tune the type-2 fuzzy set and optimize also the length of the rules by using one of the Evolutionary-Type-2 FLC such as Genetic Algorithm GA and big bang big Crunch BB-BC.

We will aim also to develop self learning fraud detection systems helping to track any fraud patterns and update the models automatically.

***REFERENCES*:**

[1] Financial Fraud Action UK, (2017) "Fraud the facts 2017", https://www.financialfraudaction.org.uk/fraudfacts17/ (retrieved 23 March 2019).

[2] Al Pascual, K. Marchini and S. Miller, (2016) "Identity Fraud: Fraud Hits an Inflection Point", https://www.javelinstrategy.com/coverage-area/2016-identity-fraud-fraud-hits-inflection-point (retrieved 23 March 2019).

[3] Global Economic Crime Survey, (2016), www.pwc.com/crimesurvey (retrieved 15 May 2018).

[4] Report to the nation on occupational fraud and abuse, https://www.acfe.com/rttn2016/about/executive-summary.aspx (retrieved 21 May 2018).

[5] Kroll, "GLOBAL FRAUD Vulnerabilities on the Rise," http://www.kroll.com. (retrieved 2 May 2017).

[6] Takhar H., (2001) "Credit card fraud elimination system" U.S. Patent 20010053239 A1.

[7] Khan A. A., (2013). "Preventing phishing attacks using one time password and user machine identification" International Journal of Computer Applications 68: 7-11.

[8] Sharma R. and Wang S., (2007) "Location based credit card fraud prevention" US Patent 2009/0012898 A1.

[9] Chiu C.-C. and Tsai C.-Y, (2004) "A web services-based collaborative scheme for credit card fraud detection" in: Proceedings of the IEEE International Conference on e-Technology, e-Commerce and e-Service, pp. 177–181, Taiwan.

[10] Jha S., Guillen M. and Westland J. C., (2012), "Employing transaction aggregation strategy to detect credit card fraud" Expert systems with applications. 39: 12650–12657.

[11] Whitrow C., Hand D. J., Juszczak P., Weston D. and Adams N. M., (2009) "Transaction aggregation as a strategy for credit card fraud detection" Data Mining and Knowledge Discovery 18: 30–55.

[12] Delibašić B., Vukićević M., Jovanović M. and Suknović M. (2013) "White-Box or Black-Box Decision Tree Algorithms: Which to Use in Education?" IEEE Transactions on Education 56: 287–291.

[13] Wang S.-C., (2003) "Artificial neural network" Interdisciplinary computing in java programming: 81–100.

[14] Zareapoor M., Seeja K. R. and Alam M. A., (2012) "Analysis on Credit Card Fraud Detection Techniques: Based on Certain Design Criteria" International Journal of Computer Applications 52: 35-42.

[15] Patidar R. and Sharma L., (2011) "Credit card fraud detection using neural network" International Journal of Soft Computing and Engineering (IJSCE)1: 32–38.

[16] Zojaji Z., Atani R. E. and Monadjemi A. H., (2016) "A Survey of Credit Card Fraud Detection Techniques: Data and Technique Oriented Perspective".

[17] Aleskerov E., Freisleben B. and B. Rao, (1997) "Cardwatch : A Neural Network Based Database stern for Credit Card Fraud Detection" In: Proceedings of the IEEE/IAFE 1997 In Computational Intelligence for Financial Engineering (CIFEr), pp. 220–226, New York City, NY, USA.

[18] Syeda M., Zhang Y.-Q. and Pan Y., (2002) "Parallel granular neural networks for fast credit card fraud detection" in Fuzzy Systems, In: Proceedings of the 2002 IEEE International Conference on Fuzzy Systems , pp. 572–577, Honolulu, HI, USA.

[19] Akhilomen J., (2013) "Data mining application for cyber credit-card fraud detection system" in Industrial Conference on Data Mining, In: Proceedings of the World Congress on Engineering 3, pp. 218–228, WCE, London, U.K.

[20] Benaddy M. and Wakrim M., (2012) "Simulated annealing neural network for software failure prediction" International Journal of Software Engineering and Its Applications 6: 35–46.

[21] Khan A. U. S., Akhtar N. and Qureshi M. N., (2014) "Real-time credit-card fraud detection using artificial neural network tuned by simulated annealing algorithm" In: Proceedings of International Conference on Recent Trends in Information, Telecommunication and Computing, ITC, pp. 113–121.

[22] Cortes C. and Vapnik V., (1995) "Support-vector networks" Machine learning 20: 273–297.

[23] Kamboj M. and Gupta S., (2016) "Credit Card Fraud Detection and False Alarms Reduction using Support Vector Machines" International Journal of Advance Research, Ideas and Innovations in Technology, 2.

[24] Srivastava v., Kundu A., Sural S. and Majumdar A., (2008) "Credit card fraud detection using hidden Markov model" IEEE Transactions on dependable and secure computing 5: 37–48.

[25] Gade V. and Chaudhari S., (2012) "Credit card fraud detection using Hidden Markov Model" International Journal of Emerging Technology and Advanced Engineering, 2: 511-513.

[26] Raparty L. V. P. and Nammi S. R., (2012) "Credit Card Fraud Detection Using Hidden Markov Model" International Journal of Advanced and Innovative Research (IJAIR) 1: 212-217.

[27] Bhusari V. and Patil S., (2011) "Application of Hidden Markov Model in credit card fraud detection" International Journal of Distributed and Parallel Systems 2: 203-211.

[28] Aamodt A. and Plaza E., (1994) "Case-based reasoning: Foundational issues" methodological variations, and system approaches, AI communications 7: 39–59.

[29] Wheeler R. and Aitken S., (2000) "Multiple algorithms for fraud detection" Knowledge-Based Systems 13: 93–99.

[30] Khodabakhshi M. and Fartash M., (2016) "Fraud Detection in Banking Using KNN (K-Nearest Neighbor) Algorithm" In: Proceedings of the 5th international conference on research in science and technology, pp 27-34, London, UK.

[31] Leonard K. J., (1993) "Detecting credit card fraud using expert systems" Computers & industrial engineering 25:103–106.

[32] HaratiNik M. R., Akrami M., Khadivi S. and Shajari M., (2012) "FUZZGY: A hybrid model for credit card fraud detection" In: Proceedings of the 2012 IEEE 6th International Symposium on Telecommunications (IST), pp. 1088–1093, Tehran, Iran.

[33] Shen A., Tong R. and Deng Y., (2007) "Application of classification models on credit card fraud detection" In: Proceedings of the 2007 IEEE International Conference on Service Systems and Service Management, pp 1- 4, Chengdu, China.

[34] Hu H.-W., Chen Y.-L. and Tang K., (2009) "A dynamic discretization approach for constructing decision trees with a continuous label" IEEE Transactions on Knowledge and Data Engineering 21: 1505–1514.

[35] Patil D. D., Wadhai V. M. and Gokhale J. A., (2010) "Evaluation of decision tree pruning algorithms for complexity and classification accuracy" International Journal of Computer Applications 11.

[36] Bentley P. J., Kim J., Jung G.-H. and Choi J.-U., (2000) "Fuzzy darwinian detection of credit card fraud" In: Proceedings of the 14th Annual Fall Symposium of the Korean Information Processing Society.

[37] Sánchez D., Vila M. A., Cerda L. and Serrano J.-M., (2009) "Association rules applied to credit card fraud detection" Expert systems with applications 36: 3630–3640.

[38] Lee C.-C., (1990) "Fuzzy logic in control systems: fuzzy logic controller. part II" IEEE Transactions on systems, man, and cybernetics 20: 404–418.

[39] Hagras H., (2007) "Embedding computational intelligence in pervasive spaces" IEEE Pervasive Computing 6.

[40] Karnik N. N., Mendel J. M. and Liang Q., (1999) "Type-2 fuzzy logic systems" IEEE transactions on Fuzzy Systems 7: 643–658.

[41] Mendel J. M., (2003) "Type-2 fuzzy sets: some questions and answers" IEEE Connections, Newsletter of the IEEE Neural Networks Society 1: 10–13.

[42] Hagras H., (2007) "Type-2 FLCs: A new generation of fuzzy controllers" IEEE Computational Intelligence Magazine 2: 30–43.

[43] Li H. L. L. and Lin W. H., (2006) "Type-2 fuzzy logic approach for short-term traffic forecasting" In: Proceedings IEEE transactions on Fuzzy Systems, pp. 33–40.

[44] Kesemen O., Tezel Ö. and Özkul E., (2016) "Fuzzy c-means clustering algorithm for directional data (FCM4DD)" Expert Systems with Applications 58: 76–82.

[45] Bernardo D., Hagras H. and Tsang E., (2013) "A genetic type-2 fuzzy logic based system for the generation of summarised linguistic predictive models for financial applications" Soft Computing A Fusion of Foundations, Methodologies and Applications 17: 2185–2201.

[46] Fawcett T., (2006) "An introduction to ROC analysis" Pattern recognition letters 27: 861–874.

[47] Ishibuchi H, Yamamoto T, (2005) "Rule weight specification in fuzzy rule-based classification systems" IEEE Trans Fuzzy Syst 3(4):428–435.

[48] Sanz J, Fernandez A, Bustince H, Herrera F, (2010) "Improving the performance of fuzzy rule-based classification systems with intervalvalued fuzzy sets and genetic amplitude tuning" Inf Sci 180:3674–3685.

[49] Ishibuchi H, (2001) "Three-objective genetic-based machine learning for linguistic rule extraction" Inf Sci 136:109–133.

[50] Mendel J., (2001) "Uncertain Rule-Based Fuzzy Logic Systems: Introduction& New Directions" Upper Saddle River, NJ: Prentice-Hall.