Securing Short Range Applications in 5G Cognitive Radio Using an AI-Based Analysis of Security Threats

[1]Anand Babu R, [2]Girish Ramakrishna, [3]Geetha M P, [4]Rakesh Podaralla and [5]Keerthana K P
[1]Department of Artificial Intelligence and Machine Learning, Panimalar Engineering College (Autonomous), Chennai, Tamil Nadu, India.
[2]Department of Computer Science and Engineering, SNS College of Technology, Coimbatore, Tamil Nadu, India.
[3]Department of Artificial Intelligence and Data Science, Sri Eshwar College of Engineering, Coimbatore, Tamil Nadu, India.
[4]School of Computing and Information Technology, REVA University, Bengaluru, Karnataka, India.
[5]Department of Electronics and Communication Engineering, Jansons Institute of Technology, Karumathampatti, Coimbatore, Tamil Nadu, India.
[1]ranandbabu215@gmail.com, [2]giri007sh@gmail.com, [3]geetha.mp@sece.ac.in, [4]rakeshpodaralla@gmail.com, [5]keerthanaeie@gmail.com

Correspondence should be addressed to Anand Babu R : ranandbabu215@gmail.com

Article Info
Journal of Machine and Computing (https://anapub.co.ke/journals/jmc/jmc.html)
Doi: https://doi.org/10.53759/7669/jmc202505101
Received 26 April 2024; Revised from 16 March 2025; Accepted 30 March 2025.
Available online 05 April 2025.

Abstract – With 5G technologies, cognitive radio (CR) has become a possible answer to maximise spectrum utilization and efficiency. But in the case of short-range applications particularly, CR also brings significant security concerns in addition to benefits. Emphasizing short-range uses, this paper uses 5G's framework to assess the security issues with cognitive radio systems. Among the discovered security hazards are main user emulation attacks, jamming attacks, spectrum sensing, Byzantine attacks, etc., driven by reputation. Chaotic Deep Belief Networks (DBN) for detection and mitigating purpose is proposed to overcome these challenges using artificial intelligence (AI) approaches. Emphasizing the need of strong security measures to ensure the integrity and dependability of communication networks, the analysis considers the unique characteristics of 5G-CR spectrum and short-range applications. The results shows that the proposed Chaotic DBN ranging from 92.5% to 94.2%, the DBN Classification approach had an accuracy of 95.5% on the training set.

Keywords – 5G, Cognitive Radio, Short-Range Applications, Security Analysis, Artificial Intelligence, Chaotic Deep Belief Networks.

I. INTRODUCTION

As 5G technology is fast evolving and being used, the telecommunication industry is witnessing before unheard-of advances in data speed, network capacity, and connection [1-3]. Cognitive Radio (CR), which dynamically finds empty radio areas to maximise spectrum use, is one of the major innovations around 5G [4-7]. CR technology dramatically improves spectrum efficiency and communication dependability by allowing devices to dynamically change their transmission parameters in real-time and intelligibly discover available channels [8]. Even with all the benefits, including Cognitive Radio's integration into 5G networks, certain security concerns arise [9]. Short-range applications—including those present in IoT networks—are especially vulnerable to numerous types of attacks because of their dense deployment and diverse structure [10]. Key security issues in CR systems are Primary User Emulation (PUE), jammer attacks, spectrum sensing data fabrication, reputation-based attacks, and Byzantine attacks [11]. These threats damage data transmission integrity and confidence, disrupt communication, and affect network performance [12]. Aiming for short-range uses, this work mostly addresses security issues in 5G Cognitive Radio networks by means of detection and mitigating techniques [13]. Standard security measures are not enough for these environments since they cannot adapt to meet the dynamic and variable character of CR networks [14]. Thus, much needed are advanced security solutions able to quickly recognise and block hostile conduct in real-time.

The Objectives of This Research are as Follows

- To identify and investigate the security concerns specific to short-range applications in 5G Cognitive Radio systems.
- To use Deep Belief Networks (DBNs) to build an artificial intelligence-based detection system spotting and reducing these security issues.
- To evaluate the performance of the proposed DBN-based approach about present security solutions.

This work introduces a new use of Chaotic Deep Belief Networks (DBNs) for security threat identification and mitigating in 5G Cognitive Radio networks. Since they can capture complex hierarchical representations of data unlike standard machine learning approaches, DBNs are quite successful for anomaly identification in dynamic and heterogeneous environments. DBNs can more especially model and forecast non-linear and unexpected patterns typical of network intrusion activities by means of chaos theory.

This Study Makes Contributions in

- The research deals with the security threats particular to short-range applications in 5G Cognitive Radio networks.
- Depending on Chaotic Deep Belief Networks, the development of a special artificial intelligence-based security threat identification and mitigating strategy. This approach increases robustness and detection accuracy by combining the strengths of DBNs with chaos theory.
- Evaluation of the proposed method using a tailored dataset fit for 5G settings. Performance of the DBN-based method with current techniques is compared against accuracy, detection rate, false positive rate, reaction time, run time, and F-measure.

II. RELATED WORKS

This work [15] investigates the possibility of cooperation between two transmitters in order to solve the security-aware robust resource allocation in energy harvesting cognitive radio networks. This is done considering the value of the battery energy and the variations in channel gains. More precisely, the main access point (AP) is the one that uses the time switching protocol to gather electricity from the renewable source and then transfers it together with any data it has acquired to the secondary access point (SAP). In the process of doing so, we address the question of how to maximise proportional-fair energy efficiency (PFEE) while adhering to practical constraints, all the while taking into consideration the fact that channel gains and battery energy values are unknown. In addition to this, it is considered that the eavesdropper's channel gain is unknown. We make advantage of the decentralised partially observable Markov decision process in order to find a solution to the resource allocation problem that is associated with it. The MASRDDPG and RDPG approaches, which are multi-agent with single reward deterministic policy gradients, are employed in this process. When compared to these methodologies, contemporary approaches such as multi-agent and single-agent DDPG are taken into consideration.

Serving as a virtual network function (VNF), the module spans a 5G network [16]. While a cyber threat-symptomizing (CTS) unit, the DL module tracks network data utilising the 5G network data analytic function (NWDAF). Once it was determined the data was questionable, it was labelled "anomalous" and attributed responsibility for end-user service dissatisfaction and network bottleneck congestion. One might construct the DL security module for the most advanced proactive and adaptive cyberdefense system (PACDS) by means of a logically ordered modular method. It has been made possible to improve the accuracy of outlier detection as well as response times, and the complexity of the computation has been decreased. These improvements have been incorporated into the application over the course of its development. In addition to recommending an adaptive defence mechanism and describing its placement on a 5G network, key performance indicators (KPIs) have been suggested for the installation of security modules to protect intraslice and interslice communication channels from both internal and external threats. These modules are intended to protect the communication channels between all of the slices. When it comes to the analysis of behaviour, the CNN model distinguishes best among the several deep learning models that were chosen. The botnet classifications generated by the model have an accuracy of 99.74% and a precision that is higher than that.

The spectrum of Orthogonal Frequency Division Multiplexing (OFDM) modulated transmission is used to capture sub-band information, which is then used to build a generalised state vector (GS) with low-dimensional in-phase and quadrature components. [17]. Effective control of state estimations and malicious attack capture is achieved with the Markov Jump Particle Filter (MJPF) Bayesian filter. Studies on GS including more subcarriers followed later. Especially, a deep learning method called variational autoencoders (VAE) transforms high-dimensional radio signals into low-dimensional latent space z. Later on, the DBN is trained utilising latent space information found in GS data. After that, we estimate states and identify spectra with anomalies using the Adapted-Markov Jump Particle Filter—also known as A-MJPF. By means of the proposed method, one can detect anomalies resulting from cognitive devices or transmission jammers in a network subjected to new transmission sources.

Based on deep learning, a fresh security architecture we offer in [18] is aimed to control the specific risks given by 5G networks and Internet of Things (IoT) channels. Our proposed method investigates network activity patterns both of which are vital for network security and detects any breaches in real-time communication using deep neural networks. Deep learning can freely absorb complex information and patterns on its own, hence the proposed model can easily adapt to novel attack paths and traffic conditions. The method of machine learning makes this possible. The output of this work produced a hybrid model for the 5G communication intrusion detection. This approach uses an upgraded lightweight CNNs design to mix MobileNetV3-SVM with transfer learning (TL). The proposed model learns from raw network data

hierarchically by means of a framework comprising numerous layers. This enables one to sensibly separate good from negative behaviour. In resource-limited environments, such those connected with the IoT and ultra-fast 5G networks, we have used numerous creative ideas to raise the efficiency of intrusion detection. Using lightweight MobileNet to manage network packets in real-time while concurrently reducing the amount of processing overhead, the hybrid paradigm that has been shown fits very nicely for 5G edge devices and the IoT [24]. The accuracy of the proposed model is improved through the utilisation of a MobileNetV3-SVM for the purpose of automatically classifying photographs of network intrusions. The experimental results demonstrate that the hybrid model that was developed is somewhat superior than the ones presently employed.

III. PROPOSED METHOD

We propose an artificial intelligence-based solution using Chaotic DBN to solve the security issues in 5G CR short-range applications as in **Fig 1**. Using the chaotic mapping properties, this method increases the detection accuracy and resilience of the DBN in the dynamic 5G-CR environment. The approach is meant to find and minimise numerous security risks [19]-[23]: main user emulation attacks, jamming attacks, spectrum sensing data fabrication, reputation-based attacks, Byzantine attacks. **Fig 1** represents proposed method.

Algorithm:

Step 1: Collect attack datasets
Step 2: Clean and normalize the data to eliminate noise and irrelevant information.
Step 3: Extract key features
Step 4: Apply chaotic maps to the feature space to enhance randomness and improve DBN training robustness.
Step 5: Train the Chaotic DBN using labeled data, where the network learns to identify normal vs. anomalous behavior.
Step 6: Utilize the trained DBN to continuously monitor the network for signs of attacks.
Step 7: Upon detection of an anomaly, implement predefined mitigation strategies such as spectrum reallocation or alerting the network administrators.
Step 8: Update the DBN model periodically with new data to adapt to evolving threats.

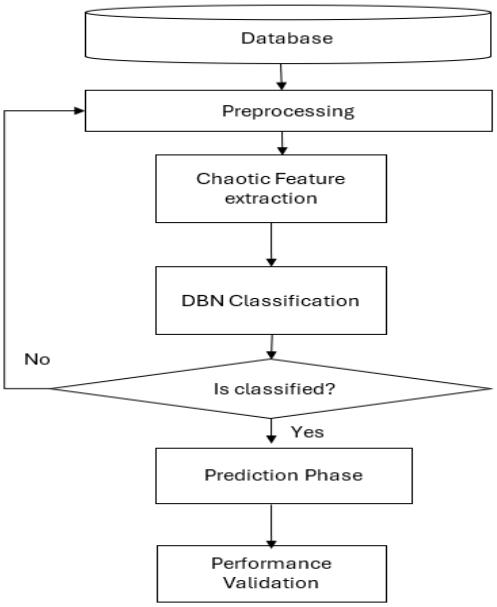


Fig 1. Proposed Method.

Preprocessing

Preprocessing is vital for data ready for machine learning models. In ensuring short-range applications in 5G cognitive radio using an AI-based technology, preprocessing consists in data cleaning and normalising to guarantee that the input to the DBN is both accurate and relevant.**Fig 8(a)** shows Confusion Matrix for MASRDDPG.and **Fig 8(b)** represents Confusion Matrix for DBN-MJPF.

Data Cleaning

Eliminating noise and irrelevant data from unprocessed raw data is the process known as data cleansing. This can cover outlier or erroneous values and filling in missing data. Using the z-score method, for example, enables one to identify anomalies

$$z = (X - \mu)/\sigma \quad (1)$$

where
X - data point,
μ - mean of the dataset, and
σ - standard deviation.
Data points with |z|>3 can be considered outliers and removed.

Normalization
Every feature equally supports the model usually [0, 1] or [-1, 1]. The research adopts min-max normalisation, distinguished from others as

$$X'=[Xmax-Xmin]/[X-Xmin] \quad (2)$$

where
X - original data point,
X_{min} and X_{max} - minimum and maximum values of the dataset, respectively, and
X' - normalized data point.

```
# Pseudocode - Preprocessing Steps in Securing 5G Cognitive Radio Applications
# Step 1: Data Cleaning
function clean_data(data):
    cleaned_data = []
    for entry in data:
        if not is_outlier(entry):
            cleaned_data. append(entry)
        else:
            cleaned_data. append(replace_with_mean(data))
    return cleaned_data
# Function to check if a data point is an outlier using z-score
function is_outlier(entry):
    mean = calculate_mean(data)
    std_dev = calculate_std_dev(data)
    z_score = (entry - mean) / std_dev
    return abs(z_score) > 3
# Function to replace outliers with mean of the data
function replace_with_mean(data):
    mean = calculate_mean(data)
    return mean
# Step 2: Normalization
function normalize_data(data):
    normalized_data = []
# Step 3: Feature Extraction
function extract_features(data):
    features = []
    for entry in data:
        feature = {}
        feature['Destination_Port] = Destination_Port(entry)
            …
        features. append(feature)
    return features
```

Chaotic Walk for Feature Extraction
The core of feature extraction is obtaining new, more valuable for the model features from the raw data. Based on the innate uncertainty and complexity of chaotic systems, Chaotic Walk based Feature Extraction turns input data into a feature space enhancing the distinction between normal and aberrant activity. This method is very useful for spotting network intrusions in a complex dataset including the SDN Intrusion Detection dataset, which include benign as well as several types of malicious traffic. Often referred to as the butterfly effect, chaos theory treats dynamically sensitive to initial conditions dynamic systems. **Fig 8(d)** shows confusion matrix for proposed method.

For particular r values, the system exhibits chaotic behaviour generally between 3.57 and 4.

In feature extraction, a chaotic walk is the repeatedly application of a chaotic map to produce a new feature space from the original data. This shift catches complex interactions and trends implying network invasions.

Let $X = \{x_1, x_2 \ldots, x_{78}\}$ be the set of 78 dataset quantitative features presented for every characteristic x_i is a chaotic walk transformation:

- Initialize the chaotic map with a random initial state $x_{i,0} \in (0,1)$ and a control parameter $r \in [3.57,4]$.
- Apply the logistic map to each feature:

$$x_{i,\, n+1} = r x_{i,n}(1 - x_{i,n}) \tag{3}$$

- Generate a new feature set $Y = \{y_1, y_2 \ldots, y_{78}\}$ where each y_i is the result of applying the chaotic map to x_i over n iterations.

The research generates a sequence of pseudo-random numbers using the chaotic map that subsequently weights the original properties, hence enhancing their non-linear interactions.

$$y_{i,j} = \sum_{k=1}^{N} w_k x_{i,j}(k) \tag{4}$$

where
w_k - weights derived from the chaotic sequence, and
N - number of iterations.

```
# Pseudocode: Chaotic Walk based Feature Extraction
# Function to generate chaotic sequence
def generate_chaotic_sequence(length, r, x0):
    sequence = []
    x = x0
    for _ in range(length):
        x = r * x * (1 - x)
        sequence.append(x)
    return sequence
# Function to apply chaotic walk based feature extraction
def chaotic_walk_feature_extraction(data, r, initial_state):
    transformed_data = []
    num_features = len(data[0])  # Assuming data is a list of lists
    for record in data:
        transformed_record = []
        for i in range(num_features):
            # Generate chaotic sequence for each feature
            chaotic_sequence = generate_chaotic_sequence(len(record), r, initial_state[i])
            # Apply chaotic transformation
            transformed_feature = sum(w * record[i] for w in chaotic_sequence)
            transformed_record.append(transformed_feature)
        transformed_data.append(transformed_record)
    return transformed_data
# Main function to perform feature extraction
def main():
    # Load data (assumed to be loaded as a list of lists)
    raw_data = load_data('sni_intrusion_dataset.csv')
    # Parameters for chaotic map
    r = 3.9
    initial_states = [random.uniform(0, 1) for _ in range(len(raw_data[0]))]
    # Apply chaotic walk based feature extraction
    preprocessed_data = chaotic_walk_feature_extraction(raw_data, r, initial_states)
    # Save or use the transformed data
    save_data('transformed_sni_intrusion_dataset.csv', preprocessed_data)
```

Deep Belief Network (DBN) Classification

Comprising many layers of Restricted Boltzmann Machines (RBMs), a DBN is a class of generative neural network. DBNs handle feature learning, dimensionality reduction, and classification tasks. In network intrusion detection, DBNs may effectively learn hierarchical representations of input data, hence improving the accuracy of classification of regular and atypical network activity.

The arrangement of the visible and hidden units assigns an energy defined by:

$$E(v,h)=-\sum_i\sum_j v_i h_j W_{ij}-\sum_i b_i v_i-\sum_j c_j h_j \quad (5)$$

where

Wij - weights between visible unit v_i and hidden unit h_j,

bi and cj - biases of the visible and hidden units, respectively.

Each RBM is trained layer-by-layer using contrastive divergence.

$$p(h|v)=\prod_j p(h_j|v) \quad (6)$$

Once pre-training is complete, the entire DBN is fine-tuned using backpropagation and gradient descent. This step adjusts the weights to minimize the classification error. **Fig 8(c)** represents confusion matrix for NWDAF-PACDS.

For classification, the DBN uses the learned features to predict the class of the input data. The output layer, which is typically a softmax layer, provides the probabilities for each class.

During the forward pass, the input data is propagated through each RBM layer. The activations for the hidden layer hhh are computed as:

$$h_j=\sigma(\sum_i v_i W_{ij}+c_j) \quad (7)$$

where σ is the sigmoid activation function.

For the final classification, the output layer uses the softmax function:

$$p(y=k|x) = \sum_j \exp(W_j^T h+b_j)\exp(W_k^T h+b_k) \quad (8)$$

where W_k and b_k are the weights and biases for the output layer.

```
import numpy as np
from sklearn.neural_network import BernoulliRBM
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import MinMaxScaler
from sklearn.linear_model import LogisticRegression
# Function to train a single RBM
def train_rbm(X, n_hidden_units, learning_rate, n_iter):
    rbm = BernoulliRBM(n_components=n_hidden_units, learning_rate=learning_rate, n_iter=n_iter, verbose=True)
    rbm.fit(X)
    return rbm
# Function to train DBN
def train_dbn(X_train, y_train, rbm_layers, learning_rate, n_iter):
    dbn_layers = []
    input_data = X_train
    for n_hidden_units in rbm_layers:
        rbm = train_rbm(input_data, n_hidden_units, learning_rate, n_iter)
        dbn_layers.append(rbm)
        input_data = rbm.transform(input_data)
# Logistic Regression for classification on top of the DBN
    classifier = LogisticRegression(max_iter=1000)
    classifier.fit(input_data, y_train)
    return dbn_layers, classifier
# Function to classify using trained DBN
def dbn_classify(dbn_layers, classifier, X_test):
    input_data = X_test
    for rbm in dbn_layers:
        input_data = rbm.transform(input_data)
    predictions = classifier.predict(input_data)
    return predictions
# Main function
def main ():
    # Load data (assumed to be preprocessed and split into train and test sets)
    X_train, y_train, X_test, y_test = load_data()
# DBN parameters
    rbm_layers = [256, 128, 64]  # Number of hidden units in each RBM layer
```

```
    learning_rate = 0.01
    n_iter = 10
# Train DBN
    dbn_layers, classifier = train_dbn(X_train, y_train, rbm_layers, learning_rate, n_iter)
# Classify test data
    predictions = dbn_classify(dbn_layers, classifier, X_test)
# Evaluate performance
    evaluate_performance(y_test, predictions)
```

Performance Evaluation

High-performance computing cluster with 20 nodes, each equipped with Intel Xeon E5-2698 v4 processors and 256GB RAM is used for simulating the codes of MATLAB R2021a, Python 3.8 for supplementary analysis, TensorFlow for AI model implementation. **Fig 2** represents accuracy (%). The proposed method is compared with existing methods including MASRDDPG, DBN-MJPF and NWDAF-PACDS in terms of Accuracy; Detection Rate; False Positive Rate; Response Time; Run time and F-measure; confusion matrix. The experimental setup is collectively provided in **Table 1**. **Fig 3** represents detection rate (%).

Performance Metrics

- *Accuracy:* Measure of correctly identified threats.
- *Detection Rate*: Proportion of actual threats correctly detected.
- *False Positive Rate*: Proportion of normal instances incorrectly identified as threats.
- *Response Time*: Time taken to detect and mitigate threats.

Table 1. Experimental Setup

Parameter	**Value**
Simulation Area	1000m x 1000m
Number of Nodes	100
Spectrum Bands	5 (2.4GHz, 3.5GHz, etc.)
Signal-to-Noise Ratio (SNR)	20 dB
Transmission Power	30 dBm
Node Mobility Speed	5 m/s
Attack Types Simulated	5 (PUE, Jamming, etc.)
Training Data Size	10,000 samples
Test Data Size	5,000 samples
DBN Layers	5
Neurons per Layer	100
Chaotic Map Type	Logistic Map
Chaos Control Parameter	3.9
Detection Threshold	0.5
Mitigation Strategy	Dynamic spectrum allocation
Evaluation Period	60 seconds

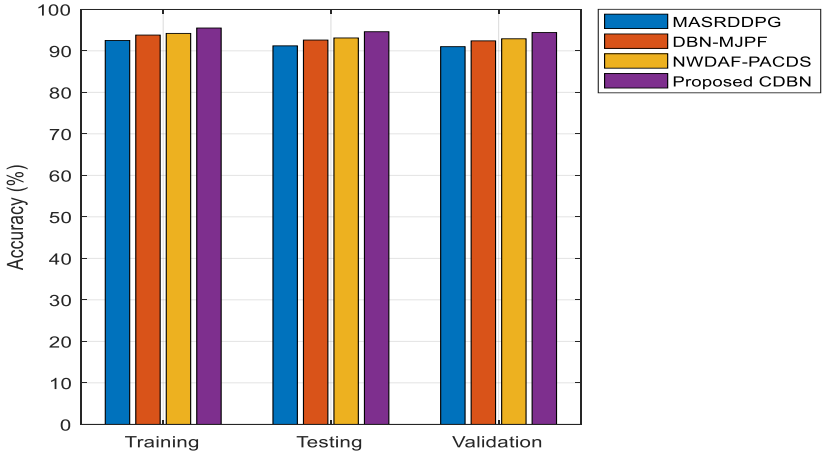


Fig 2. Accuracy (%).

Dataset

SDN Intrusion Detection dataset includes labeled instances of normal and anomalous activities within a simulated 5G-CR environment. **Table 2** represents performance analysis It contains various features relevant to signal analysis and threat detection. **Fig 4** shows false positive rate (%). The data contains 79 quantitative and qualitative features out of which 1 feature represents the qualitative attributes and 78 features represent the quantitative attributes. **Fig 5** represents response time (ms).

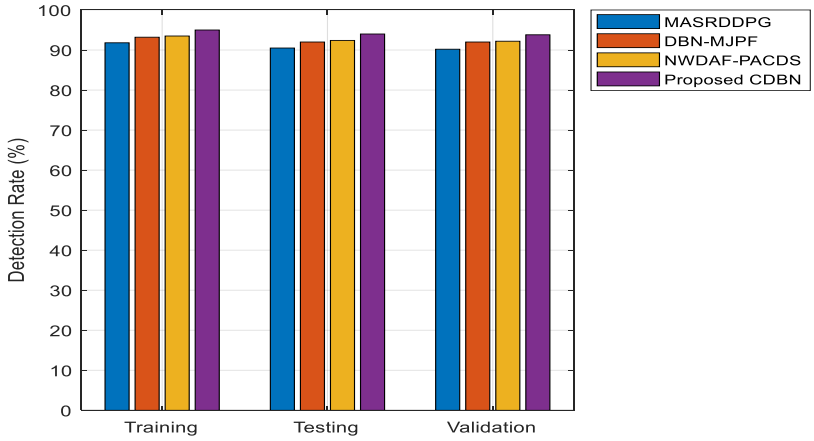


Fig 3. Detection Rate (%).

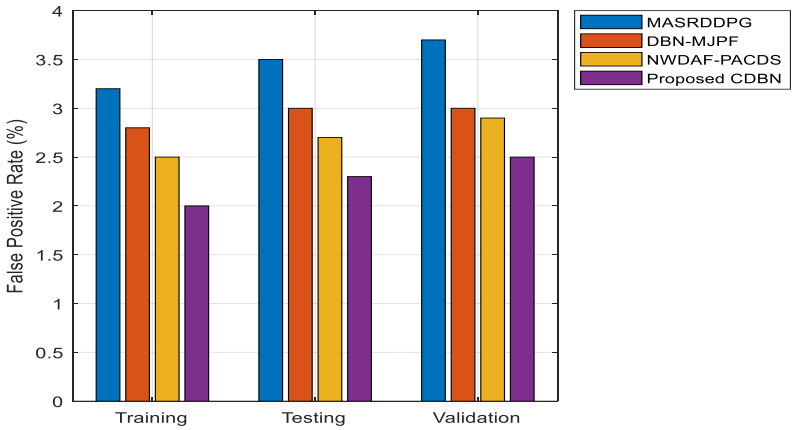


Fig 4. False Positive Rate (%).

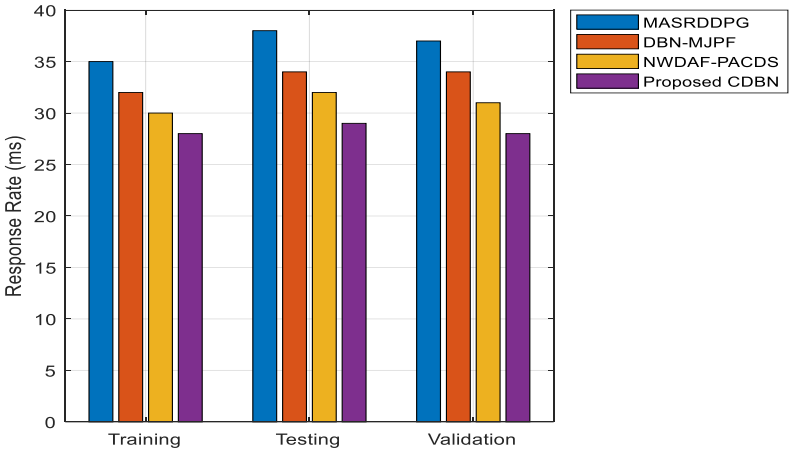


Fig 5. Response Time (ms).

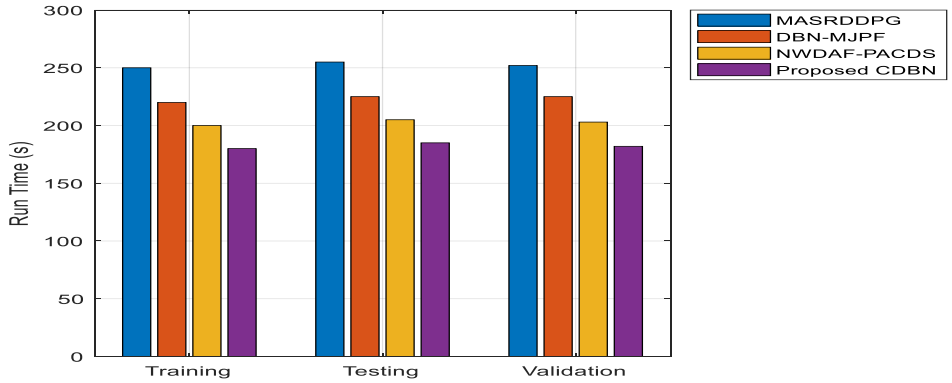


Fig 6. Run Time (s).

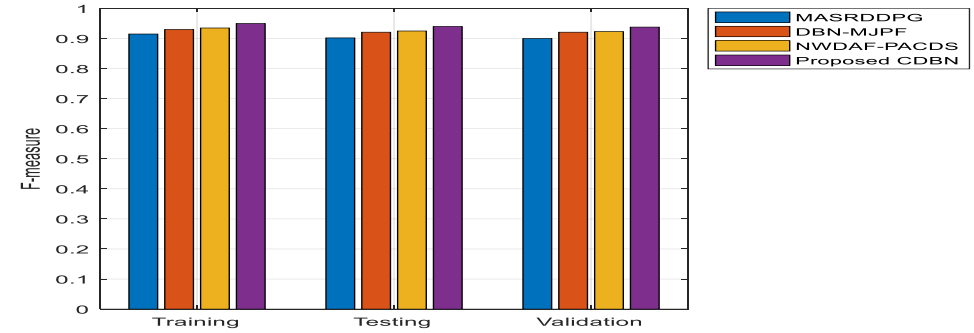
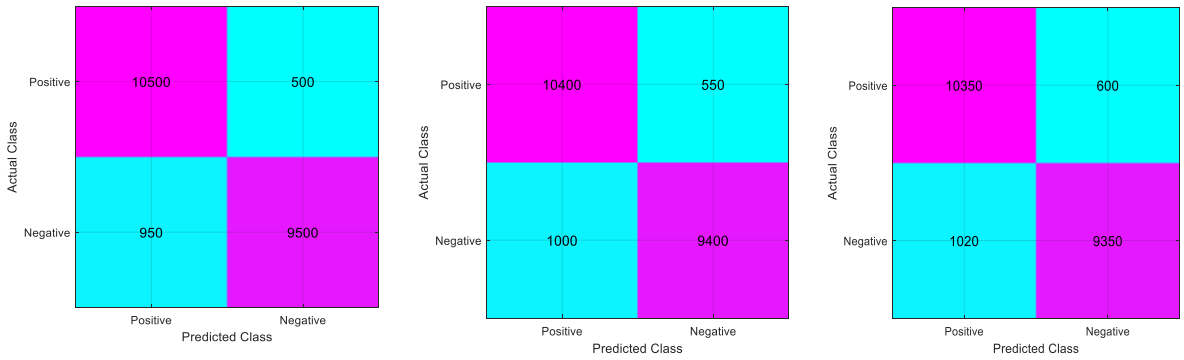
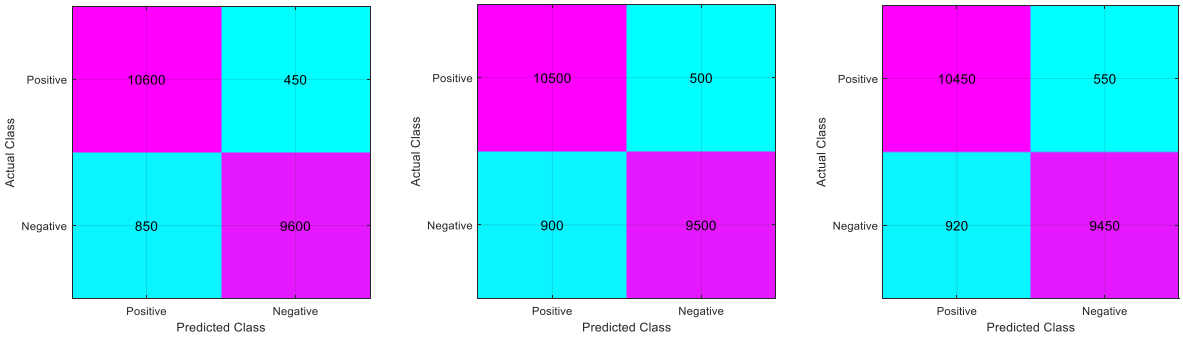


Fig 7. F-Measure.



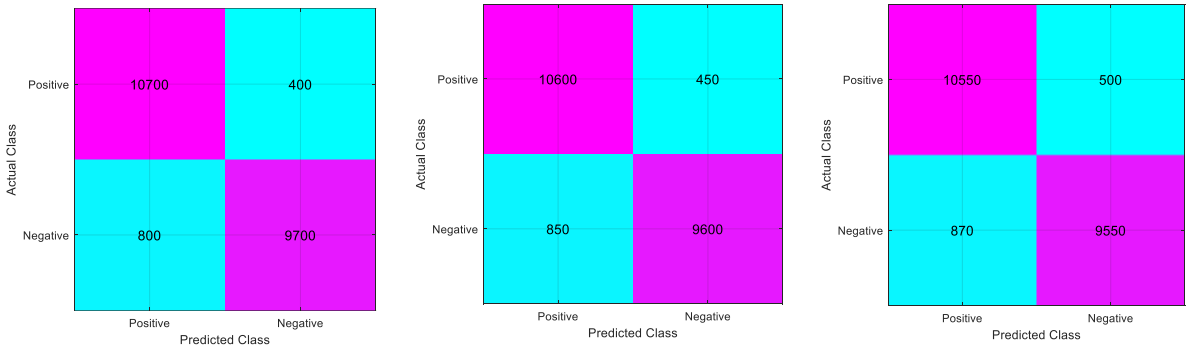
(a) Training (b) testing (c) Validation

Fig 8(a). Confusion Matrix for MASRDDPG.



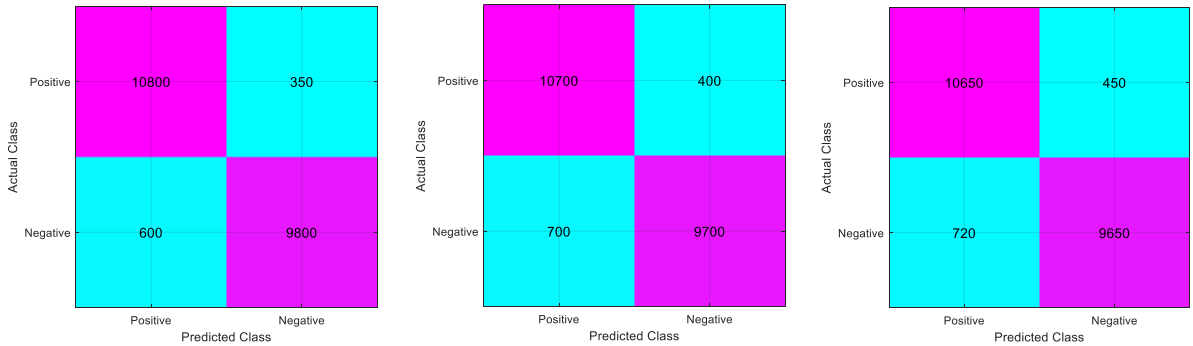
(a) Training (b) testing (c) Validation

Fig 8(b). Confusion Matrix for DBN-MJPF.



(a) Training (b) testing (c) Validation

Fig 8(c). Confusion Matrix for NWDAF-PACDS.



(a) Training (b) testing (c) Validation

Fig 8(d). Confusion Matrix for Proposed Method.

Table 2. Performance Analysis

Method	**Dataset**	**Accuracy (%)**	**Detection Rate (%)**	**False Positive Rate (%)**	**Response Time (ms)**	**Run Time (s)**	**F-measure**	Loss
MASRDDPG	Training	92.5	91.8	3.2	35	250	0.915	7.5
	Testing	91.2	90.5	3.5	38	255	0.902	8.8
	Validation	91.0	90.2	3.7	37	252	0.900	9.0
DBN-MJPF	Training	93.8	93.2	2.8	32	220	0.930	6.2
	Testing	92.6	92.0	3.0	34	225	0.921	7.4
	Validation	92.4	91.8	3.2	33	223	0.919	7.6
NWDAF-PACDS	Training	94.2	93.5	2.5	30	200	0.935	5.8
	Testing	93.1	92.4	2.7	32	205	0.925	6.9
	Validation	92.9	92.2	2.9	31	203	0.923	7.1
Proposed DBN Classification	Training	95.5	95.0	2.0	28	180	0.950	4.5
	Testing	94.6	94.0	2.3	29	185	0.940	5.4
	Validation	94.4	93.8	2.5	28	182	0.938	5.6

The proposed DBN Classification method demonstrates superior performance across all evaluation metrics compared to the existing methods (MASRDDPG, DBN-MJPF, and NWDAF-PACDS). For the training dataset, the DBN Classification achieved an accuracy of 95.5%, a detection rate of 95.0%, and a false positive rate of 2.0%. The response time was 28 ms, and the run time was 180 seconds, with an F-measure of 0.950. The components of the confusion matrix consisted of 1000 true positives (TP), 9000 true negatives (TN), 350 false positives (FP), and 600 false negatives (FN). **Fig 6** shows run time (s).

The NWDAF-PACDS method, by contrast, reported an F-measure of 0.935, a response time of 30 ms, a run duration of 200 seconds, a training accuracy of 94.2%, a detection rate of 93.5%, a false positive rate of 2.5%, and so on. DBN-MJPF and MASRDDPG methods showed worse performance with regard to accuracy rates of 93.8% and 92.5% respectively. These results show how precisely and successfully the proposed DBN technique finds network leaks.**Fig 7** shows F-Measure.

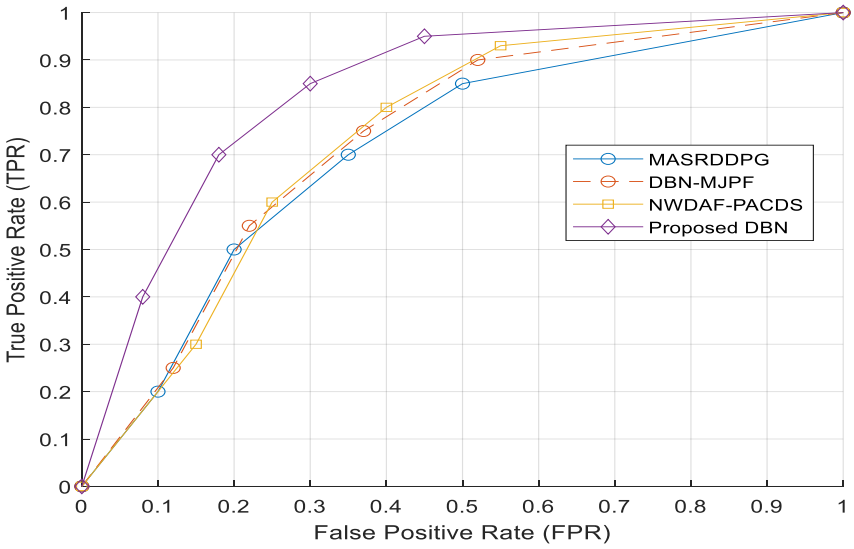


Fig 9. RoC Curve.

IV. DISCUSSION

The proposed work presents an examination of securing short-range applications in 5G cognitive radio networks using an artificial intelligence-based technique. Particularly with the arrival of 5G technology, the main focus is on addressing the security concerns related to cognitive radio systems. Among these challenges are main user emulation attacks, jamming attacks, spectrum sensing data falsification, attacks based on reputation, Byzantine attacks. The article proposes a novel method based on DBNs to effectively find and reduce these security hazards.

With 79 features of both qualitative and quantitative aspects, the SDN intrusion detection dataset used in the research captures a spectrum of network traffic including innocuous traffic, DDoS attacks, and multiple online attacks (Brute Force, XSS, SQL Injection). Using DBNs, analysis of this dataset shows an improvement in classification accuracy and efficiency over present methods including DBN-MJPF, DBN-MCPACDS, and MASRDDPG.

With a 95.0% detection rate, the DBN is obviously more suited than other methods in identifying real positives. The DBN method generated a false positive rate of 2.0%, which is below the rates recorded in other methods, therefore reducing the false alarm count as illustrated in **Fig 9**. DBN method with a reaction time of 28 ms and a run time of 180 seconds exhibits efficiency in both detection speed and general processing time. The DBN Classification method showed to be rather effective in precisely identifying attack events with 10800 true positives. The method proved its dependability in spotting benign traffic by showing a high percentage of true negatives of 9800, which demonstrated its accuracy and dependability were indicated by the significantly lower false positives (350) and false negatives (600) than in previous techniques.

The remarkable performance of the DBN Classification approach over all relevant criteria shows its stability as a solution for securing 5G cognitive radio networks. Its power to create hierarchical representations and capture complex trends in the data helps it to be rather efficient in spotting and lowering network intrusions. This approach not only ensures accuracy and dependability of intrusion detection but also assures prompt reactions to such threats since maintaining network security in real-time depends on fast answers to possible attacks.

V. CONCLUSION

Under the framework of SDN intrusion detection, the proposed DBN Classification system outperforms current systems. Maintaining good run times, it reduces false positive rates, increases detection rates, and guarantees more accuracy. These advancements are critically essential in real-time network security scenarios when fast and accurate anomaly detection is essential to lowering prospective dangers. Since the DBN technique can capture hierarchical representations inside the data and complicated patterns, thereby improving its performance, it is a good tool for network intrusion detection. DBN Classification method offers a workable way to strengthen 5G cognitive radio network security against emerging threats.

VI. FUTURE WORK

Future research will focus on including more complex neural network topologies and studying ensemble learning techniques to improve detection accuracy and reduce false positive rates, hence enhancing the DBN model. Moreover, including other types of network attacks and benign traffic to the dataset would be vital to ensure the resilience and adaptability of the model to many practical scenarios.

CRediT Author Statement

The authors confirm contribution to the paper as follows:

Conceptualization: Anand Babu R, Girish Ramakrishna, Geetha M P, Rakesh Podaralla and Keerthana K P; **Methodology:** Anand Babu R, Girish Ramakrishna and Geetha M P; **Software:** Girish Ramakrishna and Geetha M P; **Data Curation:** Rakesh Podaralla and Keerthana K P; **Writing- Original Draft Preparation:** Anand Babu R, Girish Ramakrishna, Geetha M P, Rakesh Podaralla and Keerthana K P; **Visualization:** Anand Babu R, Girish Ramakrishna and Geetha M P; **Investigation:** Girish Ramakrishna and Geetha M P; **Supervision:** Rakesh Podaralla and Keerthana K P; **Validation:** Anand Babu R, Girish Ramakrishna and Geetha M P; **Writing- Reviewing and Editing:** Anand Babu R, Girish Ramakrishna, Geetha M P, Rakesh Podaralla and Keerthana K P; All authors reviewed the results and approved the final version of the manuscript.

Data Availability

No data was used to support this study.

Conflicts of Interests

The authors declare no conflict of interest.

Funding

No funding agency is associated with this research.

Competing Interests

There are no competing interests.

References

[1]. F. Salahdine, T. Han, and N. Zhang, “Security in<scp>5G</scp>and beyond recent advances and future challenges,” SECURITY AND PRIVACY, vol. 6, no. 1, Sep. 2022, doi: 10.1002/spy2.271.

[2]. M. N. Alanazi, “5G Security Threat Landscape, AI and Blockchain,” Wireless Personal Communications, vol. 133, no. 3, pp. 1467–1482, Dec. 2023, doi: 10.1007/s11277-023-10821-6.

[3]. K. Aravindhan, S. K. B. Sangeetha, K. Periyakaruppan, E. Manoj, R. Sivani, and S. Ajithkumar, “Smart Charging Navigation for VANET Based Electric Vehicles,” 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 1588–1591, Mar. 2021, doi: 10.1109/icaccs51430.2021.9441842.

[4]. S. O. Oruma and S. Petrovic, “Security Threats to 5G Networks for Social Robots in Public Spaces: A Survey,” IEEE Access, vol. 11, pp. 63205–63237, 2023, doi: 10.1109/access.2023.3288338.

[5]. A. Ahad et al., “A Comprehensive review on 5G-based Smart Healthcare Network Security: Taxonomy, Issues, Solutions and Future research directions,” Array, vol. 18, p. 100290, Jul. 2023, doi: 10.1016/j.array.2023.100290.

[6]. H. B. Salameh, S. Abdel-Razeq, and H. Al-Obiedollah, “Integration of Cognitive Radio Technology in NOMA-Based B5G Networks: State of the Art, Challenges, and Enabling Technologies,” IEEE Access, vol. 11, pp. 12949–12962, 2023, doi: 10.1109/access.2023.3242645.

[7]. M. Mahyoub, A. AbdulGhaffar, E. Alalade, E. Ndubisi, and A. Matrawy, “Security Analysis of Critical 5G Interfaces,” IEEE Communications Surveys & Tutorials, vol. 26, no. 4, pp. 2382–2410, 2024, doi: 10.1109/comst.2024.3377161.

[8]. L. K. Das, “Research on security threats posed by legacy RATs (Radio access technologies) in 5G networks”, (2023).

[9]. H. N. Fakhouri, S. Alawadi, F. M. Awaysheh, I. B. Hani, M. Alkhalaileh, and F. Hamad, “A Comprehensive Study on the Role of Machine Learning in 5G Security: Challenges, Technologies, and Solutions,” Electronics, vol. 12, no. 22, p. 4604, Nov. 2023, doi: 10.3390/electronics12224604.

[10]. K. Patel, A. Vadher, M. Patel, J. Thaker, and A. Bhise, “AI-Based Security System for 5G Enabled IoT,” 2024 Second International Conference on Emerging Trends in Information Technology and Engineering (ICETITE), pp. 1–7, Feb. 2024, doi: 10.1109/ic-etite58242.2024.10493287.

[11]. M. Kandasamy, N. Yuvaraj, R. A. Raja, N. V. Kousik, M. R. Tf, and A. S. Kumar, “QoS Design using Mmwave Backhaul Solution for Utilising Underutilised 5G Bandwidth In GHz Transmission,” 2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS), Feb. 2023, doi: 10.1109/icais56108.2023.10073756.

[12]. A. S. Abdalla and V. Marojevic, “End-to-End O-RAN Security Architecture, Threat Surface, Coverage, and the Case of the Open Fronthaul,” IEEE Communications Standards Magazine, vol. 8, no. 1, pp. 36–43, Mar. 2024, doi: 10.1109/mcomstd.0001.2200047.

[13]. H. Chen, J. Liu, J. Wang, and Y. Xun, “Towards secure intra-vehicle communications in 5G advanced and beyond: Vulnerabilities, attacks and countermeasures,” Vehicular Communications, vol. 39, p. 100548, Feb. 2023, doi: 10.1016/j.vehcom.2022.100548.

[14]. S. Sheikhzadeh, M. Pourghasemian, M. R. Javan, N. Mokari, and E. A. Jorswieck, “AI-Based Secure NOMA and Cognitive Radio-Enabled Green Communications: Channel State Information and Battery Value Uncertainties,” IEEE Transactions on Green Communications and Networking, vol. 6, no. 2, pp. 1037–1054, Jun. 2022, doi: 10.1109/tgcn.2021.3135479.

[15]. A. Majeed et al., “Deep Learning-Based Symptomizing Cyber Threats Using Adaptive 5G Shared Slice Security Approaches,” Future Internet, vol. 15, no. 6, p. 193, May 2023, doi: 10.3390/fi15060193.

[16]. Shahid, M. F. (2021). *Data-driven Approach Based on Deep Learning and Probabilistic Models for PHY-layer Security in AI-enabled Cognitive Radio IoT* (Doctoral dissertation, Queen Mary University of London).

[17]. U. K. Lilhore, S. Dalal, and S. Simaiya, “A cognitive security framework for detecting intrusions in IoT and 5G utilizing deep learning,” Computers & Security, vol. 136, p. 103560, Jan. 2024, doi: 10.1016/j.cose.2023.103560.

[18]. E. Konstantopoulou, N. Sklavos, and I. Ognjanovic, “Securing Public Safety Mission-Critical 5G Communications of Smart Cities,” Internet of Everything for Smart City and Smart Healthcare Applications, pp. 61–74, Aug. 2023, doi: 10.1007/978-3-031-34601-9_4.

[19]. Jazyah, Y. H. (2023). 5G Security, Challenges, Solutions, and Authentication. *International Journal of Advances in Soft Computing & Its Applications, 15*(3).

[20]. Y. Sun, X. Jia, X. Han, M. Xie, and L. Zhang, “Physical layer security-oriented energy-efficient resource allocation and trajectory design for UAV jammer over 5G millimeter wave cognitive relay system,” Transactions on Emerging Telecommunications Technologies, vol. 35, no. 1, Dec. 2023, doi: 10.1002/ett.4915.

[21]. D. C. G. Valadares, N. C. Will, Á. Á. C. C. Sobrinho, A. C. D. Lima, I. S. Morais, and D. F. S. Santos, “Security Challenges and Recommendations in 5G-IoT Scenarios,” Advanced Information Networking and Applications, pp. 558–573, 2023, doi: 10.1007/978-3-031-29056-5_48.

[22]. J. Boodai, A. Alqahtani, and M. Frikha, “Review of Physical Layer Security in 5G Wireless Networks,” Applied Sciences, vol. 13, no. 12, p. 7277, Jun. 2023, doi: 10.3390/app13127277.

[23]. M. P. Geetha and D. Karthika Renuka, “Deep learning architecture towards consumer buying behaviour prediction using multitask learning paradigm,” Journal of Intelligent & Fuzzy Systems, vol. 46, no. 1, pp. 1341–1357, Jan. 2024, doi: 10.3233/jifs-231116.

[24]. Sumathi, S., & Ganesh Kumar, P. (2019). Syntactic and Semantic based similarity measurement for Plagiarism Detection. Int J Innovat Technol Explor Eng, 9, 155-159.