Enhancing Cloud Security with a Fuzzy Rule-Based Classifier for Intrusion Detection and Classification

Dr. Manish Vyas
Associate Professor
Computer Science
Vikrant Institute of Technology and Management, Indore
mai.vyas@gmail.com

Purushothaman
Associate professor
Department of Mathematics
St.Joseph's College of Engineering, Chennai-119
gpmanphd@gmail.com

K.S.Jayareka
Assistant Professor
Department of Computer Science and Engineering,
Sona College of Technology, Salem.
jayareka.ks@sonatech.ac.in

Dr. B. Sankara Babu
Professor, Deptartment of CSE,
Gokaraju Rangaraju Institute of Engineering and Technology (Autonomous),
Bachupally, Telangana
sankarababu.b@griet.ac.in

k.Deepthika,
Assistant Professor
Department of CSE
Dr.N.G.P Institute of Technology, Coimbatore.
deepthi.karuppusamy@gmail.com

Aviral Srivastava
Student
Department of CSE
Amity University Rajasthan
Amity University Rajasthan, NH 11 C, kant kalwar,
Jaipur-300202,
aviral.srivastava6@student.amity.edu

2023 5th International Conference on Inventive Research in Computing Applications (ICIRCA) | 979-8-3503-2142-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICIRCA57980.2023.10220692

Abstract- Cloud computing (CC) is a model of distributed computing that makes it possible to access data, applications, and computer infrastructure through the Internet whenever and wherever it is needed. CC is a method of providing online consumers with access to virtualized, dynamically scaled resources. The importance of safety in this on-demand CC cannot be overstated. That's why the authors of this study are presenting a new method of cloud intrusion detection using fuzzy rules. The IDFRC method can monitor the distributed CC platform for intrusions and protect it from any dangers. Each client has their own unique IDS instance installed, with its own dedicated controller. In order to better detect and categorize intrusions, the authors of this piece provide the improved intrusion detection using Fuzzy Rule-based Classifier (IDFRC) model for use in the cloud. The proposed ID-FRC paradigm seeks to distinguish between malicious and benign cloud-based data flows. The improved results of the suggested method are analyzed by a comprehensive simulation study. A hybrid intelligent fruit fly optimization algorithm (HFOA) is used to fine-tune the FRC model's parameters. The FRC is a powerful paradigm in pattern recognition that provides useful results by the incorporation of language labels into the rules' antecedents. The KDD99 and NSL-KDD dataset is utilized to evaluate the suggested approach. Improvements of the current method over recent state-of-the-art methods were guaranteed by a simulation study of the IDFRC model. Maximum detection performance was reported by the model, with an F-score of 98.5 and a reliability of 100%.

Keywords— Fuzzy Rules Based Classifier, Intrusion Detection, Cloud Computing, Classification, Fruit Fly Optimization Algorithm.

I. Introduction

An enhanced genetic algorithm (EGA) is gaining significant popularity among cloud computing users. Even though several operational and security challenges exist [1], the move to the cloud platform might be less indirect. The growing use of cloud storage highlights the importance of taking precautions to protect data before sending it. The vulnerability of virtual machine (VM) and hypervisor systems to VM-level attacks [2,3] also make them a security risk. There are a few local PC shops that provide supplementary software and equipment [4,5] are part of such a process. Virtual machine (VM) vulnerabilities are exploited by attackers in order to steal information or launch attacks. This happens due to flaws in the way the internet's protocols are designed. Additionally, several new attacks have emerged recently that employ metamorphisms and polymorphisms to evade detection [6]. Flooding assault, and IP spoofing are only some of the network threats that plague cloud computing. Cloud-based data and apps are vulnerable to attacks because hackers target security flaws in the underlying network and protocols. Data for the intended user may become unavailable. Therefore, Intrusion Detection Systems are employed in the cloud computing setting to foil such attempts and protect the cloud from such intrusions.

Attacks against VMs are made possible in an IaaS cloud setting by exploiting and acquiring information about victims' computers. Given the transparency of the cloud framework, and widely scattered, it is an easy target for aggressors [7,8]. As a result, cloud users are vulnerable to both traditional network attacks and cloud-specific attacks. Firewalls and other traditional network security mechanisms are superior at preventing some types of attacks from the outside world. However, these solutions are still inadequate for dealing with attacks from within the network and some complicated attacks from the outside.

Enhanced genetic algorithm (EGA)is the most widely used method in discovering patterns and artificial intelligence with the use of fuzzy rule-based classification systems (FRBCSs) [9]. These systems include language labels in the antecedents of their rules, which contributes to their high performance while also offering interpretable models. Various businesses are succeeding in implementing FRBCSs, from bioinformatics and medicine to finance and economics. Also two distinct groups of difficulties arise from classification tasks.

Classifier learning is more challenging for solving issues with several classes. This is because there is more overlap between different problem classes, making it more difficult to define decision limits. However, many practical issues need to be taken into account in more than one category. As a result, applying decomposition strategies [10] is a simple way to handle multi-class issues, as they enable any classifier to deal with them. The original multi-class problem is broken down into simpler binary ones using the divide-and-conquer methodology, which may then be tackled using separate binary classifiers referred known as base classifiers.

In order to improve the rule foundations' accessibility and settle on a single rule basis, Cintra et al. [11] employed a genetic fuzzy system. In order to test the efficacy of their suggested method, Fazzolari et al. [12] used genetic fuzzy rule-based classification systems to conduct an in-depth study of 36 training set selection techniques. Sanz et al. [13] exploited the concept of interval-valued fuzzy groups in an afterwards gene tuning stage to enhance the performance of fuzzy rule-based classification algorithms. Tuning and rule selection were carried out by Fazzolari et al. [12] using a fuzzy technique that was included into a multi-objective evolutionary algorithm. Stepnicka et al. [14] employed a fuzzy rule4-based approach to aggregate separate projections. Fuzzy rule bases were used to determine the weights of the combination based on time series properties such as trend, seasonality, and stationarity. Particle swarm optimization (PSO) and a heuristic technique were coupled by Alikar et al. [15] to improve the performance of fuzzy rule-driven classification methods on a set of data.

The paper [16] proposed an IDS for cloud environments that uses deep learning. The system was evaluated on a dataset that included different types of network attacks and was compared with other IDS.

In the past several years, locating a model that is both efficient and appropriate to use In regards to the topic of intrusion detection has been considered to be a significant challenge. Researchers have focused their attention on hybrid algorithms, fuzzy approaches, artificial neural networks, genetic algorithms, and the importance of intrusion detection and the fact that information extraction is one of the practical tools that suggests novel trends from the data of massive systems make these two subjects interrelated. In addition, data mining is one of the practical methods that recommends a new pattern from information on mass networks. However, despite the fact that the aforementioned methods are able to generate passable models of intrusion detection, they were not able to get the best possible outcome.

The suggested strategy is predicated on a fusion of anomaly detection and abuse prevention methods. Misuse detection approaches focus on known attack patterns, whereas anomaly detection methods look for unusual activity as evidence of an attack. Fuzzy rule-based classifiers are an intriguing method for detecting and categorizing intrusions. Intrusion detection systems frequently make use of fuzzy logic because of its capacity to describe imprecise and uncertain data.

II. Materials and Method

A. Fuzzy Rule-based system

The classification method used in this research is based on a set of fuzzy if-then rules that are initially built using numerical data. The created rules become potential candidates. The training patterns for an M-class issue in an n-dimensional feature space are assumed to be k real vectors

$$z_s = z_{s1}, z_{s2}, \ldots, z_{sk}, \quad s = 1, 2, \ldots, k \tag{1}$$

Each z_k property is considered to be normalized to the interval [0, 1]. The suggested fuzzy classifier system makes use of fuzzy if-then rules of the subsequent format,

$$if\ z_1 is\ Xc_1, and \ldots\ldots and\ \ z_k\ is\ Xc_k \tag{2}$$

$$then\ class\ l_c\ with\ LF_k \tag{3}$$

The pattern vector is represented by $z = (z_1, z_2, \ldots, z_k)$ and the antecedent fuzzy set is represented by, $X_c = (X_{c1}, X_{c2} \ldots\ldots. X_{ck})$. LF_k is the confidence rule.
Using the product operation, we compute the compatible score of each training pattern X_c with the rule R_c as,

$$\mu_c(X_c) = \prod_{a=1}^{k} \mu_{ca}(X_{ca}) \tag{4}$$

$\mu_{ca}(\cdot)$ *membership function of* antecedent fuzzy set. However, we may define the support of $(LF_k \rightarrow l_c)$ in the following way:

$$G(LF_k \rightarrow l_c) = \frac{\sum_{z_c \in class\, l_c} \mu_c(X_c)}{k} \tag{5}$$

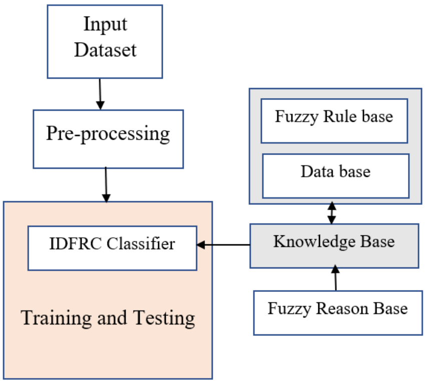

Fig.1. intrusion detection using Fuzzy Rule-based Classifier (IDFRC) model

The confidence is defined as,

$$C(LF_k \to l_c) = \frac{\sum_{z_c \in class l_c} \mu_c(X_c)}{\sum_{s=1}^{k} \mu_c(X_c)} \qquad (6)$$

Designing a classification system involves the challenge of discovering the best possible mapping with respect to some criterion that will ultimately decide the effectiveness of the system. The purpose of this mapping is to construct a classifier that assigns class labels with minimal error throughout the whole feature space, and this is done by starting with correctly labeled instances (training examples). Finally, the real error of the classification system is estimated by calculating the system's performance on testing data. Generally speaking, categorization systems fall into one of two categories based on their intended application: those designed to function independently, and those meant to assist the user in making decisions. In the former, the percentage of right classifications serves as the primary metric of success in the design process. In the latter, other qualities, such as understandability, sturdiness, flexibility, modifiability, and consistency with prior knowledge, may be crucial to the system's eventual acceptance and implementation. The suggested learning technique is geared at creating systems that fall under the second category, which may be thought of as "human-centred," and the framework for fuzzy rule-making utilized and the process by which they are produced are both determined by the desired qualities. Figure 1 presents the method, structure, and application of the IDFRC learning environment.

B. Hybrid intelligent fruit fly optimization algorithm (HFOA)

Particle swarm optimization method and the cognitive and social learning rates and the inertia weight are utilized to update the particle velocities in the hybrid intelligent approach, which uses the same optimizing method seen in fruit flies. The fruit fly optimization feature shifts the flies about by some factor and some randomness. The objective is to optimize the classifier's parameters so that it reliably detects intrusions in the test dataset.

Algorithm 1: Hybrid intelligent fruit fly optimization algorithm (HFOA)

```
Input:
- Dataset: a set of training instances with labeled classes
- MaxIterations: the extreme number of iterations allowed
- PopSize: the population size
- Beta: the scaling factor used to update the position
- C1, C2: the cognitive and social learning rates
- W: the inertia weight
Output:
- BestSolution: the best solution found
1. Initialize population randomly:
  For each i in the population:
    Generate a random position $X_i$
    Evaluate the fitness of $X_i$using a fitness function that
            measures the accuracy of a classifier
2. Initialize the particle swarm:
  For each fruit fly i in the population:
    Generate a random velocity $V_i$
    Set the best position Pbest $_i$ to $X_i$
3. For each iteration t = 1 to MaxIterations:
  For each fruit fly i in the population:
    3.1. Update the fruit fly position:
       Generate a random position R
       Update the position of fruit fly i as follows:
       $X_i = X_i + Beta * (R - X_i) + C1 * rand() *$
    (Pbest $_i$ $- X_i) + C2 * rand() * (BestSolution - X_i)$
    3.2. Evaluate the fitness of the new position $X_i$ using the
    fitness function
    3.3. Update the best position Pbest $_i$ if necessary
    3.4. Update the particle swarm:
       Update the velocity of i as follows:
       $V_i = W * V_i + C1 * rand() *$ (Pbest $_i$ $- X_i) +$
    $C2 * rand() * (BestSolution - X_i)$
       Update the position of i as follows:
       $X_i = X_i + V_i$
       If $X_i$ is out of bounds, reposition it randomly
   3.5. Update the best solution found so far:
      Find the the best fitness value among all
      BestSolution to its position
4. Return BestSolution
```

In the context of intrusion detection and classification, HIFOA can be employed in order to fine-tune the classification variables of a Fuzzy Rule-Based System (FRBC), which is a popular approach for intrusion detection. FRBCs use a set of fuzzy rules to classify network traffic as normal or malicious based on a set of input features. The performance of FRBCs depends on the selection of appropriate fuzzy rules and their associated parameters. HIFOA can be used to optimize the parameters of the FRBC by trying to find the optimal combination of uncertain regulations and their setting parameters that minimize the classification error rate. The algorithm can also be used in order to pick out the best qualities for categorization by evaluating the importance of each feature using a fitness function. The hybrid nature of HIFOA allows it to explore the search space efficiently, making it a promising technique for improving the accuracy of intrusion detection and classification systems. However, several variables, including the selection of optimization parameters, affect how well HIFOA perform and the complexity of the classification problem. Therefore, careful experimentation and tuning are required to obtain optimal results.

C. Dataset

The NSL-KDD data set, suggested by Tavallace et al. [18], is a trimmed-down version of the full-fledged KDD 99 dataset. Due to the manageable size of both the training and test sets, it is feasible to conduct the experiments on the whole data set without resorting to random subset selection. There are a total of 41 feature characteristics in the dataset, 38 of which are numerical and 3 of which are symbolic.

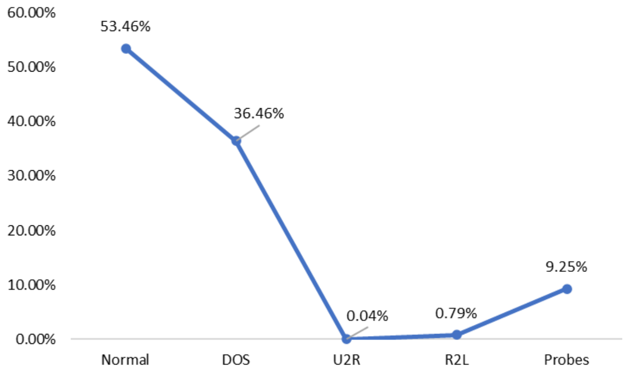

Fig.2. data distribution from the dataset

The fig2. Demonstrate the data distribution form the dataset. There are 125973 recordings in all, 67343 of which are considered "normal" and 58630 of which are considered "attacks." There are a total of 24 distinct attacks included in the dataset, and they can be broken down into four distinct classes: DoS, R2L, U2R, and Probing.

D. Pre-processing

Here are some common preprocessing steps for the KDD99 dataset:

Data cleaning: The KDD99 dataset contains a large number of records, and some of them may be incomplete, inconsistent, or contain errors. Data cleaning involves identifying and correcting these errors, such as missing values, duplicates, and inconsistencies.

Data normalization: The dataset contains a mixture of categorical and numerical features. Data normalization involves scaling the numerical features to a common range, typically between 0 and 1. Because of this, algorithms may function better.

Feature selection: There are a lot of characteristics in the KDD99 dataset, and some of them might not even be useful for detecting intrusions. In the decision-making process, only the most relevant features are kept. As a result, the effectiveness of algorithms can be enhanced by reduced dimensionality of the dataset.

Data balancing: The KDD99 dataset contains a highly imbalanced class distribution, with the majority of records belonging to the normal class and only a small percentage belonging to the various attack classes. Data balancing involves addressing this class imbalance, such as by biased sample that favors one group over another.

Data encoding: It contains categorical features that need to be encoded into numerical values before they can be used. A few examples of popular encoding methods include one-hot encoding and label encoding, and binary encoding.

These preprocessing steps can help to raise the standard and dependability of the KDD99 dataset and make it more suitable for use in intrusion detection tasks.

III. PERFORMANCE EVALUATION

Using a split of the data into a training set and a testing set, the accuracy of the model may be calculated using cross validation. A dataset is split into 10 parts for 10-fold cross validation; nine of these parts are used for training, while the residual part is used for testing. There are ten rounds of this cross-validation procedure. (The number of folds). The data from all 10 samples is then averaged into a single model estimate. While random sub-sampling uses different subsets for training and testing in each iteration, 10-fold cross validation uses the same dataset throughout.

The efficacy of a given intrusion detection model may be gauged by how well it can anticipate such intrusions. Attack classes and normal classes are the primary ones that systems to detect intrusions look for when making distinctions. The confusion matrix details the proportion of correct diagnoses, incorrect diagnoses, false negatives, and true positives.

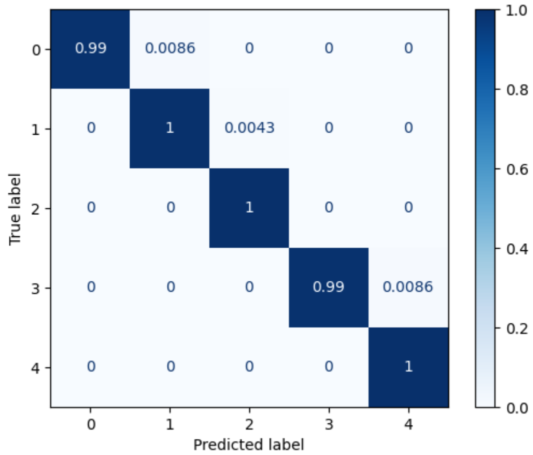

Fig.3. Proposed IDFRC model- confusion matrix

The detection of an assault and subsequent alert is a True Positive (TP). False positive (FP) is an ordinary connection that is mistakenly identified as an intrusion attempt and an alert is triggered. True Negative (TN) is a common link does not trigger any warnings. False negative (FN) is an assault is not recognized and no alert is issued. The table 1 presents the model's results.

TABLE I

PERFORMANCE METRICS OF PROPOSED MODEL

	Accuracy	Precision	Recall	F1-Score
0	0.99	0.99	0.99	0.98
1	0.99	0.99	0.98	0.98
2	1.00	1.00	1.00	1.00
3	1.00	1.00	0.99	0.99
4	1.00	1.00	0.98	0.98

These numbers allow for the following performance metrics to be calculated:

The likelihood that an algorithm makes accurate predictions for both positive and negative instances is measured by its accuracy as shown in fig.4, which is provided by:

$$Accuracy_c = \frac{TR_p + TR_n}{TR_p + TR_n + FA_p + FA_n} \tag{7}$$

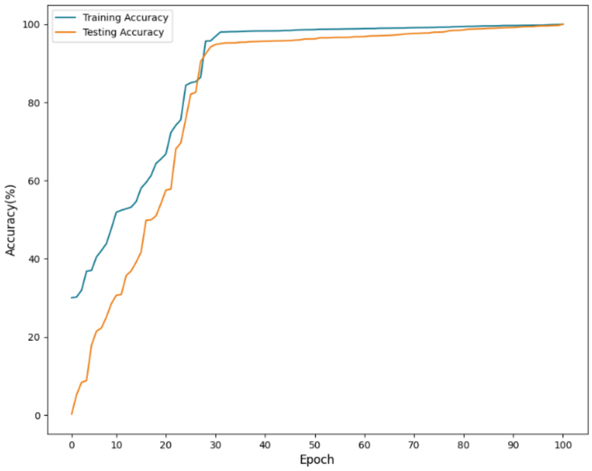

Fig.4. Accuracy of Proposed IDFRC model

The fig.5. shows the loss of the proposed model obtained from training and testing. The term "precision" refers to the rate at which situations are accurately classified, and it is computed as:

$$Precision_c = \frac{TR_p}{TR_p + FA_p} \tag{8}$$

The algorithm's capacity to make accurate predictions of positive cases is quantified by its recall/detection rate, which is defined as,

$$Recall_c = \frac{TR_p}{TR_p + FA_n} \tag{9}$$

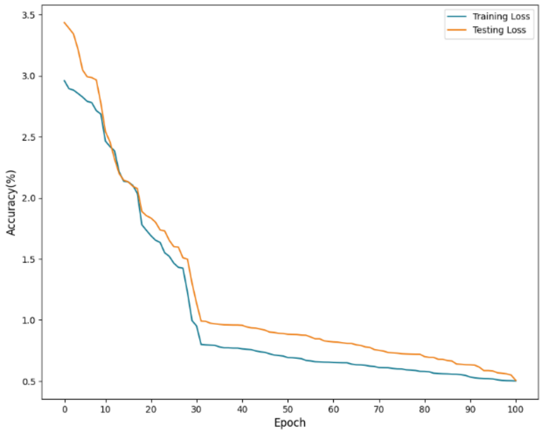

Fig.5 Loss of Proposed IDFRC model

The suggested model is tested in simulation using the Python 3.6.5 tool on a PC with an i5-8600k processor, 4GB of GeForce 1050Ti graphics memory, 16GB of RAM, a 250GB solid-state drive, and a 1TB hard drive. Here, the IDRBC model's effectiveness in identifying intrusions is evaluated using the NSL-KDD data set.
The results of the executed trials show that the IDSFRC model achieved a perfect detection rate of 100%. Based on the calculated performance metrics, it was determined that the IDSFRC model achieved a detectability value that was satisfactory, while also successfully reducing the false positive rate. The overwhelming number of IDS warnings can be mitigated by lowering the false positive rate. Based on the findings above, the IDSFRC model may be a viable option for use as a detection system.

IV. CONCLUSION

In conclusion, a better intrusion detection and sorting system was developed in this work by employing a fuzzy rule-based classifier in a cloud framework. The system was optimized using a hybrid intelligent fruit fly optimization algorithm (HFOA) on the NSL-KDD99 dataset. The proposed system demonstrated superior performance in detecting and classifying network intrusions compared to existing approaches. Studies validated the suggested system's effectiveness, showing it to be accurate to within a small margin of error, resistant to assaults, and with low false-positive and false-negative rates. The HFOA approach improved the performance of the fuzzy rule-based classifier by optimizing the feature selection process and enhancing the classification accuracy. In order to find a concise collection of fuzzy if-then classification rules, this study employs a fuzzy rule-base classification system. Then, a novel genetic-algorithm based approach to rule weights formulation is given. When applied to anomalous rule-based IDS, the suggested technique yields a highly precise and easily explainable fuzzy system for intrusion detection. Additionally, tests are run using the KDD99 data set. When compared to existing algorithms in simulated trials, the suggested technique achieves a greater detection rate and a smaller FAR. It's important to note that no single approach can guarantee optimal performance across the board. Therefore, while assessing the efficacy of intrusion detection systems, it is essential to employ many performance metrics. In the future, we want to use a feature selection approach on an intrusion detection dataset in order to determine which subsets of features are most likely to yield maximum efficacy with minimum effort.

The proposed system can be useful for enhancing the security of cloud-based systems and networks. The study demonstrated that the use of a hybrid intelligent approach can significantly improve the accuracy and efficiency of intrusion detection and classification systems in cloud surroundings. Further research can be done to investigate the pertinency of the projected system in real-world scenarios and to explore its potential in other domains.

REFERENCES

[1] Negi, P.S.; Garg, A.; Lal, R. Intrusion detection and prevention using honeypot network for cloud security. In Proceedings of the 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 29–31 January 2020; pp. 129–132.

[2] Meryem, A.; Ouahidi, B.E. Hybrid intrusion detection system using machine learning. Netw. Secur. 2020, 2020, 8–19.

[3] Achbarou, O.; El Kiram, M.A.; Bourkoukou, O.; Elbouanani, S. A new distributed intrusion detection system based on multi-agent system for cloud environment. Int. J. Commun. Netw. Inf. Secur. 2018,

[4] 10, 526.

[5] Singh, D.A.A.G.; Priyadharshini, R.; Leavline, E.J. Cuckoo optimisation-based intrusion detection system for cloud computing. Int. J. Comput. Netw. Inf. Secur. 2018, 11, 42–49.

[6] Hatef, M.A.; Shaker, V.; Jabbarpour, M.R.; Jung, J.; Zarrabi, H. HIDCC: A hybrid intrusion detection approach in cloud computing. Concurr. Comput. Pract. Exp. 2018, 30, e4171.

[7] Ma, X.; Fu, X.; Luo, B.; Du, X.; Guizani, M. A design of firewall based on feedback of intrusion detection system in cloud environment. In Proceedings of the IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 9–13 December 2019; pp. 1–6.

[8] Fontaine, J.; Kappler, C.; Shahid, A.; Poorter, E.D. Log-based intrusion detection for cloud web applications using machine learning. In Advances on P2P, Parallel, Grid, Cloud and Internet Computing, Proceedings of the 14th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC-2019), Antwerp, Belgium, 7–9 November 2019; Barolli, L., Hellinckx, P., Natwichaivol, J., Eds.; Springer: Cham, Switzerland, 2019; pp. 197–210.

[9] Chang, V.; Golightly, L.; Modesti, P.; Xu, Q.A.; Doan, L.M.T.; Hall, K.; Boddu, S.; Kobusi ´nska, A. A survey on intrusion detection systems for fog and cloud computing. Future Internet 2022, 14, 89.

[10] H. Ishibuchi, T. Nakashima, M. Nii, Classification and Modeling with Linguistic Information Granules: Advanced Approaches to Linguistic Data Mining, Springer-Verlag, 2004.

[11] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, F. Herrera, An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes, Pattern Recognit. 44 (8) (2011) 1761–1776.

[12] Cintra M, Camargo H, Monard M (2016) Genetic generation of fuzzy systems with rule extraction using formal concept analysis. Inf Sci 349:199–215

[13] Fazzolari M, Giglio B, Alcala´ R, Marcelloni F, Herrera F (2013) A study on the application of instance selection techniques in genetic fuzzy rule-based classification systems: accuracy-complexity trade-off. Knowl Based Syst 54:32–41

[14] Sanz J, Ferna´ndez A, Bustince H, Herrera F (2011) A genetic tuning to improve the performance of fuzzy rule-based classification systems with interval-valued fuzzy sets: degree of ignorance and lateral position. Int J Approx Reason 52(6):751–766

[15] Fazzolari M, Alcala´ R, Herrera F (2014) A multi-objective evolutionary method for learning granularities based on fuzzy discretization to improve the accuracy-complexity trade-off of fuzzy rule-based classification systems: D-MOFARC algorithm. Appl Soft Comput 24:470–481

[16] Alikar N, Abdullah S, Mousavi SM, Niaki STA (2013) A hybrid particle swarm optimization and fuzzy rule-based system for breast cancer diagnosis. Int J Soft Comput 8(2):126–133.

[17] Hizal, S., ÇAVUŞOĞLU, Ü., & AKGÜN, D. (2021, June). A New Deep Learning Based Intrusion Detection System for Cloud Security. In 2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) (pp. 1-4). IEEE.

[18] M. Tavallaee, E. Bagheri, Wei Lu, and A. Ghorbani, A Detailed Analysis of the KDD CUP 99 data set", Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2009), pp. 1-6, 2009, pp. 1-6