I.J. Wireless and Microwave Technologies, **2023, 6, 1-14**
Published Online on December 8, 2023 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijwmt.2023.06.01

Interpretable Fuzzy System for Malicious Domain Classification Using Projection Neural Network

Rajan Prasad*
Department of Computer Science and Engineering, Babu Banarasi Das University, Lucknow, India
E-mail: rajan18781@gmail.com
ORCID iD: https://orcid.org/0000-0002-5238-9690
*Corresponding Author

Praveen Kumar Shukla
Department of Computer Science and Engineering, Babu Banarasi Das University, Lucknow, India
Email: drpraveenkumarshukla@gmail.com
ORCID iD: https://orcid.org/0000-0002-9227-8856

Received: 05 February, 2023; Revised: 09 March, 2023; Accepted: 18 April, 2023; Published: 08 December, 2023

Abstract: In this study, we suggest an interpretable fuzzy system for the classification of malicious domains. The proposed system is integration of Sugeno type fuzzy system and projection neural network, the main advantage of interpretable fuzzy system is to classify the patterns and self-explainable capability. Whereas the projection network is used to exact mapped fuzzy inference rules to the network's projection layer. On the other hands, the system is able to deal with large amount of real-time data. The proposed model is tested malicious URL datasets collected from Alexa. The experimental results show that the system is able to classify malicious domain with high accuracy and interpretability as compared to existing methods. The proposed model is usefull to classify malicious attacks and explain the couses behind the decision. The evaluation of model based on confusion matrices, ROC and the nauck index is used for the interpretability assessments.

Index Terms: DGA domain classification, interpretable neuro-fuzzy system, malicious domain, projection network

1. Introduction

As a result of the COVID-19 epidemic, the global growth of the internet has enhanced the network's complexity. On the other hand, hackers take advantage of this position by targeting websites, compromising company privacy, or manipulating individuals through phishing websites [1]. According to a recent International Data Corporation (IDC) [2] study for 2021, 87% of firms were subjected to DNS assaults. Botnets are the principal source of malicious domains; they are a network of personal computers infected with malware and controlled by a command and control server [3].Botnet configurations include peer-to-peer, client-server or combination of both. Generally, the peer-to-peer topology is used to transmit messages from botnet devices to victim machines. According to the existing literature [4 - 14], a botnet will not directly contact the C&C server, but will instead listen for connections and orders from a specific server. Many static and dynamic analysis-based classification techniques are reported for the development of efficient malware classification system

In this paper, we propose a method of malicious domain classification using a neuro-fuzzy network that is based on Sugeno-type fuzzy inference rules and a projection neural network. First, we extract all essential features of the sample domain name dataset using various types of statistical analysis, after that using correlation map to select the most relevant features of the newly created dataset. Next, we develop a fuzzy rule-base and calculate the membership functions of each feature. Finally, we applied fuzzified dataset into proposed model for the classification of malicious domains.

The main contributions in this article as:

i. We suggest a novel interpretable neuro fuzzy-model for classification of malicious domain.
ii. Successfully integration of Sugeno fuzzy model into projection neural network.

The research work is mainly focused on the classification of malicious domains. The sections are arranged in the following order: Section 2 presents some recent related works, Section 3 presents the approch of calculating features, Section 4 presents the proposed model for malicious domain classification; Section 5 simulation results section 6 covered performance evaluation, and Section 7 present the conclusion and future research directions.

2. Related Work

The ability to detect malicious use of DNS is important in the context of security[15-16]. Several articles have been written on DGA domains in recent years, including Manasrah et al. [17] discussed exploiting botnet activity via DNS characteristics. This method removes the requirement to keep spam filters or modify bot lists. The approach takes advantage of features of DNS traffic like server name records, IP addresses, domain name life spans, and characters in the domain name. The Naive Bayes method is used to perform classification. The reliability of this strategy is poor because it depends on a mix of data sets and hence cannot be used for detection with merely subjective analysis. Rajalakshmi et al. [18] suggested a hybrid methodology for DGA domain classification. The approach employed visual signals such as letters, solid, and numerals characters to assist humans in recognizing entities. Based on a unique n-gram feature, Nagunwa T et al. [19] proposed a technique for identifying suspicious domain names. Nguyen Quoc K et. al [20] discovered a novel method for identifying domain names based on SVM and n-gram distributions of data sets. The latent semantic features of the domain name are extracted, filtered by a threshold, and saved in an external database for later use in identification procedures. Schiavoni et al. [21] offer a Phoenix technique based on domain name semantics and IP-based attributes to recognize domains produced by DGA. Botnet detection models are designed to detect botnets at the network level. More particularly, we offer a new approach to detecting botnets that involves a domain generation algorithm (DGA). During this step, the system uses the Mahalanobis distance algorithm to analyze compatibility and IP addresses to cluster DGA domains. Antonakakis et al. [22] proposed a malware detection model based on semantic and syntactic characteristics, structural domain characteristics, and the X-means algorithm. The proposed models are compared for both their classification efficiency and accuracy based on detection performance compared with other models in literature.These methods have several advantages and disadvantages, such as the feature selection method, which can solve many problems. However, the authors are using the data on a small scale to solve the problem[23]. designed an algorithm to detect domains in a certain manner. However, It only uses one data set and lacks key important DGA identification features, such as truncated domain names. Data clustering has numerous uses in data mining, including detection and segmentation, knowledge discovery, and machine learning. It is an essential part of the DGA problem. In practice, data sets contain missing, ambiguous, or uncertain values.

Zadeh invented fuzzy set theory to overcome this problem by modeling uncertain information in terms of element membership in a set. It is now used in different fields to solve real-world problems. The foundation of the traditional fuzzy set theory is to provide a framework for capturing a particular set of concepts and relationships. Zadeh's theoretical approach is based on the notion that humans use such fuzzy sets to understand, represent, and communicate information. Unfortunately, there are some serious limitations regarding the representation of "non-affinity" and "hesitation" in addition to the inability to capture non-binary values such as "yes" or "no". While the malicious domains are designed to elicit an alert response from the control server, they may still be able to connect to it with a DGA Domain. In [24], the authors presented a base set of features based on current segmentation models and enhanced the results by aggregating processes, but numerous issues remain unresolved, including the tagging of domain names with confusing phrases and the time necessary to calculate characteristic values. Based on the shortcomings of the preceding literature, we chose two models to improve their flaws. We reduced calculation time and saved storage space by deleting several features without affecting their clustering results. Correlation matrices are used to select relevant information and save computation time. Part 3 of this article goes over this method.

3. Approach

3.1 Feature extraction from domains

In this section we explain the suggested model and datasets used for the experiments. The suggested model divided in to five steps, each step has specific task as described below:

Structural features is a static behaviours of the domain name such as 'Domain length', 'Numbers of sub-domins', 'Mean of sub-domin' , 'Has www Prefix', 'Contains top level domain as sub-domain', 'Underscore Ratio', 'Contain IP Address', and 'Having a valid top level domain'.

structural characteristics of valid and invalid domain are depicted in table1 in details and computing procedure of each feacture are described as follows:

Let, "aifoundation.in", "apps.apple.com", "aifoundation.net" are valid domain name and "rpkbbuutirajan.tw" is a DGA generated domain name.

Table 1.Structural-based charcteristics

Feature	Meaning	aifoundation.in (Normal Domain)	rpkbbuutirajan.tw (DGA Domain)
Nos	Number of Subdomains	1	1
DNL	Domain Name Length	15	17
HwP	Has www Prefix	0	0
CTS	Contains Top Level Domain as Sub-Domain	0	0
UR	Underscore Ratio	0	0
HVTLD	Has a valid Top Level Domain	1	0
SLM	Sub-Domain Length Mean	12	14

VRLD: A valid root level domain and CTS: Contains Top Level Domain as subdomain that has been authenticated from the root-zone database; if the domain is not registered in the root-zone database, its bit value is zero; if it has been registered, its bit value is one.

Uunder score ratio (UR) can be calculated by using the Eq.(1)

$$UR = \frac{\sum count(\_)}{len(domain)} \tag{1}$$

Grammar-based characteristics include containing a digit, a vowel, and a digit ratio, which may be written as:

If the domain "xaronet.edu.in" has no digits and the domain "6rppd.ru" has digits, we shall label the relevant values as false and true, respectively. Similarily, Vowel ratio and digit ratio can be calculated using Eq.(2) and (3) respectively. Table 3. Shows the grammer-based parameters of two different domain names.

$$Vowel\_ratio = \frac{\sum count(Vowel(domain))}{len(domain)} \tag{2}$$

$$Digit_{ratio} = \frac{\sum count\big(Digit(domain)\big)}{len(domain)} \tag{3}$$

Semantic based features includes repeted charcters ratio in sub-domian,consecutive consonant ratio,consecutive vowels ratio, and entropy of subdomain. These values can be obtains by using the following manners.

$$Repeated_{ratio} = \frac{\sum count\big(repeated(domain)\big)}{len(domain)} \tag{4}$$

$$consecutive\_consonants\_ratio = \frac{\sum count\ \big(consonents\ (domain)\big)}{len(domain)} \tag{5}$$

Table 2. Grammer based charcteristics

Feature	Meaning	aifoundation.in (Normal Domain)	rpkbbuutirajan.tw (DGA Domain)
CD	Contains Digit	0	0
VR	Vowel ratio	0.2222222	0.166668
DR	Digit ratio	0	0

Table 3. Semantic statistic charcteristics

Feacture	Meaning	aifoundation.in (Normal Domain)	rpkbbuutirajan.tw (DGA Domain)
RRC	The proportion of repeated characters in a subdomain	0.285714	0.428571
RCC	The proportion of consonants that come after each other	0.555558	0.583333
RCD	The number of consecutive digits ratio	0	0
Entropy	The subdomain's entropy	2.725482	2.584963

$$consecutive\_digit\_ratio = \frac{\sum count(Cdigits(domain))}{len(domain)} \tag{6}$$

the entropy of subdomain: the formula determines using Eq.(7).

$$E(d) = -\sum_{t \in p} \frac{count(t)}{len(domain)} * log\left(\frac{count(t)}{len(domain)}\right) \tag{7}$$

In this case, t is a domain character and p is a group of characters.

In addition to the previously stated lexical properties, each domain name's Shannon's entropy is calculated. Obfuscation is used by cybercriminals to fool and entice visitors by replicating acceptable URLs or masking problematic ones. As a consequence, the randomization component of each URL was included in our analysis. which revealed that fraudulent URLs had greater entropy calculations on average than valid URLs. The following equation is used to compute Shannon's entropy[25].

$$H(x) = -\sum_{i=0}^{n-1} p(x_i) \log_b p(x_i) \tag{8}$$

where H(x) is the Shannon entropy and x is the string under consideration. A higher H(x) value suggests that string X is more random. similarily, we can compute Bigram[15] by using the Eq.(8)

$$P(w_1^n) = \prod_{k=1}^{n} P(w_k | w_{k-1}) \tag{9}$$

Table 3. Represent the calculated features values of two different domains.

3.2 Characteristic selection

The process of picking a subset of properties with a high associated value is known as feature selection. Using Eq, we use the pearson technique to compute the value of the correlation coefficient by using Eq. (10). The Pearson technique looks like this: We have the following for two components x and y:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})}.\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})}} \tag{10}$$

Here, n is the number of elements, x_i ,y_i element x_i and y_i

$$\bar{\mathrm{x}} = \frac{1}{\mathrm{n}}\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{x_i} \tag{11}$$

$\bar{\mathrm{y}}$ the average value of y

3.3 Pattern classification

The classification method may be described broadly as the assignment of items to classes in such a way that components within one group are as similar as feasible to one another while being as distinct from objects in other groups as possible. The more previous knowledge about the issue area that is accessible, the better the classification algorithm may be modified to fit the real circumstances. For example, if the a priori probability and dependent density of all classes in a given set are known, Bayes decision theory provides optimal solutions by reducing the expected classification error [26]. In many pattern recognition scenarios, however, an input pattern's categorization is dependent on data with small sample sizes for each class, which may not be indicative of the underlying probability distributions, even if they are known. Many techniques, such as grouping and analysis of variance, which rely on some concept of similarity [27], have been used in this situation.

3.3 Fuzzy Sets

Zadeh first introduced the concept of fuzzy sets in 1965 [28,29]. Since then, researchers have discovered several methods to apply this theory to expand current approaches and develop new theory,tools and techniques for dicision making as well as pattern recognitions [30]. Bezdek believes [31] that applying fuzzy logic concepts to input vector may results more interesting and useful. Bezdek's research focuses on integrating fuzzy set techniques into classic k-NN decision-making algorithms. In fuzzy k-NN algorithm uses fuzzy memberships for pattern classification. Moreover, there are several classification algorithms are availables in literature that are based on fuzzy logic concepts. In this study, we applied fuzzy logic concept in neural network with projection layer to make it more powerfull and accurate.

3.4 Interpretable fuzzy systems

This section describes the fuzzy system, which is intended to demonstrate features of developing interpretable fuzzy systems[32-38]. This system could have several inputs and outputs and the mapping of the system is projected by the Eq.(12)

$$\mathrm{X} \rightarrow \mathrm{Y} \tag{12}$$

Here, $\mathrm{X} = \mathrm{x}_1, \mathrm{x}_2, \dots, \mathrm{x}_\mathrm{n} \subset \mathrm{R}^\mathrm{n}$, $\mathrm{Y} = \mathrm{y}_1, \mathrm{y}_2, \dots, \mathrm{y}_\mathrm{n} \subset \mathrm{R}^\mathrm{m}$ denotes the set of relation of input and output vectors

The structures of fuzzy system consists of four blocks: First,fuzzification provides a conversion of crisp sets $\mathrm{X} \subset \mathrm{R}^\mathrm{n}$ to fuzzy sets specified in X; as a result of fuzzification, numeric values can be given at system inputs. Singleton fuzzification operation maps $\bar{x} = [\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n] \in X$ to fuzzy set $A' \subseteq X$ by using Eq.(13) Boths singleton and non-singleton fuzzification act as filter in fuzzy system.

$$\mu_{A'}(x) = \begin{cases} 1 & \text{if } x = \bar{x} \\ 0 & \text{if } x \neq \bar{x} \end{cases} \tag{13}$$

A fuzzy rule base is a collection of n numbers of fuzzy rules and these rules are represented in the form of fuzzy relation in the set $X \times Y$. As demonstrated in Eq.(14)

$$R^k\colon \begin{pmatrix} \text{if} \left(x_1 \text{ is } A_1^k\right) \text{AND}\dots\text{AND}\left(x_n \text{ is } A_n^k\right) \\ \text{then}\left(y_1 \text{ is } B_1^k\right),\dots,\left(y_m \text{ is } B_m^k\right) \end{pmatrix} \tag{14}$$

Here, , m indicates number of system inputs, outputs respectively, $x = [x_1, x_2, \dots, x_n] \in X$, $y = [y_1, y_2, \dots, y_m] \in Y$ represents the vector of linguistic value of inputs and outputs.

$A_1^k, A_2^k, \dots, A_n^k (k = 1, \dots, n)$, $B_1^k, B_2^k, \dots, B_m^k (k = 1, \dots, m)$ are representing the input and output fuzzy linguistic variables, where as, $\mu_{A_1^k}(A_i)$, $\mu_{B_1^k}(B_i)$ are indicates membership functions of input and output fuzzy sets. These groups are denoted by phrases such as "low," "medium," and "high," among others. Artificial neural networks are incapable of handling certain verbal terms.

Second,the inference block uses fuzzy input values to produce fuzzy output. Initially, fuzzy results have been obtained from the fuzzy inference block denoted as fuzzy sets $\bar{B}_j^k$ independently.

The fuzzy set $\bar{B}_j^k$ can be represented as:

$$\bar{B}_j^k = A' \circ \left(A^k \longrightarrow B_j^k\right)$$

Here, $A^k \longrightarrow B_j^k$ indicates the fuzzy relation in fuzzy rule R^k and $A^k = A_1^k \times A_2^k \times \dots \times A_n^k$ is a Cartesian product of fuzzy sets $A_1^k, \dots, A_n^k$. the membership values of the B_j^k is calculated by Eq.(15)

$$\mu_{\bar{B}_j^k}(y_i) = \sup_{x \in X} \left\{ T \left\{ \mu_{A'}(X), \mu_{A^k \to B_j^k}\left(X, y_j\right) \right\} \right\} \tag{15}$$

Here, t-norm $T\{\cdot\}$ is a conjuction operator.

Singleton defuzzification and a t-norm boundary condition can be used to minimize the dependency, as shown by Eq. (16).

$$\mu_{\bar{B}_j^k}(y_i) = \mu_{A^k} \to B_j^k\left(\bar{x}, y_j\right) = I\left(\mu_{A^k}(\bar{x}), \mu_{B_j^k}\left(y_j\right)\right) \tag{16}$$

Here, $I(\cdot)$ is a type of reasoning operator. For Improving the accuracy of the inference operator employed in eq.dd play key role for influencing interpretability.

Notation A^k is known as an activity level of rule R^k and computed as follows:

$$\mu_{A^k}(\bar{X}) = \mathop{T}_{i=1}^{n} \left\{ \mu_{A^k}(\bar{x}_i) \right\} = \tau_k(\bar{x}) \tag{17}$$

The investigation of rule activity with the precision of the aggregate operator has a substantial impact on interpretability.

Third, the fuzzy results from the inferences block's fuzzy rules $\bar{B}_j^k$ are aggregated to provide the entire rules base's fuzzy result.

The function of the defuzzification block is to transform fuzzy values to crisp values. In fact, several defuzzification operators exist, and new ones are continually being developed. For the purpose of clarity, we've assumed that the inference will be performed using centre of area method. It is shown in Eq.(18)

$$\bar{y}_j = \frac{\sum_{r=1}^{N} \bar{y}_{j,r}^B \cdot \mu_{B_j'}\left(\bar{y}_{j,r}^B\right)}{\sum_{r=1}^{N} \mu_{B_j'}\left(\bar{y}_{j,r}^B\right)} \tag{18}$$

Here, $\bar{y}_{j,r}^B$ represent descretization points of fuzzy set B_j', It should be emphasised, therefore, the majority of defuzzification approaches are dependent on the number N of system rulesIt represents an inverse correlation between defuzzification operators and the total number of fuzzy rules. An examination of this dilemma is a crucial component of affecting interpretability.

3.5 Projection neural network

The classic backpropagation training approach [39] suffers from long training periods, the risk of being stranded at local minima, and the need for a large number of hidden nodes to tackle challenging problems. It provides the added benefit of guaranteed mistake reduction. When a classification job is performed on a statistically significant training dataset, the network output will resemble the Bayian distribution.

In contrast, fast-training classification algorithms do not guarantee a reduction in classification errors. Some examples are restricted Coulomb energy networks (RCE) [40], adaptive resonance theory (ART) models [41], and Kohonen-type networks [42]. As a result, we used a projection network in our study that combines the advantages of RCE with the backpropagation technique. Classification algorithms that allow for quick training do so by creating prototypes with a variety of data points. On the other hand, radial basis functions can also be used to train the model, and the benefits of the radial basis function are that they quickly reduce the losses during training [43].

In comparison, the projection network [44] implements radial basis functions and employs a consistent method for training such hyper-parameters: backpropagation learning of feed-forward neural connection weights and threshold [45]. This successfully leads to model improvement. As a result, after building a neural network using this method, customized back-propagation algorithm training is used to alter connection weights and thresholds to reduce errors. Because the network starts close to a suitable solution, the lengthy training period required by traditional backpropagation is avoided, as is the possibility of being caught in local minima that prevent one from reaching this point. This method enables rapid prototype generation through initialization and subsequent optimization through backpropagation training. A typical projection network structure is depicted in Fig.1.

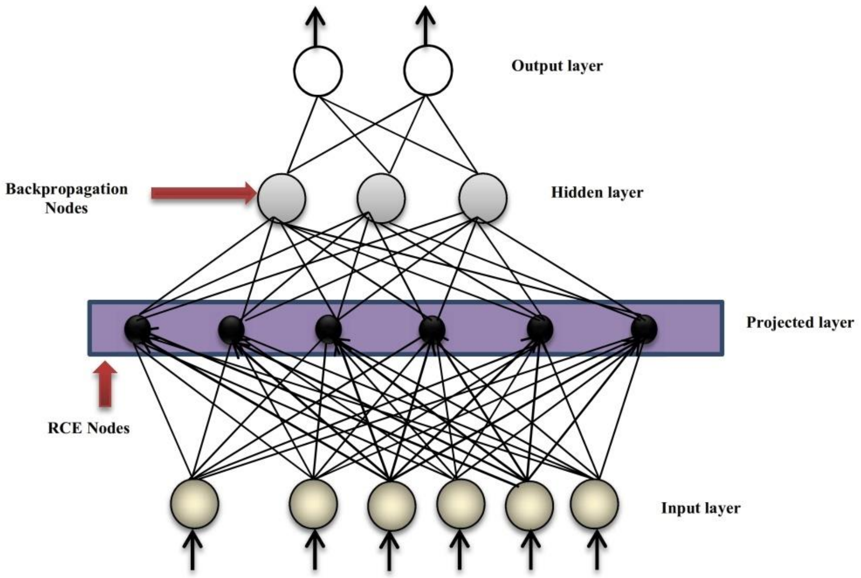


Fig.1. Architecture of a projection neural network using a fusion of RCE nodes and a backpropagation network

4. Proposed Model

In this section, we describe the archicture of suggested model (see fig.3) module by module. Thinking about the inherent benefits of neural networks against statistical approaches. The major goal of this research is to empirically determine how well this strategy works as a classifier for harmful domain categorization. We have used three criteria to extract the featurs from dataset. First structure based, second grammar based, and semantics based (see section 3). By using fuzzy logic concept in to projection neural network, As shown in Fig. 5, we suggest a homogenous architecture in which fuzzy ideas are generated simply by turning input attribute values into fuzzified data, which is then fed into the projection neural network. In the projection layer, we utilized a Sugeno fuzzy inference rule that is relatively basic, fuzzy functionality as inputs, and the network's performance increased. The simulation results indicate that the proposed model outperformed as compaired to the current techniques. The recommended model is as follows:

4.1 Input and fuzzy module

After applying the extraction and feature selection methods, the fuzzy data are used as inputs to the neural network in the proposed neuro-fuzzy model. When feature values vary widely, it might be challenging to categorize things appropriately.

To overcome this problem, we first transform each domain into three linguistic terms [46], and then we use the projection network to train with these linguistic terms. Finally, we employ the developed neuro-fuzzy method to classify malicious domains.

As illustrated in Fig. 4, we have used the trapezoidal, triangular, and gaussian membership functions to express fuzzy phenomena. We used the MAX-MIN technique to transform normalized characteristics into fuzzy data [47]. As shown in Fig. 3, we modeled three membership functions, denoted by the linguistic terms "small," "medium," and "large." The high values of linguistic phrases represent the harmful domain.

4.2 Neural network module

This module consists of four layers: input, projection, hidden, and output. as depicted in Fig. 4. A standard feedforward artificial neural network is used in the neural network module. In this study, a basic projection network is used. The main objective of the projection layer is to map the fuzzy inference rule (see table 3) from the input to the projection layer. We used Reduced Coulomb Energy (RCE) nodes in this process to ensure that the network adopts modifications and maps premise and consequence values. The inaccuracy is employed to figure out the number of neurons in the input, output, and hidden layers. A log-sigmoid transfer function is used by the output neuron. For faster convergence, updated backpropagation training rules are used.

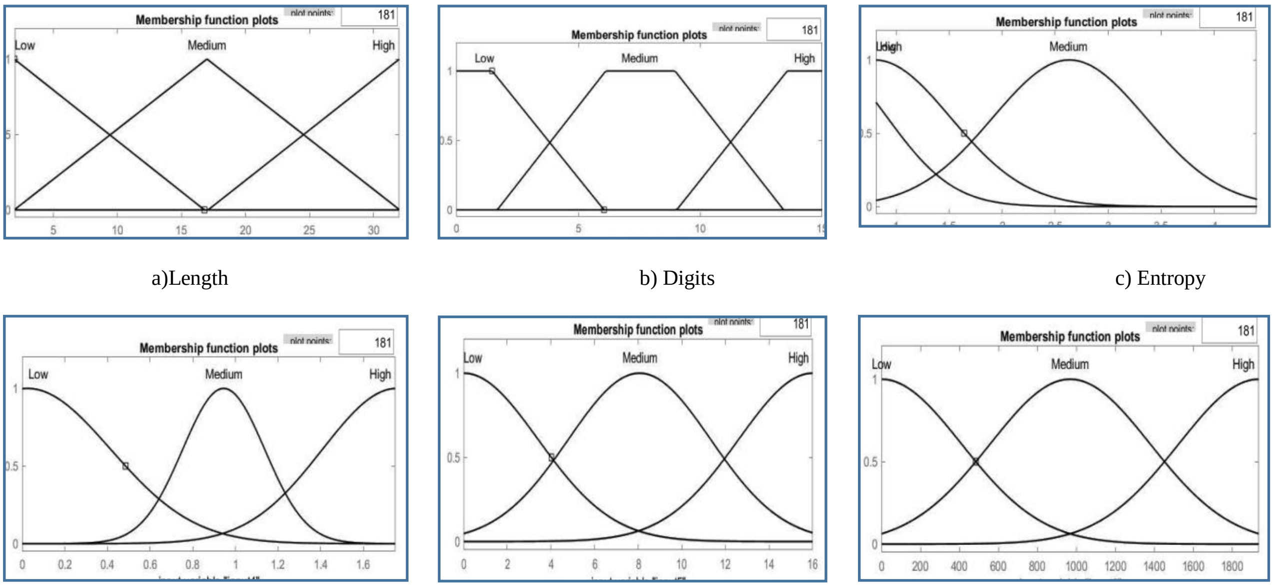
4.2 Linguistic description of the output class

A neural network undergoes two steps in general: training and testing. The supervised learning approach is used to train the model during the training phase. Instead of picking the node with the greatest activation value, every network output might be allocated a membership that is higher than zero. It enables the modelling of fuzzy data when the feature space contains spanning pattern classes. by allowing a pattern point to correspond to more than one class with non-zero membership. Each inaccuracy in membership assignment is recycled instantly during training, and the network's connection weights are adjusted accordingly. The backpropagated error, which is a membership value indicating how much the input vector belongs to a certain class, is calculated for each intended output. The testing phase of a fuzzy network is comparable to that of a regular neural network.

Assume, M_{kj} and μ_{jk} Specify the mean and standard deviation for the jth input of the kth training data sample. In the case of an m-class problem with an n-dimensional feature set. The weighted distance of the i^{th} training sequence vector F_i from the k^{th} class is given by Z_{ik}.

$$Z_{ik} = \sqrt{\sum_{j=1}^{n}\left[\frac{F_{ij}-M_{kj}}{\mu_{kj}}\right]^2 \ for\ k = 1, \dots, m\ and\ j = 1, \dots, n.} \tag{19}$$

The weight $1/\mu_{kj}$ compensates for class variance, thus a trait with a greater variance has less relevance.



a)Length

b) Digits

c) Entropy

d) Vowel-cons

e) FirstDigitIndex

f) Ngrams

Fig. 2. Modeling of input six input variables into fuzzy membership functions

Defining a class.The membership of the i^{th} pattern to class C_k is defined in Eq.(2)

$$\mu_k(F_i) = \left(\frac{Z_{ik} - \min_k(Z_{ik})}{\max_k(Z_{ik}) - \min_k(Z_{ik})}\right) \text{ for } k = 1, \ldots, m \quad (20)$$

$\mu_k(F_i)$ is obviously in the interval [0,1].The training process and network structure are identical to those of the artificial neural network classifier, except for the fuzzy membership values in the output layer.

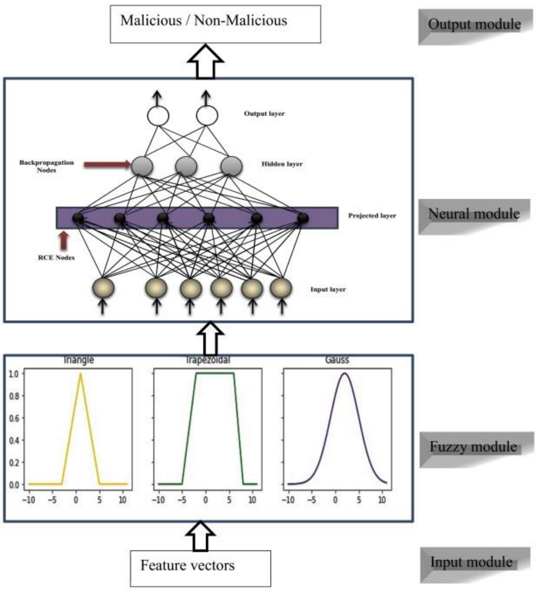


Fig. 3. The structure of a neuro-fuzzy neural network using fuzzy inputs.

Table 3. Sugeno type fuzzy rules generated by the proposed model

1	If (length is Low) and (digits is Low) and (entropy is Low) and (vowel-cons is Low) and (firstDigitIndex is Low) and (ngrams is Low) then (ISDGA is out1mf1) (1)
2	If (length is Low) and (digits is Low) and (entropy is Low) and (vowel-cons is Low) and (firstDigitIndex is Low) and (ngrams is Medium) then (ISDGA is out1mf2) (1)
3	If (length is Low) and (digits is Low) and (entropy is Low) and (vowel-cons is Low) and (firstDigitIndex is Low) and (ngrams is High) then (ISDGA is out1mf3) (1)
4	If (length is Low) and (digits is Low) and (entropy is Low) and (vowel-cons is Low) and (firstDigitIndex is Medium) and (ngrams is Low) then (ISDGA is out1mf4) (0)
5	If (length is Low) and (digits is Low) and (entropy is Low) and (vowel-cons is Low) and (firstDigitIndex is Medium) and (ngrams is Medium) then (ISDGA is out1mf5) (0)
6	If (length is Low) and (digits is Low) and (entropy is Low) and (vowel-cons is Low) and (firstDigitIndex is Medium) and (ngrams is High) then (ISDGA is out1mf6) (0)
7	If (length is Low) and (digits is Low) and (entropy is Low) and (vowel-cons is Low) and (firstDigitIndex is High) and (ngrams is Low) then (ISDGA is out1mf7) (0)
.	…
467	If (length is Medium) and (digits is High) and (entropy is High) and (vowel-cons is Low) and (firstDigitIndex is High) and (ngrams is Medium) then (ISDGA is out1mf467) (1)
468	If (length is Medium) and (digits is High) and (entropy is High) and (vowel-cons is Medium) and (firstDigitIndex is Low) and (ngrams is Low) then (ISDGA is out1mf469) (1)
469	If (length is Medium) and (digits is High) and (entropy is High) and (vowel-cons is Medium) and (firstDigitIndex is Low) and (ngrams is Medium) then (ISDGA is out1mf470) (1)
470	If (length is High) and (digits is High) and (entropy is High) and (vowel-cons is High) and (firstDigitIndex is Medium) and (ngrams is Medium) then (ISDGA is out1mf725) (1)

5. Experiments

In this section, we demostrats the simulation outcomes of the suggested model, for the experimental purpose we collect the alexa domain ranking dataset. It is malicious activities tracking websites The dataset contains 1000 sample domains. In this study, we applied six methods to calculate the features such as domain length, domain digits, domain entropy, domain contains vowel-cons, domain firstdigitIndex and domain Ngrams values as described in section 2, next, we applied feactue engineering principle to obtained highly correlated features. Following that, we separated the selected dataset into two parts: 70% of the data was used for training the classifiers, and the remaining 30% was utilised to test the classifiers. A training set of 70% samples is chosen at random from each class (benign and malicious). Furthermore, we obtains fuzzy value by using the six diffrents memberships functions as depicted in fig.2. the linguistic terms are applied to the inputs values of the model. The neural module of the model(see fig. 5) allowed to maps of input linguistic terms to projection network. in this step we used RCE nodes to projection mapping of the fuzzy rules. Backpropagation algorithms is used to optimization of fuzzy rule. Finally, in the output module we obtained the classification outcomes. For the implementation of the model we used python programming language(version 2.7). Fig.6 depict the simulation of results generated by the system. The performance of the model is depicted in fig.7. As seen the ROC is cover value is 0.91 area of the whole area, it means the model achived accuracy is 88%. The study's findings on four classification algorithms demonstrate the size of the classification accuracy difference when compared to expert classification. The total classification accuracy of the proposed neuro-fuzzy projection neural network with Gaussian membership function is determined by constructing a confusion matrix, as shown in table 5.

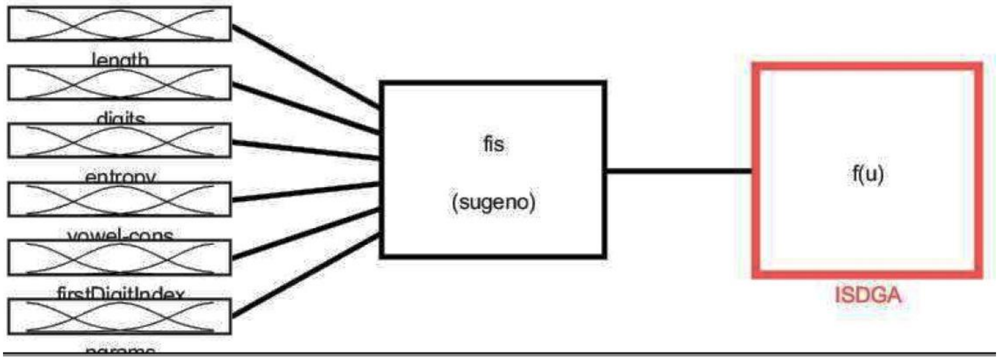


Fig. 4. Simulation of proposed neuro-fuzzy system

The classification results results of the fuzzy approach with three membership functions are shown in table 5. A fuzzy network with Gaussian membership functions has better classification rates than other classifiers, so we used it as a basis for comparison.

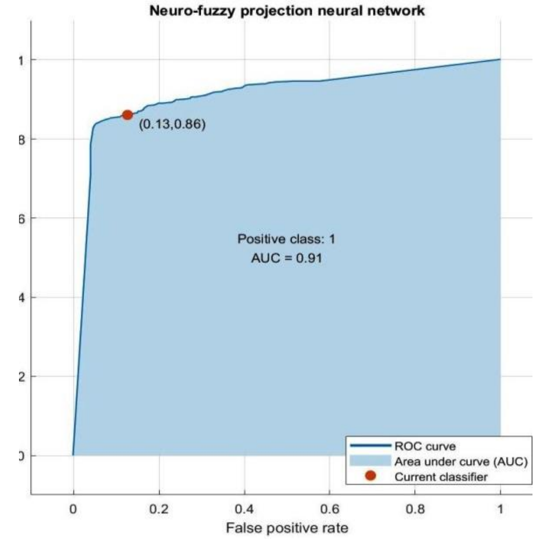


Fig. 5. ROC curve of neuro-fuzzy system

Table 4. Accuracy of neuro-fuzzy projection neural network

Classifier	Membership function	Fuzzy output	Classification Accuracy	
			benign	malicious
1	Triangular_Membership Function	no	91	81
2		yes	92	82
3	Trapezoidal_Membership Function	no	92	83
4		yes	93	84
5	Gaussian_Membership Function	no	92	87
6		yes	94	84

The interpretability of a fuzzy system reflects how easily it can be perceived by individuals[47]. Several scholars have expressed an interest in developing highly interpretable fuzzy models in recent years[48]. However, due to its subjectivity and the tremendous number of components involved, the choice of a suitable interpretability measure is still unknown. Significant research on interpretability metrics has provided interpretability indices for fuzzy systems[49–50]. The Nauck index and the Fuzzy index are the most commonly used interpretability indices.

Nauck et al. introduced a numerical index called the "Nauck Index." It is employed in order to assess the interpretability of the fuzzy system. In this work, we use it to assess the interpretability of the suggested model.

It is a multification of three terms as shown in Eq.(21)

$$\mathrm{Nauck\ index} = \mathrm{comp} \times \overline{\mathrm{cov}} \times \overline{\mathrm{part}} \tag{21}$$

Where, comp represent the complexity of the fuzzy system. It is computed as the following equation(22)

$$\mathrm{comp} = \frac{\mathrm{m}}{\sum_{\mathrm{i=1}}^{\mathrm{r}} \mathrm{n_i}}, \tag{22}$$

Here, m indicates total number of MFs in consequences, and r and ni represents the total number of rules and number of input variables used in the ith rule.

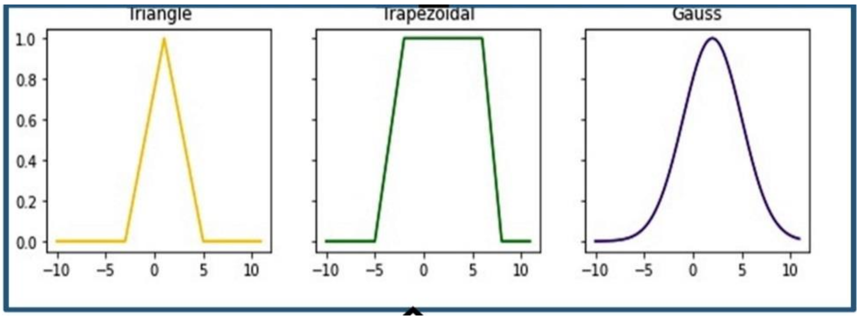


Fig. 6. Membership function showing the full membership values

cov : is a fuzzy partition coverage degree. Suppose that if $\mathrm{Z_l}$ is represent the domain of lth input variable which is partitioned by $\mathrm{d_l}$ membership function $\left\{\mu_{\mathrm{l}}^{1}, \ldots, \mu_{\mathrm{l}}^{\mathrm{d_l}}\right\}$, the coverage degree can be computed as:

$$cov_l = \frac{\int_{Z_l} \hat{h}_i(z)dx}{N_l} \tag{23}$$

$$\hat{h}_l(z) = \begin{cases} h_l(z) & if\ 0 < h_l(z) > 1 \\ \frac{d_l - h_l(z)}{d_l - 1} & otherwise \end{cases} \tag{24}$$

$$\mathrm{h_l(z)} = \sum_{\mathrm{k=1}}^{\mathrm{d_l}} \mu_{\mathrm{l}}^{\mathrm{k}}(\mathrm{z}) \tag{25}$$

Where $\mathrm{h_l(z)}$ is the total MFs of lth input variable with $\mathrm{N_i} = |\mathrm{X}|$, and the coverage $\overline{\mathrm{cov}} = \sum_{\mathrm{i=1}}^{\mathrm{r}} \frac{\mathrm{cov_i}}{\mathrm{n_i}}$, indicates the normalized value of all input variables. fig.6 shown the coverage of input membership functions.

Part: indicates the index partition, it can be computed (see eq.(26)) by taking the inverse of Membership functions and substracted by one for each input variable;

$$\mathrm{part_i} = 1/(\mathrm{p_i} - 1) \tag{26}$$

Where $\mathrm{p_l}$ indicates number of Membership Functions in the $\mathrm{l^{th}}$ input variable.

A fuzzy model is less interpretable if the Nauck index is closer to zero; when the Nauck index value is closer to 1, the fuzzy system is more interpretable. Fig. 7 shows the Nauck indicator.

0
1
Low interpretable
High interpretable

Fig. 7. nauck indicator

Using the above nauck components, we computed the nauck index of the proposed model.similarily we can calculate the nauck indix of fuzzy-knn model that is uased for the classification of malicious domains. The nature of non-fuzzy models are block-box, so that the nauck index values is 0. The final results based on accuracy and interpretability are presents in the table 5

6. Performance Evaluation and Comparison

In this section, we demonstrate the results of other five classifiers and compared our proposed model in term of accuracy.

6.1 k-NN algorithm

K-nearest neighbor (k-NN) is a nonparametric classification method. It is a multivariate normality assumption that offers support to the widely used maximum likelihood estimation method[51-52]. This strategy generates a new input vector y by assigning it the label that appears the most frequently in the K-nearest of all training data. The majority class is determined by analyzing the labels of each of a pattern's k-nearest neighbors.

In practice, one chooses $K = \sqrt[c]{n}$ where c is an appropriate constant and n is the size of training set. In the present study, $c = 1$ is used. In this experiments we got 82.32% accuracy.

6.2 Fuzzy k-NN algorithm

The fuzzy k-NN approach[53] is one of the most accurate pattern recognition algorithms. The standard K-NN method classification rule assigns an uncertain input sample vector y to the class represented by the majority of its k-nearest neighbours. To find the k-nearest neighbours, a labelled data sample is used. The fuzzy k-nn technique gives class membership to a sample observation based on its distance from its k-nearest neighbours as well as their memberships. Fig. 8 depicts the classification error of the fuzzy k-NN. In this experiment, we used the Mahalanobis distance function between point p and the distribution, and the fuzzy-known model had an accuracy of 84.67%. It outperforms the KNN methods.

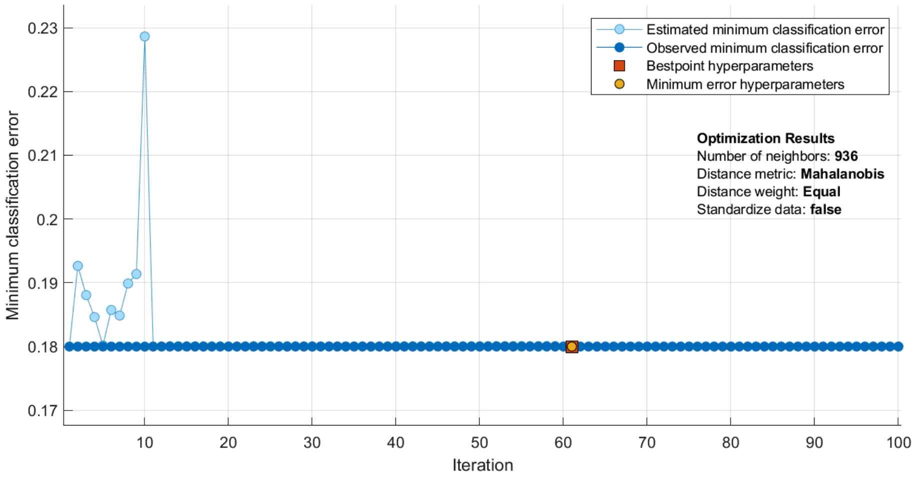


Fig. 8. Classification error of fuzzy k-nn

6.3 Backpropagation neural network

Various machine learning models, including neural networks (both supervised and unsupervised), use different learning strategies for classification tasks. In this experiment, we used an artificial neural network (ANN) based on a supervised learning approach and a backpropagation algorithm[54]. Three equations define the backpropagation method. First, weight connections are adjusted in each learning step (k) using Eq. (27).

$$\Delta w_{ij(k)}^{[s]} = \eta(t)\delta_{pj}^{[s]} x_j^{[s-1]} + m\Delta w_{ij(k-1)}^{[s]} \tag{27}$$

Here, $x_j^{[s]}$ represent the actual output of node j^{th} node in s^{th} layer , and $w_{ij}^{[s]}$ represents the synaptic weight between I^{th} node in $(s-1)^{th}$ layer and j^{th} node in s^{th} layer

Second, the values of output nodes are calculated using Eq.(28)

$$\delta_{pj}^{[o]} = (d_j - o_j) f_j'\left(I_j^{[s]}\right) \tag{28}$$

Here, $\delta_{pj}^{[s]}$ represents the measure of the actual error of j^{th} node and $I_j^{[s]}$ represent the weighted sum of j^{th} node at s^{th} layer.

And third, the rest of nodes calculated as difine in Eq.(29)

$$\delta_{pj}^{[s]} = f_j'\left(I_j^{[s]}\right) \sum_k \delta_{pk}^{[s+1]} w_{jk}^{[s+1]} \tag{29}$$

Here, $\eta(t)$ represent the learning rate that varies throughout time, $f(\quad)$ represents the transfer function, and d_j, o_j= represents the desired and actual activity of j^{th} node

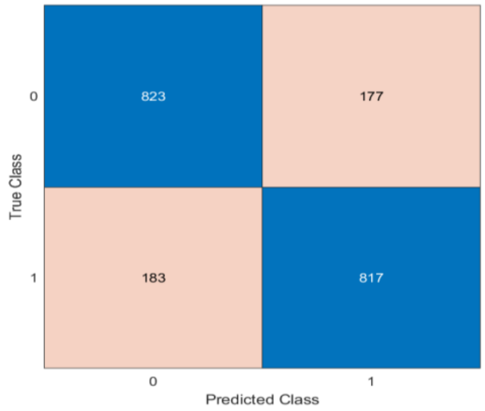


Fig. 9. Classification matrix of the Conventional Backpropagation network

Where value of $\eta(t)$, m, and h_j are chosen empirically to be those with the highest classification accuracy. both input and output nodes are chosen based on the feature vectors and classification class of the objects to be classified. The classification matrix depicted in Fig. 9 is the result of the Training algorithm network.

Table 5. Performance comparison based on Accuracy and interpretibility

Classifier	Classification methods	Accuracy	Interpretibility
1	k-NN	82.32	0
2	Fuzzy k-NN	84.67	0.40
3	Conventional Backpropagation network	87.50	0
4	Neuro fuzzy projection neural network	91.00	0.37

7. Conclusions and Future Suggestions

In this article, we proposed a neuro-fuzzy classifier that by using the notation of interpretable fuzzy system and projection neural network. The proposed model is used to classify malicious domains. Fuzzy sets are used in the input module as well as in the output module. With this technique, the proposed network can be trained with greater efficiency. In the feature extraction step, we may select six features of the domains based on the grammer and semantics. These feature values are subsequently fuzzified and employed in the classification stage of the neuro-fuzzy network. We compared the findings of proposed model with k-NN, fuzzy k-NN, and conventional neural network based on the accuracy and interpretability parameters. According to simulation findings, the proposed approach, which combines gaussian membership functions with projection neural networks, outperforms previous statistical and neural network methods in classification. The comparison findings demonstrate that the proposed model is a better in accuracy and self-explainble (having ability to explain the outcome) tool for harmful domain categorization. In future we, can apply intuitionistic fuzzy set or neutrosophic set theory and different types of membership functions to improve the performance of the model based on accuracy and interpretability.

References

[1] Sánchez-Paniagua M, Fidalgo E, Alegre E, Alaiz-Rodríguez R. Phishing websites detection using a novel multipurpose Sánchez-Paniagua M, Fidalgo E, Alegre E, Alaiz-Rodríguez R. Phishing websites detection using a novel multipurpose dataset and web technologies features. Expert Systems with Applications. 2022 Nov 30;207:118010.

[2] Nadler A, Bitton R, Brodt O, Shabtai A. On the vulnerability of anti-malware solutions to DNS attacks. Computers & Security. 2022 May 1;116:102687.

[3] Divya T, Amritha PP, Viswanathan S. A model to detect domain names generated by DGA malware. Procedia Computer Science. 2022 Jan 1;215:403-12.

[4] Feily M, Shahrestani A, Ramadass S. A survey of botnet and botnet detection. In2009 Third International Conference on Emerging Security Information, Systems and Technologies 2009 Jun 18 (pp. 268-273). IEEE.

[5] Karasaridis A, Rexroad B, Hoeflin DA. Wide-Scale Botnet Detection and Characterization. HotBots. 2007 Apr 10;7:7-.

[6] Karim A, Salleh RB, Shiraz M, Shah SA, Awan I, Anuar NB. Botnet detection techniques: review, future trends, and issues. Journal of Zhejiang University SCIENCE C. 2014 Nov;15:943-83.

[7] Saad S, Traore I, Ghorbani A, Sayed B, Zhao D, Lu W, Felix J, Hakimian P. Detecting P2P botnets through network behavior analysis and machine learning. In2011 Ninth annual international conference on privacy, security and trust 2011 Jul 19 (pp. 174-180). IEEE.

[8] Binkley JR, Singh S. An algorithm for anomaly-based botnet detection. SRUTI. 2006 Jul 7;6:7-.

[9] Garcia S, Grill M, Stiborek J, Zunino A. An empirical comparison of botnet detection methods. computers & security. 2014 Sep 1;45:100-23.

[10] Cooke E, Jahanian F, McPherson D. The Zombie Roundup: Understanding, Detecting, and Disrupting Botnets. SRUTI. 2005 Jul 7;5:6-.

[11] Alieyan K, ALmomani A, Manasrah A, Kadhum MM. A survey of botnet detection based on DNS. Neural Computing and Applications. 2017 Jul;28:1541-58.

[12] Eslahi M, Salleh R, Anuar NB. Bots and botnets: An overview of characteristics, detection and challenges. In2012 IEEE International Conference on Control System, Computing and Engineering 2012 Nov 23 (pp. 349-354). IEEE.

[13] Xie Y, Yu F, Achan K, Panigrahy R, Hulten G, Osipkov I. Spamming botnets: signatures and characteristics. ACM SIGCOMM Computer Communication Review. 2008 Aug 17;38(4):171-82.

[14] Selvaraj NP, Paulraj S, Ramadass P, Kaluri R, Shorfuzzaman M, Alsufyani A, Uddin M. Exposure of botnets in cloud environment by expending trust model with CANFES classification approach. Electronics. 2022 Jul 28;11(15):2350.

[15] Shen WY, Manickam S, Al-Shareeda MA. Review of advanced monitoring mechanisms in peer-to-peer (p2p) botnets. arXiv preprint arXiv:2207.12936. 2022 Jul 17.

[16] Nguyen Quoc K, Bui T, Le D, Tran D, Nguyen T, Nguyen HT. Detecting DGA Botnet based on Malware Behavior Analysis. InProceedings of the 11th International Symposium on Information and Communication Technology 2022 Dec 1 (pp. 158-164).

[17] Manasrah AM, Khdour T, Freehat R. DGA-based botnets detection using DNS traffic mining. Journal of King Saud University-Computer and Information Sciences. 2022 May 1;34(5):2045-61.

[18] Rajalakshmi, R., Ramraj, S., Ramesh Kannan, R. (2019). Transfer Learning Approach for Identification of Malicious Domain Names. In: Thampi, S., Madria, S., Wang, G., Rawat, D., Alcaraz Calero, J. (eds) Security in Computing and Communications. SSCC 2018. Communications in Computer and Information Science, vol 969. Springer, Singapore. https://doi.org/10.1007/978-981-13-5826-5_51.

[19] Nagunwa T, Kearney P, Fouad S. A machine learning approach for detecting fast flux phishing hostnames. Journal of Information Security and Applications. 2022 Mar 1;65:103125.

[20] Nguyen Quoc K, Bui T, Le D, Tran D, Nguyen T, Nguyen HT. Detecting DGA Botnet based on Malware Behavior Analysis. InProceedings of the 11th International Symposium on Information and Communication Technology 2022 Dec 1 (pp. 158-164).

[21] Schiavoni, S., Maggi, F., Cavallaro, L., Zanero, S. (2014). Phoenix: DGA-Based Botnet Tracking and Intelligence. In: Dietrich, S. (eds) Detection of Intrusions and Malware, and Vulnerability Assessment. DIMVA 2014. Lecture Notes in Computer Science, vol 8550. Springer, Cham. https://doi.org/10.1007/978-3-319-08509-8_11.

[22] Antonakakis M, Perdisci R, Vasiloglou N, Lee W. Detecting and tracking the rise of DGA-based malware. ; login:: the magazine of USENIX & SAGE. 2012;37(6):15-24.

[23] Bouveyron C, Brunet-Saumard C. Model-based clustering of high-dimensional data: A review. Computational Statistics & Data Analysis. 2014 Mar 1;71:52-78.

[24] Rényi A. On measures of entropy and information. InProceedings of the fourth Berkeley symposium on mathematical statistics and probability 1961 Jun 20 (Vol. 1, No. 547-561).

[25] Sidorov G, Velasquez F, Stamatatos E, Gelbukh A, Chanona-Hernández L. Syntactic n-grams as machine learning features for natural language processing. Expert Systems with Applications. 2014 Feb 15;41(3):853-60.

[26] Berger JO. Statistical decision theory and Bayesian analysis. Springer Science & Business Media; 2013 Mar 14.

[27] Friedman M. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. Journal of the american statistical association. 1937 Dec 1;32(200):675-701.

[28] Zadeh LA. Fuzzy sets. Information and control. 1965 Jun 1;8(3):338-53.

[29] Pedrycz W, Gomide F. An introduction to fuzzy sets: analysis and design. MIT press; 1998.

[30] Jain AK, Duin RP, Mao J. Statistical pattern recognition: A review. IEEE Transactions on pattern analysis and machine intelligence. 2000 Jan;22(1):4-37.

[31] Bezdek JC. Pattern recognition with fuzzy objective function algorithms. Springer Science & Business Media; 2013 Mar 13.

[32] Zhou SM, Gan JQ. Low-level interpretability and high-level interpretability: a unified view of data-driven interpretable fuzzy system modelling. Fuzzy sets and systems. 2008 Dec 1;159(23):3091-131.

[33] Shukla PK, Tripathi SP. A review on the interpretability-accuracy trade-off in evolutionary multi-objective fuzzy systems (EMOFS). Information. 2012 Jul 12;3(3):256-77.

[34] Shukla PK, Tripathi SP. A new approach for tuning interval type-2 fuzzy knowledge bases using genetic algorithms. Journal of Uncertainty Analysis and Applications. 2014 Dec;2(1):1-5.

[35] Shukla PK, Tripathi SP. Handling high dimensionality and interpretability-accuracy trade-off issues in evolutionary multiobjective fuzzy classifiers. Int. J. Sci. Eng. Res. 2014 Jun;5(6):665-71.

[36] Alonso JM, Castiello C, Mencar C. Interpretability of fuzzy systems: Current research trends and prospects. Springer handbook of computational intelligence. 2015:219-37.

[37] Alonso JM, Magdalena L. Special issue on interpretable fuzzy systems. Information Sciences. 2011 Oct 15;181(20):4331-9.

[38] Nauck D, Kruse R. Obtaining interpretable fuzzy classification rules from medical data. Artificial intelligence in medicine. 1999 Jun 1;16(2):149-69.

[39] Behret H, Korugan A. Performance analysis of a hybrid system under quality impact of returns. Computers & Industrial Engineering. 2009 Mar 1;56(2):507-20.

[40] Cho J, Jung Y, Lee S, Jung Y. Vlsi implementation of restricted coulomb energy neural network with improved learning scheme. Electronics. 2019 May 22;8(5):563.

[41] Grossberg S. Adaptive Resonance Theory: How a brain learns to consciously attend, learn, and recognize a changing world. Neural networks. 2013 Jan 1;37:1-47.

[42] Tsao EC, Bezdek JC, Pal NR. Fuzzy Kohonen clustering networks. Pattern recognition. 1994 May 1;27(5):757-64.

[43] Schwenker F, Kestler HA, Palm G. Three learning phases for radial-basis-function networks. Neural networks. 2001 May 1;14(4-5):439-58.

[44] Xia Y, Leung H, Wang J. A projection neural network and its application to constrained optimization problems. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications. 2002 Apr;49(4):447-58.

[45] Svozil D, Kvasnicka V, Pospichal J. Introduction to multi-layer feed-forward neural networks. Chemometrics and intelligent laboratory systems. 1997 Nov 1;39(1):43-62.

[46] Prasad, R., Shukla, P.K. (2022). A Review on the Hybridization of Fuzzy Systems and Machine Learning Techniques. In: Bansal, J.C., Engelbrecht, A., Shukla, P.K. (eds) Computer Vision and Robotics. Algorithms for Intelligent Systems. Springer, Singapore. https://doi.org/10.1007/978-981-16-8225-4_32

[47] Gacto MJ, Alcalá R, Herrera F. Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures. Information Sciences. 2011 Oct 15;181(20):4340-60.

[48] Pulkkinen P, Hytönen J, Koivisto H. Developing a bioaerosol detector using hybrid genetic fuzzy systems. Engineering Applications of Artificial Intelligence. 2008 Dec 1;21(8):1330-46.

[49] Magdalena L. Semantic interpretability in hierarchical fuzzy systems: Creating semantically decouplable hierarchies. Information Sciences. 2019 Sep 1;496:109-23.

[50] Guo F, Liu J, Li M, Huang T, Zhang Y, Li D, Zhou H. A concise TSK fuzzy ensemble classifier integrating dropout and bagging for high-dimensional problems. IEEE Transactions on Fuzzy Systems. 2021 Aug 20;30(8):3176-90.

[51] KZhang S, Cheng D, Deng Z, Zong M, Deng X. A novel kNN algorithm with data-driven k parameter computation. Pattern Recognition Letters. 2018 Jul 15;109:44-54.

[52] Keller JM, Gray MR, Givens JA. A fuzzy k-nearest neighbor algorithm. IEEE transactions on systems, man, and cybernetics. 1985 Jul(4):580-5.

[53] Kotsiantis SB, Zaharakis I, Pintelas P. Supervised machine learning: A review of classification techniques. Emerging artificial intelligence applications in computer engineering. 2007 Jun 10;160(1):3-24.

[54] Maimon OZ, Rokach L. Data mining with decision trees: theory and applications. World scientific; 2014 Sep 3.

Authors' Profiles

Rajan Prasad received the Bachelor of Technology in Computer Science & Engineering and Master of Technology in Software Engineering degrees from the Babu Banarasi Das University, Lucknow India. He is Research Scholar in Department of Computer Science and Engineering, Babu Banarasi Das University, Lucknow India. His research interest includes Fuzzy Systems, Machine learning and soft computing.

Dr. Praveen Kumar Shukla currently working as a Head of Department in Computer Science and Engineering, Babu Banarasi Das University, Lucknow India. He has guided several PhD Scholars and published more than 30 research articles in reputed journals. His research interests are Fuzzy System, Machine Learning and interdisciplinary areas. He is also the editor of International Conference on Computer Vision and Robotics published in Springer Book Series, ‘Algorithms for Intelligent Systems’.

How to cite this paper: Rajan Prasad, Praveen Kumar Shukla, "Interpretable Fuzzy System for Malicious Domain Classification Using Projection Neural Network", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.13, No.6, pp. 1-14, 2023. DOI:10.5815/ijwmt.2023.06.01