REPUBLIC OF TURKEY

ALTINBAS UNIVERSITY

Institute of Graduate Studies

Information Technology

COMPUTER NETWORK TRAFIC CLASSIFICATION BASED AI TECHNIQUES

Mohammed Akram ALI

Master's Thesis

Supervisor

Asst. Prof. Dr. Ayça KURNAZ TÜRKBEN

Istanbul, 2022

COMPUTER NETWORK TRAFIC CLASSIFICATION BASED AI TECHNIQUES

Mohammed Akram ALI

Information Technology

Master's Thesis

ALTINBAŞ UNIVERSITY

2022

The thesis titled “COMPUTER NETWORK TRAFIC CLASSIFICATION BASED AI TECHNIQUES” prepared by “MOHAMMED AKRAM ALI ALI” and submitted on 20/05/2022 has been **accepted unanimously** for the degree of Master of Science in Information Technologies.

Asst. Prof. Dr. Ayça Kurnaz Türkben

Supervisor

Thesis Defense Jury Members:

Asst. Prof. Dr. Ayça Kurnaz Türkben	Faculty of Engineering and Natural Sciences, Altinbas University	_____
Asst. Prof. Dr Abdullah Abdu IBRAHIM	Faculty of Engineering and Natural Sciences, Altinbas University	_____
Asst. Prof. Dr. Serdar KARGIN	Faculty of Engineering and Natural Sciences, Beykent University	_____

I hereby declare that this thesis/dissertation meets all format and submission requirements of a Master’s thesis.

Submission date of the thesis to Institute of Graduate Studiees: ___/___/___

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Mohammed Akram ALI

DEDICATION

First and foremost, I want to express my gratitude to Allah Almighty for providing me with the mental acuity, health, and stamina necessary to finish this course of study.

My parents are the true inspirations behind my dissertation. It's impossible for me to put into words how much you mean to me; I can never repay you. Please know that I'll do all in my power to meet your expectations.

Finally, I'd want to thank everyone who has supported and encouraged me during this research process, especially my family and close friends.

ABSTRACT

COMPUTER NETWORK TRAFIC CLASSIFICATION BASED AI TECHNIQUES

ALI, Mohammed

M.Sc, Information Technology, Altınbaş University,

Supervisor:Asst Prof. Dr. Ayça Kurnaz Türkben

Date: 20/05/ 2022

Pages: 60

Identifying and categorizing network traffic is the initial step in determining what sorts of applications are traveling via the network. An internet service provider or network operator may better control a network's overall performance by using this method. For the classification of network traffic data, new AI methods based on methods are provided in this thesis.

The data obtained from Kaggle was used to train the KNN to classify the dataset into normal and abnormal. The aim of KNN is to learn the features of the normal and abnormal cases of traffic and try to classify them. Then a search algorithm is applied to obtain the best parameters for the KNN. The grid search algorithm was applied to optimize the performance of the KNN and presented remarkable results when compared with previous research.

Keywords: K-Nearest Neighbor, Grid Search, Artificial Intelligence, Network Traffic.

TABLE OF CONTENT

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

ABBREVIATIONS

RNN	:	Recurrent Neural Network
KNN	:	K-Nearest Neighbor algorithm
DL	:	Deep Learning
AI	:	Artificial Intelligence
ANN	:	Artificial Neural Network
SVM	:	Support Vector Machines
AIS	:	Artificial Immune System
IDS	:	Intrusion Detection Systems
TCP	:	Transmission Control Protocol
QOS	:	Quality Of Service

1. INTRODUCTION

Today's world is very rich in terms of "Information" and computers in every field have become indispensable tools for processing, transmitting, and storing information. These capabilities of computers are supported by computer networks that allow them to exchange data between them. A computer network is a system in which multiple computers are interconnected to share software and hardware resources, including information. The concept of computer networking can be explained by the combination of more than one layer, from the physical connections of computers to the communication channels of the applications running on them. Each layer has a specific task and although they are independent of all other layers, they work as a whole. Incoming and outgoing data between the computer network and the computer are transmitted through each layer, respectively. More than one model has been introduced and applied to the informatics world to create a layered communication architecture. Among the important ones are the 7-layer OSI [1] model used for general communication and the 4-layer Internet Model, which is customized for internet communication. The Internet Model consists of Connection, Internet, Transport, and Application (Link, Internet, Transport, and Application) layers, respectively.

With the widespread use of the Internet, the number of cyberattacks is increasing rapidly. The importance of data protection when using the Internet is closely related to the importance of information security. Personal, business, and military information is exchanged between individuals or institutions over the Internet. Cryptographic techniques and protocols are used to keep this shared information confidential. The use of Secure Hypertext Transfer Protocol (HTTPS) traffic on the Internet exceeds 95% and continues to grow. Without encryption, cybercriminals can capture sensitive data and use it for malicious purposes [2].

It is popular to utilize Virtual Private Networks (VPNs) and SSL to encrypt network traffic and offer secure communication. [3] (SSL). Secure communication between a web browser and a server is made possible using the SSL protocol, which was created by Netscape. When it comes to secure communication, both VPN and SSL protocols are used, but there is a distinction between the two. In contrast to SSL encryption, which only encrypts a subset of data in each packet, VPN encryption encrypts the whole of each transmission. The Open Systems Interconnection (OSI) model's third layer, the network, is where VPN encryption takes place. Unlike SSL, which operates

at the OSI presentation layer, SSL functions at layer 6. VPNs, on the other hand, are often used to encrypt all communication between two destinations.

Encryption allows network and cybersecurity experts to decrease their capacity to examine network data and conceal cyber attackers. Hackers may conceal and attack security measures like antivirus and intrusion detection systems by encrypting the communications they send and receive. Open data may be analyzed with this present approach. Without decrypting the encrypted packet, it is not possible to do deep packet analysis and look at the data moving through the network, so it is not possible.

Data may be stolen, encrypted, and sent to command and control centers, as well as malware and other forms of cybercrime can be downloaded. According to the study, encrypted communication is frequently used by attackers. Preventative measures include scanning and decrypting encrypted transmissions. Due to the enormous volume of communication and the length of time necessary for decryption, analysis, and encryption, this solution's efficacy is diminished. It's also against the law to decode encrypted user communications [5], which is why it's not recommended.

Identifying and categorizing unknown network classes begins with the categorization of network traffic. In network administration and security, traffic categorization plays a key role. With this technique, network professionals can block traffic that they do not allow or want on the network. Also, which application can also do network resource management by detecting how much bandwidth it is using on it.

Since the early 1990s, unencrypted packets have been classified using certain signatures and patterns. In order to perform deep packet analysis, the contents of unencrypted packets are needed. Because it is necessary to compare the known signature patterns with the signature patterns in the captured packets. In cases where the package content is encrypted, the package content contains meaningless characters, and the signature

Due to the fact that it does not fit the pattern, this approach cannot be employed. These instances were classified using a flow-based approach. Statistics, rather than packet contents, are employed in a flow-based and time-based system, where packet length, packet count, packet size, and packet start and finish timings are all used instead. A package's encrypted communication is not classified

based on its Internet Protocol (IP) or port number. There is a lower chance that the AI-based method will work because encrypted traffic may be set up on different ports.

The application areas of Internet technology in the last few years have been banking transactions, online auctions, electronic commerce applications, social networks, etc. expanded to cover many different areas. The security of computer systems is frequently threatened by attackers (hackers). The use of firewalls and some access control mechanisms are among the most preferred Intrusion Detection Systems. However, such systems will fail if the attacker is within the network or if an unknown attack is encountered. Systems have been developed to protect against known external attacks such as viruses and worms. However, none of these systems is sufficient in today's possibilities.

The link layer is responsible for carrying the data from the network card to the internet layer according to certain protocols (Ethernet, ARP, etc.). Incoming packets at the internet layer have IP addresses. At the same time, ICMP packets used to control the internet connection are also found in this layer. Packets routed by IP address are sent to the Transport layer. Here, each packet is separated once again according to the protocol they belong to (TCP, UDP, etc.). Packets separated according to the relevant protocol reach the Application layer, which is the last layer, and are transferred to the applications they will be used according to their port numbers (80: HTTP, 21: FTP, etc. [6].

Network security is becoming more and more important due to the enormous development of computer networks and the rapid increase in the number of applications that use them. Moreover, nearly all computer systems have security vulnerabilities that are technically difficult to detect and costly to prevent. Internet included in the model the fact that protocols such as IP, TCP, etc.

Therefore, Intrusion Detection Systems (IDS), the most important network security element designed to detect anomalies and attacks in a network, become even more important. One of the main approaches in network traffic analysis applied by these systems is to detect suspicious traffic and separate it from normal traffic. In order to do this, various methods have been developed based on signature-based, knowledge-based, and machine learning. In today's internet environment, there are numerous devices with different processing power and portable computer features. It is clear

that anomaly detection solutions that restrict high processing power and performance will be much less preferred, especially when the subject is portable.

IDSs are designed to detect previously undetected attacks they should be developed. After an attack is detected, necessary measures must be taken. Therefore, detection and prevention systems should be considered together. Detection and detection system can be developed independently of the prevention system or they can be developed together. Such solutions are called Intrusion Detection and Prevention System (IDPS). IDSs can be broadly classified into two main groups. The first group is software-based systems (host-based) installed on user or server computers. The second group is designed as network-based. Network-based systems operate on their own special hardware or on separate computers allocated to them they work. A software-based IDS monitors the network card, the system calls of the operating system, and the behavior of the programs running on the computer on which it is installed. On the other hand, a network-based IDS, simply put, analyzes data traffic to and from the local network by tracking any events and communications between computers.

In short, traffic volume, Internet Protocol (IP) addresses, service ports, protocol usage, etc. all network-related criteria are taken into account. In this context, IDS makes important contributions to the overall network security by trying to identify attacks with their capabilities. Once an IDS detects a new type of attack on the network, it can record the signature of the attack in its database, making it better prepared for similar or identical attacks in the future. However, at the same time, a different, IDS running on the network may not have encountered such an attack before, and worse, it may fail to detect it when it does. As a solution for this purpose, it is thought that the findings that are considered important among the IDSs can be shared. Thus, an IDS with this capability can prevent other IDSs from making the same mistake. As a result, the importance of using a standard data exchange format becomes apparent in order to share data between IDSs from different manufacturers [7].

When working on IDSs, making comparisons in terms of their performance is a very important skill. False Alarm Rate and Detection Rate are two key performance measures that are often used. In computing, the Detection Rate is defined as the proportion of all threats that are properly detected (and so create an alert). The False Alert Rate is the ratio of cases where a normal event is assumed to be an attack to the total number of normal events. In these two cases, it is aimed to

measure the performance of the IDS when it gives an alarm. In some cases, IDSs do not generate any alarms. In other words, an undelivered alarm in the event of a real attack means poor performance again, while the IDS not generating an alarm during normal network activities also means good performance [8].

1.1. ATTACK TYPES

Due to the latest developments in technology, all communication and coordination such as electronic communication, e-mails, internet banking, and online records are shifting to the internet world. The attack can be defined as influencing communication, interfering with personal space / information without the permission of others (this can be behavior or property). With the increase in the use of the Internet, the incidence of cyber-attacks is also increasing. Bullying in the real world may be compared to cyber-attacks. Harassment, threats, and blackmail are all possibilities in this kind of assault. The attacks threaten the efficient operation of companies/individuals' online systems and valuable information. While firewalls, authentication systems, and encryption are often used to protect information and networks from cyberattacks, these techniques are rarely sufficient since attackers constantly create new detection and prevention mechanisms. They desire more dynamic and robust detection systems in their workplaces. As a result, networks and data are under risk from a wide variety of assaults. Attack types;

i. Active Attack
ii. Passive Attack
iii. Distributed Attack
iv. Insider Attack
v. External Attack
vi. Fraud Attack
vii. Password Attacks
viii. Close-in Attack
ix. Reconnaissance Attacks

There are also more diversified forms of these attacks. The most common attack types in the datasets examined in this thesis are DoS, Probe, U2R, R2L, Brute Force, Infiltration, SQL Injection, and Bot attacks.

a) DoS Attacks; Active attacks. These attacks are generally used to prevent users from getting service by sending multiple connection requests to a server over the security in the TCP / IP protocol structure.

b) Probe Attacks; these attacks are made to learn certain information about a server or any machine. For example, the current IP address or active port it's using, the operating system it's running on.

c) R2L Attacks; Attacks for unauthorized access as a guest or another user, even without user rights.

d) U2R Attacks; are internal attacks. These are the attacks in which a non-administrator user who has permission to enter the system pretends to be an administrator and performs unauthorized actions.

e) Brute Force Attacks; Password attacks. It basically aims to access the server by applying the simplest method. It aims to enter the system by repeatedly trying new user names and passwords that it has developed by considering the estimated user names and passwords.

f) Infiltration Attacks; It can be accomplished by directly breaching a network or by infecting a host that subsequently joins a private network. The attacker can exploit the application vulnerability by sending a malicious file to the victim in a simple way, such as via e-mail, and then open a backdoor and use it to exploit the computer and network.

g) Data-driven applications may be exploited by tampering with SQL statements, which is a code injection approach.

h) Bot / Botnet Attacks; Active attacks. Harmful by attackers. The virus aims to infect computers with software and to have the authority there. After the bot virus infects your system, the computer turns into a "zombie" and performs all the operations requested by the attacker. It is very difficult to be noticed by the user.

1.2. METHODS USED IN INTRUSION DETECTION SYSTEMS

Techniques have been developed for modelling damodeling, classifying the modeled data, and creating tables. The most common of these techniques are; Data Mining, Rule-Based / Expert Systems, Statistical, Pattern Processing. Artificial Intelligence techniques have been developed for the problems that are difficult to solve with the developed techniques. These techniques are; it is classified as Fuzzy Logic, Genetic Algorithms, Artificial Neural Networks (ANN), Artificial Immune systems (AIS), Support Vector Machines (SVM) [10].

a) Data mining; as accessing information from large-scale data is definable. Extraction of rules from data is made possible by establishing a connection among data and users. Some of the work done in data mining is related to intrusion detection as it is related to the detection of abnormal conditions. Data mining techniques can be used to detect anomalies. Real-time attack detection is possible thanks to the hashing and data reduction available in data mining. Normal traffic information obtained from network traffic is kept in databases and attack detection is made by comparing normal traffic with new login requests.

b) Rule-Based / Expert Systems; by experts in a particular field developed. Rules are created by examining system traffic, rules or signatures, which are definitions of known attack methods, are used to create these rules. The signatures used should be kept up-to-date to include new attacks. And this is how to attack detection is done. The request in the computer network and the predetermined attack rules are compared and the similarity of the request and the signature / rule is checked. If the incoming request conforms to the rule, the attack.

c) Genetic Algorithms: Genetic algorithms are used in STSs to apply simple rules to network traffic data. These simple rules must be defined to separate normal data from abnormal traffic data. It blends Darwin's theory of evolution by natural with problem-solving. Small-scale genetic algorithms are the norm. The game begins with a random set of rules, which are gradually elaborated upon. In order to use this approach in detecting attacks, researchers have created and processed hypothetical vectors showing or not attacking.

d) It's one of the cleverest approaches. Neural networks (artificial): Artificial neural networks (ANNs) are computer systems that process information in a manner similar to the brains but which use weighted connections and processing units with their own internal memory. They

produce new data by creating links between input-output vectors and then using their own algorithms. It's a method for seeing and learning from the actions of the system's users. Generally speaking, STSs use multi-layer perceptron Artificial Neural Networks (ANN). When it comes to figuring out and categorizing training functions, multilayer artificial neural networks are very good at it.

e) AIS is considering the human immune system has been developed. Coordinated attacks and internet wolves detect attacks by observing anomalies in network traffic patterns. The artificial immunity model was designed to distinguish normal network activities from abnormal network activities.

f) SVM is a high-performing learning algorithm created with a very simple idea and frequently preferred in IDS. They are generally used for the selection of the feature vector. It is based on the logic of finding the support vectors that linearly best separate the samples belonging to the two classes. There may be an infinite number of lines separating two classes, but Support Vector Machines must be used to find the line that best separates them. Support Vector Machines created using this idea are extremely successful classifiers. Along with the use of Support Vector Machines in many fields, they are used as classifiers in IDS [11].

1.3. RESEARCH QUESTIONS

In this study, several questions are presented to estimate the concrete compressive strength which is a very important factor in producing a robust system.

i. What is the computer network?
ii. How present t new method for network traffic classification?
iii. What is the machine learning techniques that can be applied to network traffic classification?
iv. How can we optimize machine learning techniques using effective techniques?

2. RELATED WORKS

2.1. CLASSIFICATION METHODS AND LITERATURE REVIEW

The classification of traffic is very important for network experts and cyber security experts to analyze the network. For this reason, traffic classification studies have been carried out since the beginning of the use of the internet. While port-based classification was made at the beginning, traffic classification was made by examining the contents of unencrypted packets using certain signatures and patterns [12]. Flow-based classification methods are used in cases where the contents of the packets are encrypted. In flow and time-based systems, features such as packet length, acknowledge packet number, and packet departure and arrival times are used instead of packet content [13].

2.2. PORT BASED CLASSIFICATION METHOD

Port-based classification is the most intensive and oldest method used to classify traffic using Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) port numbers [12]. In this method, classification is done by looking at the port number information in the packet header. The classifier classifies the port number as HTTP when it sees 80, Secure Shell (SSH) when it sees 22, or Telnet when it sees 23. Port numbers for applications are given by the Internet Assigned Numbers Authority (IANA) [14].

Plonka [15] developed a tool called Flows can that reports and visualizes network traffic. This tool is capable of collecting and processing raw traffic data. After capturing the traffic, it can separate the Internet Control Message Protocol over IP (Internet Control Message Protocol, ICMP), TCP, and UDP. However, by looking at the known port numbers, it can divide the traffic into categories such as File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), and HTTP. In addition, it can understand whether it is a local IP by looking at the IP information. However, since the vehicle cannot capture the content of the traffic, it can only classify by looking at the port number. This reduces the classification reliability.

Moore [16] and colleagues created a tool called Coral Reef for network professionals to monitor the network and solve problems. This tool listens to the outgoing traffic over the network and provides the classification of the real traffic passing over the network and the hardware

performance information of the network devices to be seen by the network experts. The coral reef has a library called App-Ports to map protocol and port information. It receives port information from the traffic it captures and performs classification through this library.

Fraleigh [17] et al. classified the applications by looking at the port numbers in their study to be able to calculate the traffic on the Sprint IP backbone. In this study, similar applications were classified according to categories such as web, mail, file transfer, Peer to Peer, P2P, and Streaming. Web connections can be sub-categorized as HTTP or HTTPS, mail as SMTP or Post Office Protocol (POP3), and file transfer as Secure Copy (SCP) or FTP, depending on the port. Since P2P connections use different ports, they could not be classified and a class named alias was created and put here.

The Other category includes applications using unknown ports. The computational cost is very low, as the classification is made only by looking at the port number. However, since it is only used for port numbers, its reliability is also very low. Because most applications can use non-standard port numbers or run applications different from known port numbers [18].

2.2.1. Content Based Classification Method

Although the classification process is easy by only looking at the port information, its reliability is very low. For this, studies are carried out to classify the traffic by examining the contents of network packets. The process of analyzing the package content by examining it is called Deep Packet Inspection (DPI) [12]. Although the classification method by examining the package content is a costly solution compared to the port-based classification method, its reliability is much higher. Analysis and classification are performed by comparing the package contents with known patterns or signatures. This method is the most widely used in today's cyber security solutions. Intrusion detection systems perform malware detection and application classification by comparing their signature databases with package contents. However, artificial intelligence algorithms are also used extensively for deep packet analysis. Because signature-based systems use only known patterns, they are weak against unknown situations. For P2P applications that port-based systems cannot classify,

Sen [19] et al. carried out application classification using packet content analysis and specific signature patterns. Signatures were created by analyzing documents and captured packets. The created structure was tested on high-bandwidth networks and the false positive rate was below 5%. When the structure created is compared with port-based studies, quite successful results have been obtained. Kazaa P2P application has been classified 3 times more successfully than port-based systems.

Moore [20] et al. carried out a study to eliminate the disadvantages of port-based systems by using the packet contents of the traffic they collected over the internet. The data set is prepared by capturing the internet traffic of 1000 researchers from the university campus network. 94.81% of the generated dataset is TCP, 3.58% ICMP, and 1.57% UDP. With the created structure, close to 100% accuracy was achieved, but no analysis was performed on live traffic.

Fennimore [21] and colleagues developed a tool called KISS for the classification of UDP-based Voice over Internet Protocol (VoIP) and P2P applications. Compared to normal content review tools, KISS has the advantages of faster updating, easy applicability, and less resource consumption. Performance measurement was carried out using the KISS tool on the test dataset and live traffic. SVM based decision mechanism is used for feature extraction. In the study, the true positive rate was 99.6% and the false-positive rate was less than 1%.

Although content-based classification is a very reliable method, it has some fundamental weaknesses and shortcomings. It cannot primarily analyze encrypted or scrambled data, and therefore cannot analyze most of today's traffic. In addition, since content analysis is performed, it may be contrary to the privacy policy and laws. In addition, it is very difficult to analyze high-speed traffic because it requires a lot of computational power and time [22].

2.2.2. Flow Based Classification Method

Detailed parsing of packets requires the unencrypted content of the package, and the parsing process itself takes a lot of time and computing power. These methods are not available if the package contents are encrypted. In these cases, flow-based statistical methods are used. In streaming systems, classification is done by examining the statistics of packets on the network, rather than information about the ports or the contents of the packets. This method does not use

the IP address and port information as a function. Traffic can be generated on several ports, which reduces the success rate of AI-based methods [23].

Al-Shammari [24] and others conducted a survey to classify SSH and Skype traffic. As part of our research, we are developing binary classification applications for SSH, No-SSH, Skype, and No-Skype. Five different ANN algorithms were used to solve this problem and the C4.5 algorithm was found to provide the best results. Algorithm C4.5: Worst Case Detection Probability 83.7%, False Positive Rate 1.5%, Best Case Detection Rate 97%, False Positive Rate 1.5%, 0.8%. Price list.

It is the work of Dimauro et al. The ability to communicate in real-time (WebRTC) Communication in real-time (WebRTC) Communication in real-time (Real-time communication) the ranking of online apps was investigated. The dataset was developed by the author using Tshark, MySQL, and Weka programs. The author of this work does binary classification using four distinct machine learning techniques. According to research, using the Random Forest method yields the greatest outcomes. Adjusted for this method, the F1 response rate was 96.4 percent and the Random Forest response rate was 96.4%.

Data on time flow were collected by Draper-Gile [26] and colleagues in 2016. These two situations are included in this dataset. For our research, we categorized both encrypted and unencrypted VPN traffic. All in all, there are a total of 14 distinct tags. The C4.5 and K-Nearest Neighbor (KNN) algorithms were used by the author in this work. The first and second scenarios' success indicators are described in great depth. According to research, the C4.5 algorithm outperforms the KNN algorithm. Using the C4.5 method, only 90.6 percent and 80.9 percent were found to be at their optimum sharpness using the C4.5 method, respectively. Seddig and colleagues [27] Studying the categorization of encrypted packets in high-speed networks as the goal of the researchers. From the campus network, the data set was gathered. Despite the fact that there are more than 200 options, we picked this one because of the significant volume of traffic. There are six different types of properties in this dataset. The Machine Learning Traffic Analysis Tool (MLTAT) is a platform built using six distinct machine learning methods (MLTAT). Vinyl and other categories were also used in the selection of hyper parameters. All of our assessments had an accuracy rate of above 88 percent, but the web surfing class was far below average.

As of 2018, Caicedo-Munoz [28] et al. VPN and non-VPN traffic were categorized based on their Quality of Service (QoS) in research. Draper-Gil [26] generated a non-VPN ISCX VPN dataset for this investigation. Five different ML algorithms have been used in this investigation, and the results are interesting. There was no discernible difference between the first and second scenarios in terms of compression and expansion performance. 92.82 percent of VPN traffic is correctly routed, whereas 94.42 percent of non-VPN traffic is correctly routed in Scenario 1. The second scenario had an accuracy of 86.94 percent.

Saqib and colleagues categorize VoIP data, the author uses static analysis techniques, rather than packet content. Applications, security protocols, and cryptographic techniques are not used to identify speech traffic in the network in this investigation. I searched the internet to find the most popular programs, such as Skype and Yahoo. In order to categorize the packets based on their nature and relationship, we utilized the packet size as a guide. Low diversity and high throughput are the characteristics of voice traffic that have been discovered. In order to classify VoIP communication that has been encrypted, more information must be taken into account, such as the number of packets, the total amount of time it took to send, the arithmetic mean, the standard deviation, and the number of packets. Offline traffic: 93.6 percent of it, 100 percent of VoIP traffic that was intercepted by attackers, and 95 percent of online traffic were found thanks to the frames that were made.

In the [30] passage, Stereo Transformation Neural Networks (STNN) was developed by Zhang et al. to categorize data that is protected using Transport Layer Security (TLS) or SSL. For the creation of STNNs, an approach combining LSTM and CNN techniques is used. In all, 17 applications were used to get the data. They were able to obtain an accuracy rate of more than 98%.

There is Chari [31] as well as [48]. An algorithm for encrypting traffic categorization. ISCXTor2016 data was utilized as the dataset for this study, and packet length was employed to select characteristics for analysis. Ninety percent of the students in all six courses passed the exam.

[32] Pradaneta. The author investigates the network's ability to identify and classify encrypted SSH traffic. In our research, we employed a hybrid radial basis function (RBFN) network-based machine learning technique. Gradient descent and K-means pooling are used in the RBFN

feedforward neural network. Ada boost, Decision Tree, Naive Bayes, and Random Forest algorithms were tested against RBFN's findings. For certain data sets, RBFNs are very accurate over 99 percent of the time.

[33] Researchers from Yang et al. deep learning neural networks and CNN algorithms are used to encrypt traffic using TLS or SSL. Let's take a deeper look at the auto-encoder and CNN settings. The TLS/SSL traffic from the top 500 Alexa-served pages was used to construct two datasets (pages 10 and 50) for deep learning. There were five machine learning algorithms compared to the Auto-encoder and CNN algorithms that performed the best. The accuracy of CNN's 10 online entries was 96.46 percent, while the accuracy of its 50 web entries was 85 percent.

Using a machine learning method, Al-Obeidi and his colleagues were able to identify and detect programs like Skype and WhatsApp that interact via encrypted channels. SVM, MLP, Naive Bayes, and C4.5 machine learning techniques were all used in this work. Wireshark was used to capture the data from the end user's machine. There are 14 items in the produced dataset. It used the C4.5 classification algorithm for the first tests, and it was able to classify Skype, WhatsApp, Facebook, and YouTube 88% of the time. The Netflix app was introduced to these four other applications and was evaluated in five more. The C4.5 algorithm was found to be the most accurate, scoring 86.33 percent of the time, making it the most efficient.

Research done by Khatuni [36] and colleagues evaluated encrypted traffic in audio, video, and social networking apps. They used the application's features to visit the website automatically and construct a dataset using Firefox and Chrome browsers. A total of thirteen machine learning algorithms were used in this investigation. Even while it enhances the random forest method's performance, the decision tree approach uses more resources. The random forest approach was seven times slower than the decision tree technique. According to research, the decision-tree algorithm is the most effective method for addressing problems. The true-positive rate is 85%, whereas the false-positive rate is 5%.

2.2.3. Other Techniques Used in Network Security

A detailed packet analysis requires the unencrypted content of the packet, and the analysis itself takes a lot of time and computing power. These methods are not available when the package content is encrypted. In these cases, statistical flow methods are used. In streaming systems,

classification is done by looking at packet statistics on the network rather than port information or packet content. This method does not use IP address and port information as a function. Traffic can be generated on multiple ports, which reduces the success rate of AI-based methods [23].

SSH and Skype traffic were studied by Al-Shammari [24] et al. SSH, No-SSH, Skype, and No-Skype are all part of our study, and all have binary classification applications. The C4.5 approach was determined to be the best of the five machine learning algorithms employed to handle this issue. Best-case detection rate is 97%, with a worst-case detection probability of 83.77%. The false positive rate is 1.5 percent, and the detection rate is 97% for the algorithm C4.5. A list of prices.

It is the work of Dimauro, et al. Communication in real time (WebRTC) The ability to communicate in real time (WebRTC) Communication in real time (Real-time communication) there are a number of web apps on this page. The author used Tshark, MySQL, and Weka to build the data collection. Four distinct machine learning methods are used in this work to achieve binary classification. The random forest method has been shown to be the most effective. The adjusted F1 response rate is 96.4 percent, whereas the baseline F1 response rate is also 96.4 percent.

Several years ago, Draper-Gile [26] and colleagues built a time-based dataset. There are two separate situations represented by the data in this collection. Encrypted and unencrypted VPN traffic were categorized as part of our research project. There are a total of 14 category tags. The author employed the C4.5 and K-Nearest Neighbour (KNN) algorithms in this work. Details of how the first and second situations will go down are given in great detail. According to recent studies, the C4.5 method outperforms the KNN algorithm. In the first and second scenario testing, only 90.6 percent and 80.9 percent of the C4.5 algorithm's maximum sharpness values were found, respectively.

It is an extension of the classic color pattern with an element called "cleanliness" used to describe hygiene. Based on the new model, the authors propose a static method for analyzing previous drawing data to identify weaknesses in the drawing style. The four key phases of this strategy are as follows: Gathering groups to form context, dragging variables through a block's base block, and lastly finding variables that function calls will travel through are all steps in this process. Test findings suggest that POSE, a tool that implements this strategy, is effective at discovering web application security holes. Outbound vulnerabilities are a special class of vulnerabilities in web

applications, and research focuses on how to detect them. Extend the traditional blot-style vulnerability model with an element called clean-up, which is used to specify the clean-up procedure. Based on the extended model, reverse static data analysis methods have been proposed to detect web application security vulnerabilities in PHP code. This method consists of four main steps, the first step is to collect groups and create a context and the second step is to go back. Monitoring of data transmitted during data collection. The base block is the third block that looks for variables between blocks, and the last block looks for variables that pass function calls. A tool called POSE implements this method and test results show that this method is suitable for finding vulnerabilities in web applications. There are still many false positives regarding the use of static methods to detect web application vulnerabilities and could suggest artificial intelligence solutions to counter this situation [39].

An essential consideration in the software development lifecycle is ensuring the security of web applications. Simple attacks from application vulnerabilities may harm an organization's reputation, abuse sensitive information of application users, steal money, or render resources unavailable. CVE estimates that there have been 86,879 vulnerabilities discovered since 1999. There were 14,518 new flaws found in 2017 alone. For a particular year, this surpasses the known vulnerabilities. Consequently, it is critical that developers who are well-versed in web application security be able to patch any flaws that may exist. This article compares and contrasts the OWASP WAP and RIPS source code web application vulnerability scanners. In one experiment, WAPO outperformed WASP RIPS in terms of performance. But it should be noted that no browser is completely secure, particularly when many flaws have been found. Improved performance is available in the commercial version of RIPS. There was no commercial version. Furthermore, I'm only interested in technologies that are accessible as open-source. Our web application's source code must be tested in order to acquire outcomes in real-world circumstances, the writers discovered. Commercial static vulnerability assessment tools will be evaluated using current open-source tools (if feasible) and open-source static analysis will be used to establish the applicability of commercial techniques when prices are minimal [40].

With the development of the Internet and networking technologies, a distributed system structure based on the network environment has emerged. Web services, on the other hand, have become popular recently, with efficient functionality and increased interoperability to handle cross-

platform heterogeneity. Service-Oriented Computing (SOC) and Service-Oriented Architecture (SOA) have become one of the dominant research areas in academia and industry. SOA can be seen as a kind of software architecture that calculates the resources of each service and takes into account the implicit services. It is a conceptual model that leaves decisions such as networks, transport protocols, and security to specific applications. However, Web Services Architecture is a business model designed for technical managers. Indeed, it can be seen as a concrete implementation of SOA. Configuring Web Services helps solve complex Web applications, increases the reusability and efficiency of primitive services and components, and reduces system build costs. Web services include features and make them available to network users. A service is an entity consisting of a set of interfaces and functions. The authors developed a DSA-based configuration architecture model to enable Web services to adapt to ever-changing conditions and requirements. Therefore, Web services configurations can be analyzed at the level of architecture that supports dynamic development. This enables DSA-based service configuration using the software architecture description language (XYZ / ADL). In the future, we plan to use XYZ / ADL [41] to check that the web service configuration description is valid.

One-size-fits-all approaches to reverse engineering online apps no longer cut it in today's world of complex web applications. Generic extractors may be challenging to create because of the growth of client and server technologies. Due to visual code, it is difficult to interpret actions based on information retrieved from the source code of most modern online apps. Modern Web applications, both server-side and client-side, make extensive use of frameworks and libraries, making it hard to fully analyze the source code. Co-application tools assist reverse engineering and re-use according to systematic ad hoc offerings as they are increasingly incorporated into web application development structures and processes. Mobile apps have lately been shown to benefit from many of the same static and dynamic analysis approaches that were initially designed for web applications. Both contexts depend heavily on synchronous and asynchronous client/server interactions, as well as on events and graphical user interfaces. The authors, for example, recommend using dynamic analysis to reverse engineer Android-based mobile apps. The purpose is to construct a thorough UML model of a mobile application's user interface and interactions between its components [42].

Static API analysis approaches that promote modularity are the focus of this work, which tries to solve these concerns. Using an object-oriented paradigm, the word "object" is represented by the phrase "legal entity" in the technique provided here. Analyze interfaces in a systematic approach to gather data on business entities and characteristics, discover the sequence in which calls are made between various business entities, and investigate new ways of invoking APIs to access services. Since it's an open-source tool, this method may be used for a broad range of commercially available verification services. You'll learn a lot about API structure and behavior from this research. As a result of this, service users will be able to more easily comprehend and apply complicated and perplexing interfaces. Services are seamlessly integrated. Service variables may now be recovered with the use of a Monte Carlo sampling approach. The fundamental advantage of this strategy is that it may be used to conduct trials across wide areas of study and yet get decent results. Brute force approaches like this can't create choices when the search space is really vast. Using this strategy also has the advantage of requiring minimum human interaction, preserving only well-known routes (i.e., service alternatives that are accepted), and providing more acceptable service options It is meant to be automatically picked. Only significant sampling is presently available for distributed optimization in Monte Carlo simulations [43].

To detect and repair online application vulnerabilities, this article offers a PHP programming technique and tools that implement input validation bugs. Three host machine learning classifiers are employed to detect false alarms, and an inductive rule classifier is used to validate their existence. All classifiers were selected after a thorough evaluation of the various alternatives. To be clear, this combination of detecting techniques does not provide totally accurate findings. It's impossible to solve the static analysis issue with merely probabilistic outcomes from data mining. Clean-up functions and h-checks are among the fixes added by this utility. To guarantee that the patch resolves the vulnerability and does not damage (fix) the application, tests are conducted. Throughout the early 1990s, the World Wide Web has grown from a platform for viewing text and other multimedia information to a platform for executing web applications. Many open-source PHP implementations and susceptible synthetic programs have been evaluated with this tool. It was also compared to Pixy and PhpMiner II, two other tools for analyzing PHP source code. Shows that scheduled sessions can be protected against vulnerabilities by the tool. A total of 388 flaws were discovered in 1.4 million lines of code by the researchers. About 5 percent more precise and accurate than PhpMiner II and 45 percent more accurate than Pixy.

In this article, the author proposed a static analysis approach to create automated tests for web applications. Compared to dynamic analysis methods, the static analysis approach can have the following advantages: A more detailed web application interface that cannot be identified by browsers. Improve input parameter values in web forms. Improve targeting during testing by using field information for inputs to provide goods or bad values for inputs. However, there are some limitations to the analysis. First, web applications can be implemented using a variety of rapidly evolving techniques and programming languages. However, as an approach to static analysis, our focus should be on a specific source language. For now, our discussion has only focused on web applications implemented as JSP methods. Second, if your web application is complex, the web navigation scheme can be large. Our approach uses the slice method to simplify analysis. However, even with this simplification, the navigation map can still be large. In the future, a hierarchical approach to serving web applications will be considered to avoid further state-space explosions. In this post, I proposed an automated static analysis approach. Create a test case for your web application. This approach parses the application source code and extracts the interface from the web application source code. It consists of a custom navigation map and input parameters for all possible URLs and information fields and uses a navigation graph to determine the order. Examples and test cases are created for each route. This approach is demonstrated by a simplified analysis of the JSP access validation page of the student information management system [45].

Smart technology, industries, and applications have advanced greatly thanks to the internet. Networking technologies are essential because they allow the cyber intelligence world to disseminate information and answer a high number of online enquiries. Networks are becoming more significant in terms of security and service quality as they proliferate in contemporary society (QoE). Security services providers are increasingly relying on efficient and trustworthy networks. Optimizing machine learning algorithms has a beneficial impact on the performance of all three datasets, according to this research. The ADFA-IDS 2017 dataset, in particular, exhibited the most significant changes. Traditional machine learning approaches used in the decision tree algorithm produced the best results across all three datasets, regardless of optimization. Even after optimization, the K-means clustering algorithm still performed poorly. RNNs outperformed the rest of the NNs in all three datasets, according to the results. Training and testing indicators are not widely varied, but the findings reveal that the best training/testing ratio is 80% training/20% testing, 90% training/10% testing. You're welcome to join. DDoS assaults will become more lethal

as more gadgets connect to 5G cellular networks. It is possible that DDoS assaults are more damaging than other types of cyberattacks. The majority of network traffic is generated by mobile devices. The new DDoS assaults are significantly smaller and harder to detect. The empirical assessment of machine learning in 5G-enabled mobile networks against security assaults, in particular DDoS [46], will be an important focus of future study.

This article describes how to find bug-like vulnerabilities in PHP code. The classic stain pattern is complemented by an element called "cleaning", used to represent hygienic procedures. Based on the new model, the authors propose a retrospective analysis method of static data transmission to identify transmission vulnerabilities. This method consists of four main steps. The first step is to create a group of contexts, the second is to map the variables to the base block, the third is to find the variables between the blocks, and the last is to find the variables. Submit a function call. A tool called POSE implements this method, and test results show that this method is suitable for finding vulnerabilities in web applications. Transmission vulnerabilities are a special class of vulnerabilities in web applications, and how they are detected is of particular interest to researchers. We've extended the traditional blot-style vulnerability model to include the clean-up element, which is used to specify the clean-up procedure. Based on the extended model, the authors propose a promising method for analyzing static data to discover vulnerabilities in web applications in PHP code. This process includes four main steps. The first step is to create the pool and the context, the second step is data within data. Recreation. The base block looks for variables between the third blocks and the last block looks for variables that pass function calls. A tool called POSE implements this method, and test results show that this method is suitable for finding vulnerabilities in web applications. So far, there have been many false positives regarding the use of static methods to detect web application vulnerabilities, and the authors can offer artificial intelligence solutions to solve this situation [47].

In recent years, many types of web applications have grown rapidly around the world. Unfortunately, attacks against these apps are on the rise. Therefore, you need to protect your web application against vulnerabilities, but finding all web vulnerabilities manually is tedious and requires a lot of effort and skill. Therefore, it is imperative to use a vulnerability scanner for your web application. There are many commercial and open-source tools to assess static and dynamic vulnerabilities. This article has described two web application vulnerability detection tools, the

OWASP WAP and RIPS source code analysis tools. In the OWASP experimental scenario, WAP was found to be superior to RIPS. However, keep in mind that no browser is perfect, especially when it comes to detecting attachment vulnerabilities. The commercial version of RIPS may work better because it comes with a newer version. The commercial version was not available. Also, I'm only interested in the open-source tools available. I also realized that to get results in a real-world scenario, I needed to test the source code of a non-vulnerable web application. Enough to compare commercial static vulnerability scanner tools (if possible) with existing open-source tools and use open source static scanner tools in the absence of commercial tools for future work. I want to determine if this is the case. They are informative. Vulnerable web applications have been intentionally exploited to easily scan detected vulnerabilities to identify true or false positives. In experimental scenarios, OWASP is superior to WAP RIPS [48].

It is now common practice to use a web application firewall (WAF) to protect online applications against attacks that exploit vulnerabilities. Web Application Firewall (WAF) system discovery is examined in this research using sophisticated learning approaches. For the purpose of detecting and blocking attacks, Waf examines HTTP requests sent back and forth between the client and the server. Deep learning approaches may be used to generate a function extractor for a single class. Transcoding architecture and a self-monitoring neural network are used to develop a deep language model for HTTP queries. For NLP activities, learning may be easily transferred and improved by employing an out-of-the-box language model. To convert an HTTP request into a function vector, use the pre-trained model as a function extractor. The class classifier is trained using these vectors. Performance measurements are also used for the purpose of calculating the optimal operating point for a single-class model. The suggested technique outperforms typical rule-based Mod Security equipped with vanilla Owasp Crs in experiments and does not need the participation of security professionals in the definition of functionality [49].

Because to advances in machine learning, MOD SECURITY has been able to enhance its detection skills by decreasing false positives and boosting the number of accurate discoveries. It also describes the issue by describing several possible outcomes based on the accessibility of relevant training materials. Real application traffic logs are ideal in the rarest of circumstances, but they are not required in the majority of instances. When doing functional testing, for example. According to this research, the strategy employed to enhance MOD SECURITY performance has been found

to be effective by the writers. Machine learning algorithms can rapidly define decision limits based on training data but constructing intrusion detection rules is a challenging effort for human specialists. Findings from sc2 show the necessity of having a representative attack learning example, as shown in the prior results. Despite the low false positive rate, intrusion detection is ineffective. The authors plan to use automated methods to develop synthetic assaults in the future. Final scenario (sc3) results demonstrate the single class solution yields many processing points in excess of Mod Security, even when only legitimate requests are considered. They will discuss how to automatically identify trigger sites using selective or synthetic attacks in the following tasks, as in the sc2 scenario.

According to the authors, the outcomes of the three situations presented here prove that machine learning can be used to develop WAFs. Single-class algorithms like SVM will be used instead of distances to investigate particular single-class situations that solely include attack requests. The absence of publicly available logs including entire HTTP requests was one of the greatest problems we encountered. In all, only three datasets could be located. A third dataset (1998 DARPA Intrusion Detection Evaluation Dataset) was created by network traffic and hence just removed web application traffic, which I did. Two of these datasets were utilized in the experiment. The datasets are also almost a decade old. These datasets, according to the authors, are no longer representative of contemporary technological developments. In the future, more datasets will be created by the authors.

This white paper describes an architecture that implements an end-to-end unsupervised deep learning approach to automatically detect web application attacks and their consequences. Use and analyze your web application with a powerful software modeling tool (RSMT) that automatically tracks and characterizes the behavior of your web application at runtime. Next, I used the automatic noise reduction encoder to show a detailed view of the call trace captured when launching the app. To test the intrusion detection system, the authors created several test applications and synthetic monitoring datasets to evaluate the effectiveness of unsupervised learning for these datasets. Cross-validation is commonly used in traditional machine learning but is not commonly used to evaluate deep learning models due to high computational costs. The authors needed to compare our automated coding approach to other machine learning methods. No cross-validation was used in our experiments to ensure a fair comparison. In the future, the authors plan to consider more

complex network structures like LSTM auto-encoder and CNN auto-encoder. The authors wanted to see if these releases could improve the accuracy of web attack detection activity. The results show that auto-encoders are superior to other approaches, but there is still work to be done to prove their effectiveness against zero-day attacks. This result hopes that automated cryptosystems can detect zero-day attacks, but more research is needed in this area. The main purpose of this study is to evaluate the effectiveness of automated cryptosystems against unknown attacks. Determining the frequency of remodeling is also an open question that needs to be explored in future research. After training online with real usage data, the potential for integrating attack data into regular behavioral datasets increases. Excessive use of the app can resolve this issue and lead to accurate detection, but this hypothesis requires further investigation. There are also plans to distribute the machine learning analysis workload to remote machines and to develop mechanisms to support coordinated distributed discovery between hosts [51].

Detection and categorization methods for web application assaults are presented in this work. Rather of using signature-based security techniques, we've developed a system based on ontologies. Semantic rules, result context, and application logging may be used to identify online application assaults. It is possible for the system to identify more sophisticated assaults by carefully examining the request of a compromised user. Application context, possible threats, and protocols utilized may all be captured using semantic rules. Complex polymorphic versions of web application assaults may also be detected using these rules, which can be developed from the ontology model. The Web Ontology Language (WOL) was used to construct the ontology model (OWL). The Apache JENA framework is used to build inference rules, which are time-based. Because of this, the system is platform and technology agnostic. Make sure the ontology concept specifications are error-free and consistent before testing the system using Onto Clean. Our technologies outperform current state-of-the-art solutions in terms of detecting capabilities and performance. Web application assaults are properly detected, however a small number of false positives are generated as a result of the approach. A semantic method can efficiently identify zero-day and more sophisticated assaults in a real setting, as shown in the examples given.

Application layer detection may be improved by using an attack taxonomy and communication protocols. A prototype web application firewall is used to test different attack methods and associated vulnerabilities. For the first time, semantic technology may be used for online

application security. Use web semantics with information security to create synergy. As a result, web application security research has reached a new level of sophistication. The system uses the HTTP protocol ontology model to interpret and evaluate incoming HTTP requests. After verifying the request is legitimate, the program goes on to its next component. The parser uses the rules and ontologies contained in the knowledge base to contextually analyze and identify attacks on the extracted material. The test system surpassed current systems like Mod Security in test protocols and intrusion detection for the top ten OWSAP threats, using an ontological approach to testing. On the other hand, the ontological model may be expanded and enhanced as required. Attack scenarios for web applications and HTTP communication protocols are defined in this model. It focuses on the weak points in the HTTP paradigm that might be exploited by new attack methods. As increasingly sophisticated assaults on web applications are discovered, models with inference skills are able to draw conclusions on many claims. Models that rely on semantic rules may process more data more quickly. An increase in speed and a large decrease in the area of search.

One of the most prevalent online application vulnerabilities is cross-site scripting (XSS). A user's or a website's security may be jeopardized if this flaw is exploited. This vulnerability is found in the PHP source code using a variety of tools and methodologies. Even yet, XSS detection in PHP online applications remains a concern for the time being. For the most part, static analysis has been utilized to discover XSS vulnerabilities. As a matter of fact, you can get almost complete code coverage and track all program routes using this technique. Studies have demonstrated that static analysis is more successful than other techniques in discovering these vulnerabilities. The detection and execution speed of static analysis has been enhanced by combining it with other methods (e.g., evolutionary algorithms, pattern matching, machine learning, etc.). Static analysis issues persist, and the frequency of false positives is substantial, even when these approaches are used in conjunction with static analysis. In order to avoid false positives while looking for XSS vulnerabilities, static analysis and genetic algorithms are the only methods that work. This method, however, is labor-intensive and unsuitable for complex online applications. As a result, this strategy has the potential to be automated. When it comes to finding XSS vulnerabilities, additional techniques such as (Particle Swarm and Ant Nest Optimization) have yet to join the market [54].

The current state of cyberattacks threatens even the most secure systems. Attackers use new technologies every day to penetrate systems and exploit the smallest vulnerabilities. In such

scenarios, it is important to design and develop web applications that are not only secure but also flexible. Security is the most important aspect of a web application, but reliability should also be a priority. This is because performance improves when the web application provides both to the user. In this context, it is important to consider mechanisms that effectively assess the irretrievability and security of web applications during development. In this article, the authors used the Fuzzy Analytic Hierarchy Process (Fuzzy AHP) to evaluate the effectiveness and performance of effective global security in achieving a high level of security in web applications. Additionally, this study tests the AHP fuzzy method on 10 consecutive releases of enterprise web applications to test the effectiveness of this approach. Despite the privacy of web applications, different versions of web applications from three universities were used in this study. The results were more complex and far more efficient than the robust security testing process of existing web applications. In addition to saving time and resources, the mechanism recommended in our research is to create a pattern that web developers can follow to optimize security when developing web applications [55].

With the recent development of cloud computing, applications are hosted by cloud application providers instead of being purchased from cloud service providers. For example, cloud provider Amazon EC2 allocates cloud resources to application providers based on virtual machine scalability. The provider is responsible for using the web application with cloud resources. Static cloud proprietary deployments are considered inefficient when the web application provider pays attention to the dynamics of the web environment and the needs of end-users. With static deployments, frequent user requests can lead to under-deployment, delayed responses, and frequent dropouts of user requests. Additionally, reduced IoT traffic often leads to over-provisioning issues and increases the cost of services provided by application providers. Recently, web-based applications such as banking, online shopping, and news browsing have become popular. This study proposes to use the Montgomery recursive multiplier to reduce the inner product of the ECC security protocol. Processing time has been reduced by implementing the recommended protocol for Web Application Cloud Service. Additionally, Montgomery's lightweight ECC multiplier reduces encryption and decryption times. This improves the network scalability of web applications on cloud servers. Therefore, the authors can conclude that the proposed scheme has a lower computational load for cloud web services than other security

protocols and provides better security. Moreover, the proposed method can be integrated with other cloud services to improve the security of other methods [56].

2.3.THE ORIGINALITY OF MODERN INDUSTRIAL SYSTEMS IN TERMS OF INFORMATION SECURITY

Intelligent transportation systems based on decentralized VANET networks, which include various smart devices that provide vehicle communication, monitor the traffic situation, and inform the traffic participants, are developing in many countries [57]. Especially in Spain (an intelligent urban transport system in Barcelona), Great Britain (an intelligent monitoring system to modernize the subway, London), Japan (a system to diagnose and predict the transport situation in Tokyo), etc. In the Russian Federation, smart traffic lights are used in many cities, and measures are taken to create smart urban transportation within the framework of the Smart City standard adopted in March 2019. Production substations include the Siemens AG factory (Germany, Amberg), which produces mechanisms for various automotive companies (eg for BMW) [58]. Management claims that about 75% of the facility is automated, with 1,150 employees mostly working on computers and controlling the production process. German chemical giant BASF has partnered with an artificial intelligence research center to launch a pilot smart factory in Kaiserslautern, which uses IoT technology to produce fully customized shampoo and liquid soap packages.[59] The customer, who wants to place an order, uploads the sample order to the system and determines which fragrance to the production machines in the factory using RFID technology It is informed that soap or shampoo is needed and that the bottles should be labeled with their contents. German automaker Audi introduced the "Smart Factory 2035" in 2015: the factory of the future: no assembly lines, modern flexible production stations. Currently, a significant part of the declared functionality has been implemented. At the newly opened Audi Hungary electric motor plant in Gyr, the automaker introduced a new production concept consisting of modular assembly. Production is carried out at special production stations, and the pieces of equipment produced are transported between stations by unmanned transport systems. Thus, around 400 electric drives can be produced per day with just 100 employees. Bosch's Homburg plant in southern Germany achieved a 10 percent increase in production and a 30 percent reduction in inventory balances over a year, thanks to the use of smart technology. The plant, which used to make hydraulic valves for mobile machines such as

tractors, had a manual production line. He created six basic types of hydraulic valve parts. Bosch needed to improve costs and delivery times for a line that needed to be very flexible. Smart technology reduced logistics and installation times from 450 seconds to 0 seconds for one year. Days spent on inventory halved and lead time reduced by 8%.

The result of the plant was a savings of €500,000 in the first year alone.[60] Numerous smart devices that make up modern SS have vulnerabilities that allow attackers to use such devices as illegitimate entry points to SS. The heterogeneity and a large amount of data generated by network devices, servers, personal computers, as well as smart devices, only complicate and slows down the security analysis process. With large amounts of data, anomalies caused by low-intensity and short-lived computer attacks also become invisible. The PS has a communication medium that, in conjunction with the mutual control of each other through data exchange, makes it possible to assume that data flows from the PS components will not always be controlled by humans. This leads to security issues associated with hidden, illegitimate influence on the functioning of the system. PSs implement irreversible physical processes along with reversible information processes. The information processes of such systems are aimed at ensuring the correct operation of the network infrastructure of the SS, which creates the conditions for the flow of physical processes. Even a slight runaway of the physical process achieved through a devastating information impact on the parameters of the network infrastructure can have disastrous consequences. For PS, early detection of computer attacks is particularly important; this makes it possible to obtain a certain time margin for a reaction that counteracts the violation of the correctness of the course of physical processes. Considered features lead to a number of information security issues:

a. the problem of detecting computer attacks on substations;

b. The problem of predicting the behavior of PS components;

c. Problem of resisting computer attacks.

The problem of detecting computer attacks is due to the active digitization of critical infrastructure components, their integration with smart devices (IoT devices). Typically, manufacturers of such devices focus on the large-scale sale of their products, which requires a fast pace of production. In such cases, the security level of the devices created cannot be high, a comprehensive testing and security checks take time and involve experts. A similar situation is observed with the release of device firmware updates - updates are released quite rarely, and a significant part of security issues remain unresolved. The importance of the problem of preventing computer attacks, especially in terms of recognizing attacks, is underlined by the increasing trend in the number of computer attacks against industrial and critical infrastructure over the past few years. With the digitization of technological infrastructure, an explosive increase is observed in the number of computer attacks against critical infrastructure objects represented by automated and cyber-physical systems all over the world.

At the end of 2015, Ukraine suffered a large-scale computer attack on the national electricity grid, leaving more than 600,000 people without power.[61] According to Kaspersky Lab ICS CERT researchers [62], the number of holes in IoT devices integrated with industrial organizations has increased. The report also states that 63 vulnerabilities were detected in industrial and cyber-physical systems in 2017, and automated systems that control critical energy and production processes of various enterprises became the leader in the number of vulnerabilities. There has been an increase in the number and scale of computer attacks on critical infrastructure facilities in Germany since the end of 2018. The number of computer attacks on critical infrastructure systems more than quadrupled in 2018 (34 computer attacks in 2017) to 145 computer attacks in 2018). It is also used in critical energy systems, information telecommunications, transportation, health, etc. In early April 2018, we became aware of computer attacks on four major gas companies in the United States in March: Boardwalk Pipeline Partners, Eastern Shore Natural Gas, Oneok, and Energy Transfer. As a result of the attacks, some information systems did not work for several days - they were closed for security reasons. Energy Transfer has shut down EDI, a customer data exchange platform designed to speed document transfers and reduce costs. Disabling EDI systems does not stop the gas transfer, but significantly complicates business processes as companies must look for workarounds to interact with each other [63]. On March 7, 2019, a major accident caused by a computer attack occurred in Venezuela's largest hydroelectric power plant, El Guri, which

meets 80% of the country's electricity needs.[64] The ensuing power outage, dubbed the largest in the history of the Latin American republic, shut down most of the states. The accident hit the transportation system and utilities hard. According to Rostelecom-Solar's analytical report on cyber-attacks against Russian companies in 2018, the number of attacks aimed at gaining control over IT infrastructure increased by 20% in the second half of the year [65].

The attackers aim to sneak into the infrastructure to conduct further investigation and gain deep access to information and technological systems. Using the JSOC information security event monitoring and response service, Rostelecom-Solar experts recorded more than 765,000 computer attacks in 2018, 89% more than in 2017. The share of critical security incidents rose to 19%, the highest since 2015. Also in 2018, a series of large-scale virus infections of technological networks isolated from the Internet (ICS segment) were observed. As a rule, anti-virus databases are updated manually in ICS, so the infection of one network node quickly spreads to other nodes, and the process of neutralizing the cyber threat takes a long time. Therefore, for modern software systems, it is of great importance to solve the problem of preventing computer attacks as a combination of methods of predicting, detecting, and countering attacks. Cyber security problems inherent in developing systems and technologies slow down the pace of development of the country's technological infrastructure.[66]

2.4. RESEARCH INTO THE PROBLEM OF DETECTING COMPUTER ATTACKS ON INDUSTRIAL SYSTEMS

The most important aspect of the problem of detecting computer attacks in PS is the ability of methods and methods to detect all kinds of attacks - the immutability of attacks. The increasing trend in the number of attacks in conjunction with the creation of an increasing number of smart devices (highlighted by the active growth of the microcontroller market) indicates that we should expect and pre-implement new computer attack vectors. predictive security systems.

Numerous publications have been devoted to the problem of detecting computer attacks, including those on industrial systems [67]. Therefore, the study draws attention to the shortcomings of existing methods for securing digital economy platforms. The main issue is the slow speed of

anomaly detection and response. These factors are critical because many attacks are characterized by an "avalanche" of breaches, that is, an increase in the number of failures with an increase in the time elapsed since the start of the attack. This means the need to counter the attack in the first place before counter-action is no longer possible. The efficiency and performance characteristics of any anomaly detection system are highly dependent on the application scheme. This article explores the impact of sensor placement policies and options for creating additional (service) information channels to replace named properties. This approach can be extremely effective given the application principles of modern substations, especially smart power supply networks Smart Grid.

The authors of [68] consider computer attacks to be the most dangerous form of implementing threats to information security in terms of the consequences of their implementation. In terms of the degree of negative impact, computer attacks can occur in government systems, energy and transportation sectors, communication and communication systems, banking sector, etc. poses the greatest threat to The authors especially draw attention to the possible aggravation of the problem of computer diffusion. Attacks for the following reasons:

1) the introduction of information technology in all spheres of life.
2) an increasing trend in the number of targeted computer attacks aimed at breaching the security of certain commercial or government organizations and computer networks.
To solve this problem, the authors of the resource [69] (CII) propose a solution based on its functional representation. The development of a functional representation of an information system is carried out based on an analysis of the space of the parameters of the processes in the system and the definition of parameters characterizing the action of an attack. Therefore, finding all the safe states allows us to identify the signs of a computer attack. System analysis of the process parameter space can be used to better understand the nature of an attack. This approach is undoubtedly promising, but in some cases, it will be ineffective for modern PSs. The main disadvantage of this approach to securing software systems can be computational complexity, as software systems consisting of many objects with limited resources can have a large number of states that can be called secure. In addition, using a large number of parameters will complicate calculations, and with a limited number of parameters, it can sometimes be difficult to distinguish a safe state from an unsafe one.

Source [70] proposes an approach to information sequencing that can be applied when implementing procedures for storing and processing data in the memory of an intelligent cybernetic system to counter computer attacks. In particular, the authors proposed a relational resource network - a memory model that is the basis of the predicted intelligent system for early detection of a computer attack on critical information infrastructure. This model has the ability to synthesize and promptly execute neutralization and response actions in an automatic (automatic) mode in conditions of information conflict.

In the framework of article [71], the authors presented the development of a new method to detect anomalies in the functioning of the Digital Economy's critical information infrastructure. The method is based on the theory of similarity and dimensions. Important features of the method are the accumulation of measures against previously unknown cyber-attacks, the detection of group and mass effects leading to catastrophic conditions, and partial restoration of calculation processes. The approach is based on the principles of the immune system that allow a living organism to resist cyberattacks, malware, and hardware attacks even before the end is complete. This approach is extremely promising, as the characteristics of the immune system, which ensures the ability of a living organism to resist harmful effects, are transferred to Digital Economy systems. However, the authors apply this method to the technological platforms of the Digital Economy, so it is difficult to assess the suitability for industrial, transportation systems, and other critical infrastructures. It is important to note that the authors set the restoration of the computational process as their goal, the main task to ensure the security of the software system is to preserve and timely restore physical processes, the accuracy of which is determined by information. Infrastructure - the environment in which these physical processes are implemented.

The authors of the article [72] also draw attention to the difficulties inherent in the process of securing critical objects of the Digital Economy of the Russian Federation and point to such disadvantages of traditional methods of maintaining cyber resilience. Generally critical infrastructure before deriving important algorithmic features and generalizing certain experimentally generated behavior patterns of the critical infrastructure. Therefore, the authors argue that traditional methods ignore the real conditions for the implementation of computational processes. This leads to simplified ideal results, which’s a real situation may have a significant

error. The article proposes and validates a new method to ensure cyber resilience of the critical information infrastructure of the Russian Federation Digital Economy, based on the use of new models and methods of Industry 4.0 software technology. This method eliminates the disadvantages of traditional methods and makes it possible to provide cyber resilience in the face of the previously unknown group and mass cyber-attacks.

In [73], the authors propose a method for detecting anomalies in the functioning of a distributed computer system. The method checks whether the qualitative conditions are met and uses deterministic algorithms that reduce the level of incorrect answers during testing. The method allows you to detect anomalies that arise both during the processing of data in the stacks of network protocols at user stations, and during the transmission of messages over the network. The qualitative characteristics of detecting abnormal activity do not change for both previously known and new ways of performing unauthorized or unintentional actions. This feature is explained not by modeling various abnormal situations, but by modeling the normal functioning of the data transmission network.

In [74], the authors propose an approach to detect network attacks using a series of detectors. Detectors, one from each of the first training set It is divided into groups trained using sub-sample i. To obtain the final decision from a set of classifiers, a hybrid rule-based on majority voting and a maximum gain voting rule is applied. It should be noted that the test was conducted using the 1998 DARPA test dataset, which does not fully reflect the modern network infrastructure of PS, since then the networks used and the data circulating in them have undergone significant changes. In this regard, this method may require some refinement when integrating with the SS infrastructure.

Some research focuses on detecting certain types of attacks. The most common investigation is to identify Denial of Service (DoS) attacks. Therefore, the authors of the article [75] treat traffic as a time series similar to itself and propose using the properties of such a series to detect anomalies in network behavior. Based on inbound traffic, the Hurst coefficient is calculated, which characterizes the presence or absence of self-similarity in the analyzed time series. Fluctuations in rate indicate a change in traffic behavior and can be used as an indication of the presence of

abnormal outliers. The authors consider applying this method to detect DoS attacks. However, it is also unclear whether this method will be effective in determining other types of attacks and what the effectiveness of this method will be if it is applied to time series that do not have self-similarity at the beginning.

In [76], the authors propose a method for selecting network traffic indicators that most accurately reflect the difference between the normal operation of the system and the operation of the system when a Distributed Denial of Service (DDoS) attack is performed on it. To reduce computational complexity, the authors used two types of metrics: metrics that describe the behavior of a particular packet (size, interval between sending a particular packet, protocol) and metrics that describe the overall traffic behavior (bandwidth, multiple capacities. receiving IP addresses, to a particular IP address). posting frequency). These metrics, along with machine learning algorithms, are used to detect attacks against routers or other intermediate network devices. Using machine learning-based approaches can indeed significantly improve classification accuracy when detecting anomalies, but the study does not say how long the training sample should be and how long the training should take. In addition, while this approach focuses on the characteristics of network traffic, the values of sensors and similar smart devices are also important indicators.

Many studies proposing approaches to detect attacks and anomalies are based on bio-inspired approaches [77]. In their study [78], the authors suggest using the immune detection method to detect malicious programs. Immune detectors are presented as a Kohonen layer. System quality control was carried out on the KDD Cup 99 dataset, which does not reflect the full range of services and applications used in modern networks due to the rapid development of technologies and changes in information infrastructure. As a result, it is not possible to say with confidence that the results obtained in this study apply to ensuring the safety of the aircraft.

Based on the results of the analysis of resources, it can be concluded that for the development of systems (for example, Digital Economy systems), signature-based methods are rarely used to detect computer attacks. Much more common are methods based on probability theory and the application of mathematical statistics, as well as artificial intelligence methods using machine learning algorithms and neural networks as the main tool. There is also a tendency to use bio-

inspired technologies, particularly immune systems; this indicates the need to view the PS and similarly complex systems as a single organism whose vital activity is subject to regular internal and external destructive influences.

2.5. METHODOLOGY OF PREVENTION OF COMPUTER ATTACKS ON MODERN INDUSTRIAL SYSTEMS

An analogy is made between an industrial system and a living organism capable of analyzing and predicting its condition, self-regulating in the event of adverse environmental effects, and at the same time maintaining the constancy of the internal environment. Destructive external and internal influences. To prevent computer attacks, it is proposed to transfer these features of the organism to industrial systems, where the principles of immutability, self-regulation, and cyber-resistance are formulated, which form the basis of the proposed methodology to prevent computer attacks on industrial systems.

He presented a universal graphical model of the functioning of the PS, which allowed to reflect the interaction of system components, the implementation of the PS target function, and the consequences of any computer attack on the system.

3. MATERIAL AND METHODS

3.1. MACHINE LEARNING

Artificial intelligence, a sub-branch of computer science, is the study and development of intelligent machines and software that can learn about objects, gather information, communicate and perceive changes in their environment. Artificial intelligence works with many branches of science such as engineering, mathematics, statistics, and psychology. The basis of artificial intelligence dates back to the mid-1900s. In 1950, A. M. Turing brought a new approach to the world of science with the question of whether machines can think. Since this date, studies have continued on the ability of machines to think. There are several applications of artificial intelligence and machine learning today, from voice recognition to natural language processing to engineering and medical to cyber security. In response to the requirement for robots to be able to learn and adapt to their surroundings, the area of machine learning has grown rapidly. To construct computers capable of learning, machine learning evolved as a sub-branch of AI. In order to produce more accurate forecasts and increase performance, machines use mathematical calculations based on prior experience. An experience is a piece of information gained from previous data, and machines use this information to create it. To employ machine learning, the issue first has to be recognized and defined. Upon completion of this phase, data is collected and analyzed. Converted into a format that machines can read and interpret As a result, training and testing data are segregated. Using the test data, a model's performance is evaluated after it has been trained using the training data. In the unsupervised learning approach, inputs are provided but no outputs are produced in response to them. There are no labels or outcomes included in the data that this approach collects. Learn the structure and distribution of data by studying it in great detail using this approach. The model is not under the direction of a supervisor. Having data that isn't tagged is a problem.

This method is more advantageous because all data are unlabeled in real life. Therefore, it has more possibilities for use in real life. The biggest advantage of this method is that there is no burden of labeling the data. The purpose of unsupervised learning is not to search whether the data belongs to a class or not by classification or prediction. The aim is to search for similarities between the data and to cluster the data. This learning method is generally used for clustering and size

reduction purposes used. In the semi-supervised learning method, a data set consisting of both labeled and unlabeled data is used and predictions are made for all invisible points [62]. This method is used when there is a lot of unlabeled data and less unlabeled data. This method can be used for many problems such as clustering, classification, and estimation. Reinforcement learning is a method in which the model and the environment interact. While the model learns from the environment, the environment is the area where the model learns. In this method, training and testing phases are done together. In the learning phase, the environment is contacted to gather information and in some cases, it can change the environment. The model receives a reward or punishment for each action. The main aim is to guide the model so that it can get more rewards. The model tries to find out which one is best by trying different strategies.

We can explain the difference between artificial intelligence algorithms that do not include machine learning, which we can call "classical" or "traditional" algorithms, and machine learning algorithms with the following simple example: Imagine a human baby who has no knowledge of animals. This baby will not be able to distinguish between cats and dogs. In this thought experiment, let's imagine the baby as an artificial intelligence algorithm and the person who will teach him the differences as a computer scientist [79].

In classical artificial intelligence applications, instead of teaching the baby the difference between cat and dog, we give him a full set of instructions that will enable him to distinguish between cat and dog. These instructions should be prepared in such a way that this ignorant baby should be able to both follow those directions and distinguish between cats and dogs mostly correctly. It is obvious how difficult this is for both the baby and the teacher.

On the other hand, as you may have noticed, we don't need to do this to increase baby's success: If we give the baby lots of pictures of cats and dogs and repeatedly tell which is which, the baby will at some point begin to distinguish the two with high accuracy. This is exactly how machine learning algorithms work, as the name suggests. Of course, computers, unfortunately, do not come with a learning algorithm in them when they are built. Computer scientists themselves design and adapt the most suitable machine learning algorithm for each problem. Machine learning is essentially built around this logic. Many artificial intelligence applications that we have seen recently are machine learning applications. Machine learning is indispensable in real life for most

artificial intelligence applications that we are used to seeing in science fiction movies; therefore, the concepts of "machine learning" and "artificial intelligence" are getting more confused [80].

At this point, although we have largely avoided the conceptual confusion by revealing the difference between the concepts of machine learning and artificial intelligence, there is one more step we need to take: Although machine learning is a narrower concept than artificial intelligence, it still includes many different algorithms and it is a method field. That's why the concept of "machine learning" still doesn't adequately describe the products of popularization we're talking about. Therefore, now, by talking about deep learning, which is a sub-branch of machine learning, we will both avoid popular broadcast-level conceptual confusion and better understand why artificial intelligence has been so popular for the last 10 years. Integrating the ability to think like a human being within the possibilities of technology to computers, which makes life easier in many areas, has provided machine learning. ML enhances performance by gaining knowledge from past mistakes. It's a promising topic that might help streamline operations and allocate resources more effectively. Supervised learning, unsupervised classification, and recurrent neural networks are the three subfields of machine learning that make up the majority of the field's work. Labeled data is needed to train supervised learning, which contains inputs and intended outputs. In contrast to supervised learning, unsupervised learning needs no training data and only inputs with no goal result are having to learn. Reinforcement learning is a kind of learning that is based on feedback from interactions with the outside world.

3.2.DEEP LEARNING

Artificial intelligence is a broad field of study that deals with the methods by which human intelligence can be simulated by a computer. Basic topics of artificial intelligence, knowledge representation (Ing. knowledge representation), inference (Ing. inference), and learning (Eng. learning). Artificial intelligence subsets including machine learning and deep learning are shown in Figure 3.1. Deep learning is the sub-branch of artificial intelligence in learning and is the most has widespread use. Instead of a machine making decisions according to predetermined rules, it learns to use large data sets and aims to make better inferences.

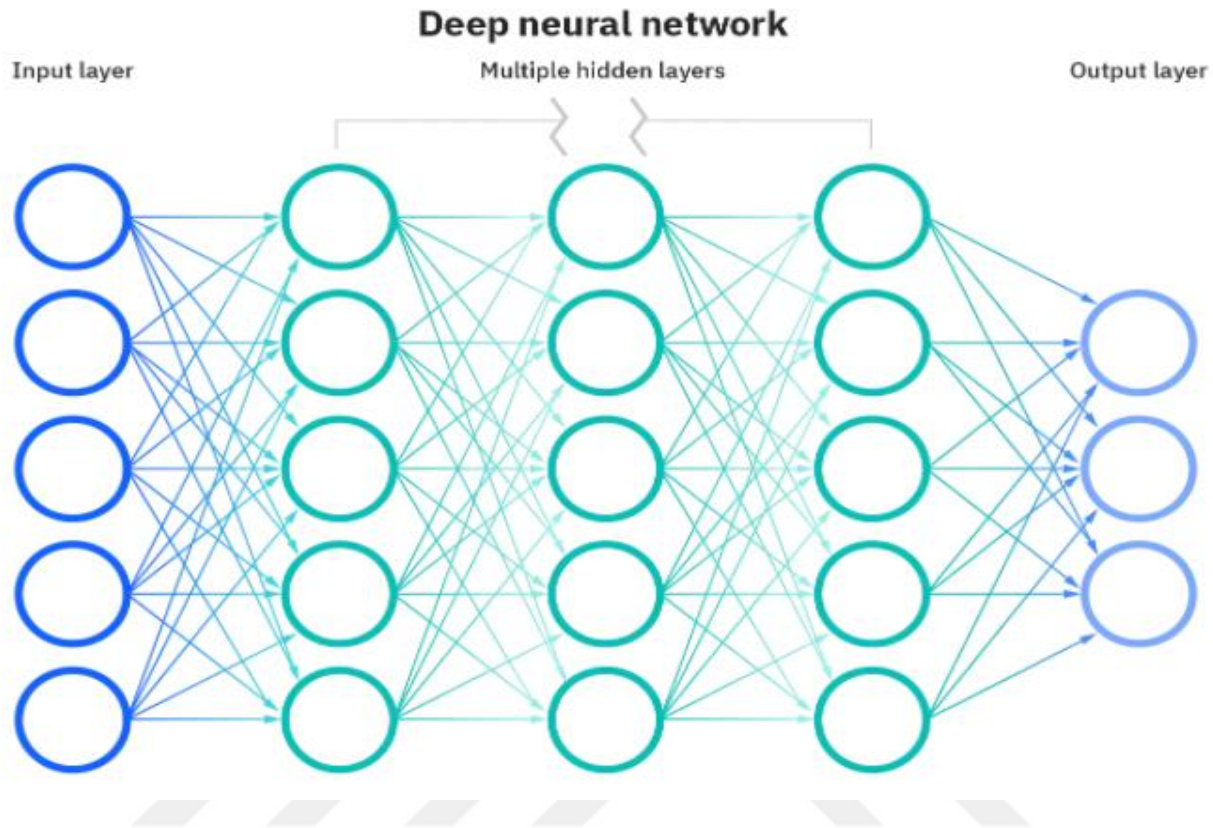

Figure 3.1: Deep Neural network [81]

Machine learning is used in every aspect of our daily lives, even if some of us are not aware of it. Object recognition, speech-to-text software, electronic commerce sites that make product recommendations based on the interests of users and the products they have viewed in the past are the most basic examples of these areas. Machine learning is divided into three controlled learning, uncontrolled learning, and reinforcement learning, as can be seen in Figure 3.2, and its usage areas are listed. In the supervised learning method, input is given to the algorithm and the expected result is given. The result obtained as a result of the algorithm and the expected result is compared by the algorithm and the calculation parameters are recalculated. Unsupervised learning is a method that uses sets of information given as input, without the expected output being given to the algorithm. This method is especially used in clustering big data. Reinforced learning, on the other hand, has a reward mechanism. The algorithm learns which parameters to use to reach the highest reward. Deep learning aims to learn the relationship between inputs and outputs [82]. For this purpose, deep learning approaches provide the representation of deep learning models consisting of multiple layers and data with various qualities.

Speech recognition, object identification, and object detection have all benefited from the advent of deep learning algorithms. Machine learning is made easier with bigger datasets. As an example, the complexity of a job may be influenced by the quality of labels used to categorize datasets. In order to implement machine learning, programmers must create algorithms that provide precise predictions (Mohri, Rostamizadeh & Dikmen). Machine learning is the most often employed approach in research looking at early diagnosis in the medical industry. Hypotheses in the realm of health are used as a basis for applications [83].

3.2.1. Recurrent Neural Network (RNN)

The short-term memory of a strict feedforward structural business is not maintained. The network's re-presentation of past inputs is responsible for any remembrance properties. Short-term memory is shown by activation feedback in a simple RNN. The efficiency of a national layer is not limited to the external input of the network, but also to the engagement after the previous forward propagation of data. There are weights assigned to the feedback, which enables unintentional learning (such as backpropagation) [84].

Learning may be achieved by using the same gradient descent events as those used for feed-forward networks, which have a simple structure and deterministic activation functions. Pretend hardening approaches may also be appropriate when installations are stochastic. The following will display at a low level of RNN's most important types and features. This kind of artificial neural network (RNN) has a directed loop of connections between nodes. As a result, it's able to demonstrate rapid changes in time. Like feed - forward networks, RNNs can utilise their input memory for processing any sequence of inputs, unlike feedforward networks that can only process one input at a time. Figure 3.2 shows a typical RNN.

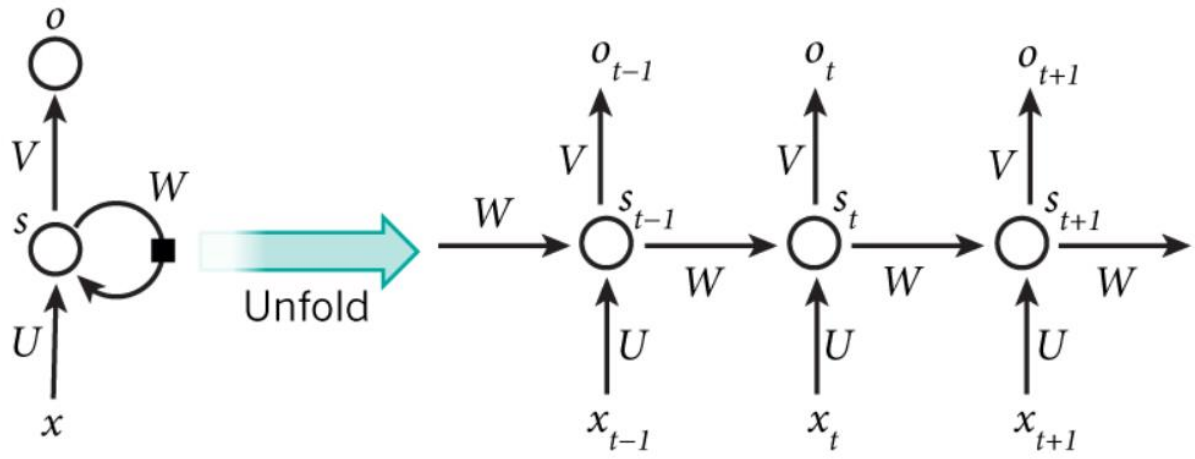

Figure 3.2: RNN Structure [85].

Figure 3.3 above shows an RNN unpacking into a full network.

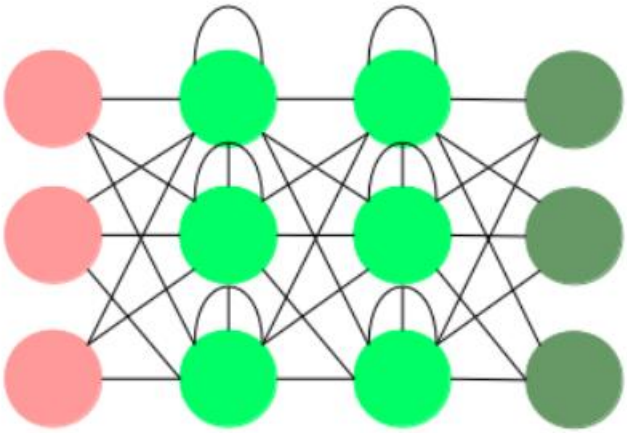

Figure 3.3: RNN Process 1 [86].

Based on the premise that production at time t may be dependent on future items as well as on past items in queues, bidirectional RNNs (Bidirectional RNN) are used. If you're trying to figure out where a word is in a sequence, for example, you'll want to look at both the left and right content. Bidirectional RNNs are rather simple to build. This is nothing more than a pair of RNNs layered on top of one another. After that, the results are computed using the most recent states of both RNNs.

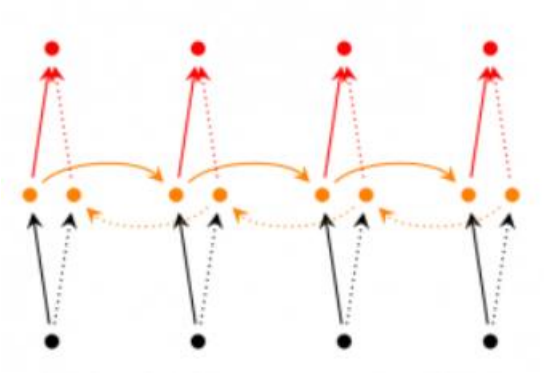

Figure 3.4: RNN Process 2 [87].

Similar to Bidirectional RNNs, Deep RNNs also have numerous layers, but only in the time step. As a result, we are more equipped to take in new information (but also needs a lot of training data).

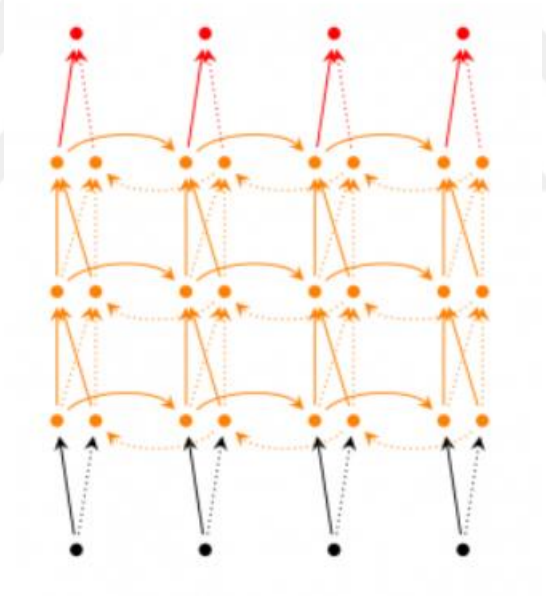

Figure 3.5: RNN Process 3 [88].

3.3. PROPOSED METHOD

In this thesis new framework-based, AI is presented for network traffic classification. In the first stage, the dataset was obtained from Kaggle and preprocessed. Then, the dataset is wired to the second stage which the dataset is divided by the model to the train and test automatically. In the next stage, the KNN is trained by the training section. Then, after completing the training stage the model is saved with optimum parameters by using the grid search algorithm. The selected best

parameters of the KNN are used in the testing section. In the testing section, the model is used to predicate the labels of the model as shown in Figure 3.6.

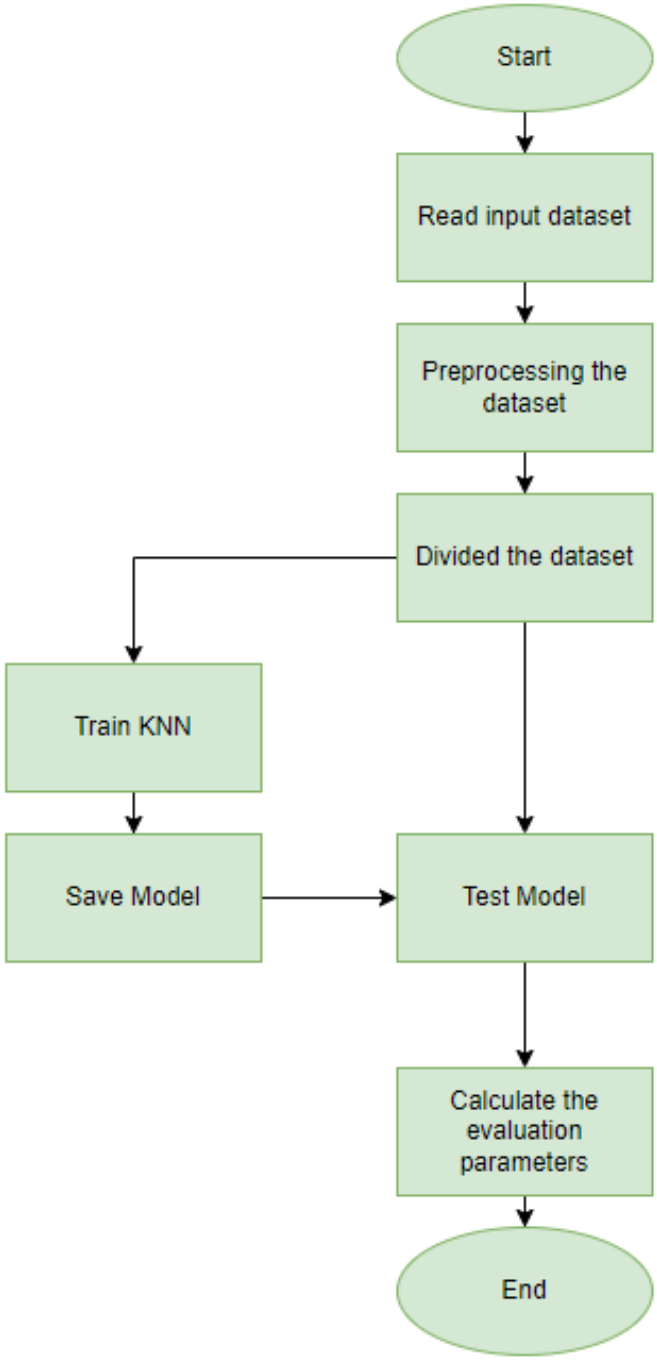

Figure 3.6: Proposed Hybrid model.

4. EXPERIMENTS AND DISCUSSIONS

4.1. RESULTS

4.2. DISCUSSIONS AND COMPARISONS

Here, the findings of our technique are shown and reviewed in both accuracy and processing time for 2 techniques that are used for person identification using finger vein pictures. Equation (Eq.) shows how we arrived at the primary evaluation variable for our approach (4.1)

$$Acc = \frac{TP + TN}{Total} \tag{4.1}$$

TP indicated the genuine positive; TN the true negative; and overall its total estimate, and the findings are provided in Table 4.1. Table 4.1: ACC, TP, TN, and Total

Table 4.1: Comparisons between Accuracies of Two Methods

Testing Ratio (%)	KNN based Grid Search (%)	KNN (%)
Test (20%)	99.34	96.12
Test (30%)	98.33	93.21
Test (40%)	96.63	90.09

The testing findings demonstrate that the KNN alone produced the lowest results KNN based Grid Search in all test instances. All findings are provided in Figure 4.1 to prove the acquired outcomes too. Figure 4.2 depicted the accuracies comparison of two procedures done in this investigation.

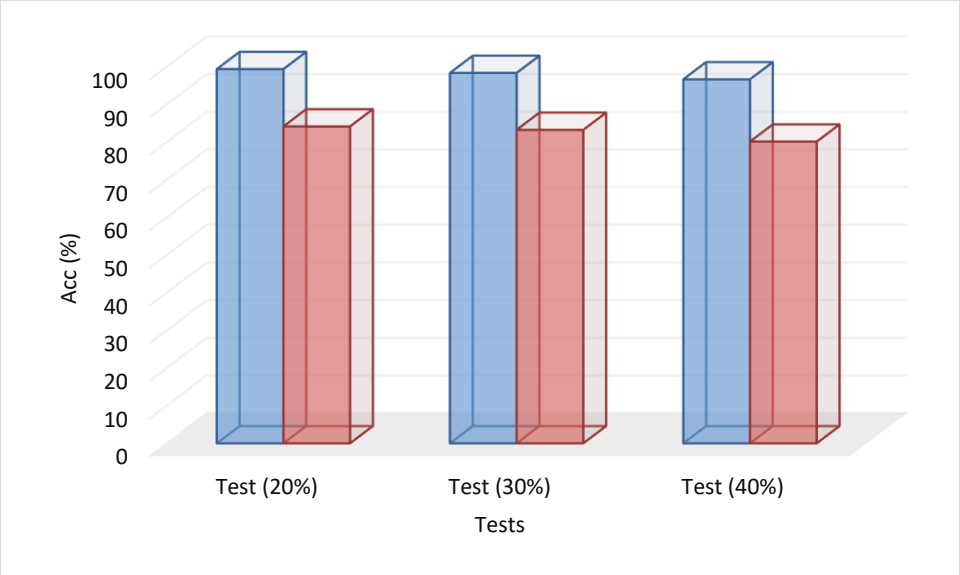

Figure 4.1: Comparisons between Two Classifiers

On the other hand, execution time was also compared between KNN and Grid Search in Table 4.2.

Table 4.2: Comparisons between Execution Time of Two Methods

Testing Ratio (%)	KNN and Grid Search (sec)	KNN (sec)
Test (20%)	2104	2309
Test (30%)	1972	2287
Test (40%)	1834	1982

By investigating Table 4.2, the KNN presented low performance than KNN and Grid Search. For all validation cases 40%, 30%, and 20% KNN and Grid Search presented best execution times then KNN sees Figure 4.3.

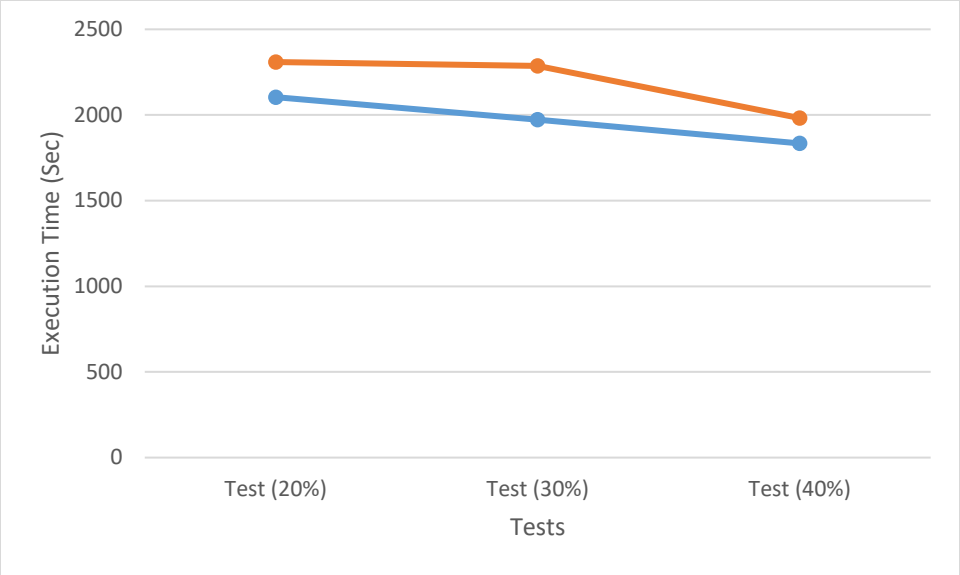

Figure 4.2: Compression of Execution Times

Furthermore, the precision of the model is also calculated to be important in order to ensure we are getting the correct results. The precision is calculated as shown below in equation 4.2.

$$precision = \frac{TP}{tp + fp} \tag{4.2}$$

Then, the results of the proposed method precision are presented in Table 4.3.

Table 4.3: Precision Results in Comparisons between Two Methods

Testing Ratio (%)	KNN and Grid Search (%)	KNN (%)
Test (20%)	97.45	94.43
Test (30%)	96.34	93.34
Test (40%)	95.93	91.64

Table 4.3 show that the proposed method presented results vary between 97.45% and 95.93 which are highest than DBN alone presented results between 75.43% and 70.64%. Precision is the classification model's ability to identify only relevant data points. Mathematically, precision is the

number of true positives divided by the number of true positives plus the number of false positives. Figure 4.3 represented the precision values of both techniques.

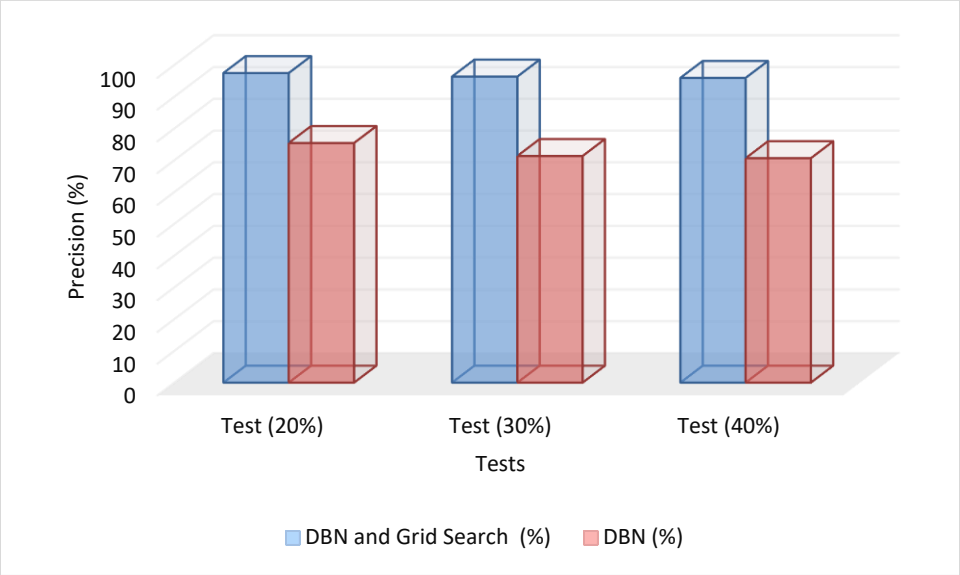

Figure 4.3: Compression of Precision Results of Two Techniques

On the other hand, the sensitivity of the model was calculated to evaluate the results. Sensitivity (True Positive Rate) indicates the percentage of those who have the disorder (when evaluated by the 'Gold Standard') that obtained a positive outcome on this test.

Table 4.4: Sensitivity Results in Comparisons between Two Methods

Testing Ratio (%)	KNN and Grid Search (%)	KNN (%)
Test (20%)	98.54	93.43
Test (30%)	96.54	92.54
Test (40%)	95.31	92.06

Furthermore, Figure 4.4 represented the comparison between the proposed method and KNN. By analysing the figure, we can calculate that the proposed method presented the best Sensitivity results than KNN.

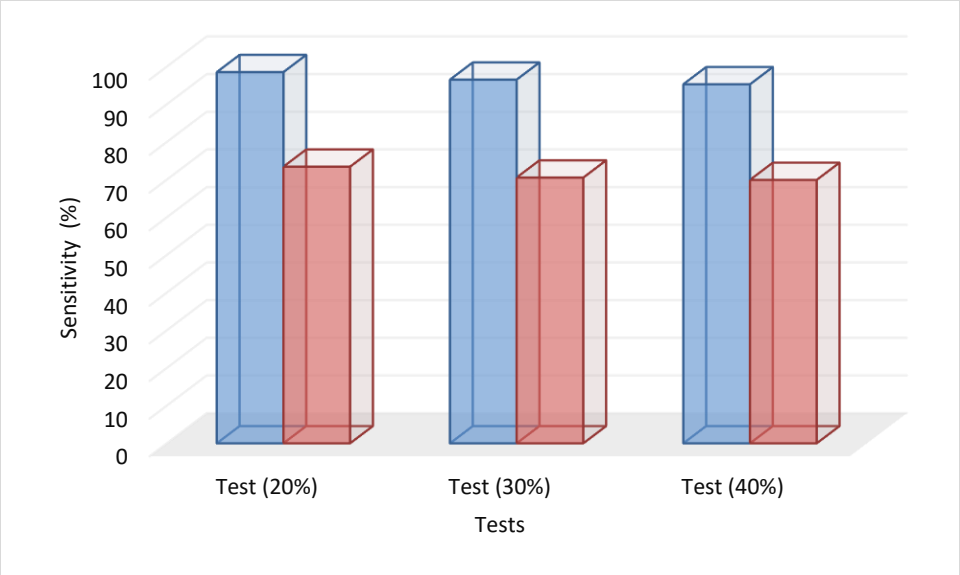

Figure 4.4: Compression of Precision Results of Two Techniques

Table 4.5: Comparison of our method with previous studies

Methods	Methods	ACC (%)
[93]	COS-P-Poly3	88.84
[94]	Logistic Regression	86.6
[95]	Decision Tree	96.7
[89]	Linear SVM	93.1
[90]	Ensemble learning	95.1
[91]	fairly adequate+ classifier	89
[92]	Markov chain + Random forest	92
Our Method	KNN based grid search	99.00

In [89] Alzaylaee et. L. Proposed another study that used real phones for automatic feature extraction to overcome the limitations of dynamic analysis of certain malware in Android. Dynamic attributes are automatically taken from sample apps running on Android phones. In addition, some of the tests performed benchmarking of malware detection methods using simulators and devices. This study by Alzaylaee et al. is particularly important because most studies based on dynamic and behavioral analysis use a simulator. We used 222 malware samples

from the MalGenome project and 1222 innocent samples from McAfee Labs associated with Intel Security. Furthermore, in another recent study supported by artificial learning, the source codes of 200 malicious and 200 harmless Android applications were analyzed and an F1 score of 95.1% was obtained [90]. In another study using real-time machine learning techniques on the device [91], researchers used reports from the Cuckoo Sandbox file analysis platform to extract behavioral attributes from Android malware. They obtained the malware samples from the malware sharing platform called Virus Share and tested the machine learning algorithm on the device they developed on a total of 2000 samples belonging to 18 different malware families. The accuracy rate obtained using the 10-fold cross-validation method (10-fold-cross-validation) was 89% accuracy. In a more recent study published by Onwuzurike et al. [92], using a modeling approach similar to MAMADROID, a structure that combines static, dynamic and hybrid analysis methods is proposed. The researchers emphasized that the combination of static and dynamic analysis techniques gave the best results and they achieved an F1 score of 92%, in addition, static analysis techniques were at least as effective as dynamic analysis techniques.

Then, our method presented 99.0 which is remarkable when compared with previous studies. The main issue in our method we combine KNN with Grid search to obtain the best accuracy. The one the using KNN alone lead to low accuracies in some time when compared with optimized KNN.

5. CONCLUSION

Although network traffic classification has many advantages, they have not been adequately addressed in terms of information security threats. For this reason, it is of great importance to evaluate network traffic classification in terms of information security threats. In this study, the vulnerabilities of network traffic classification against information security threats are compared with traditional computer networks and solutions are presented.

In simulation studies using Hping3 and Dsniff applications, IP Spoofing, SYN Flood, RST/FIN Flood, SYN-ACK Flood, UDP Flood, ARP Poisoning, and DDoS attacks were carried out. The results of the simulation studies show that network traffic classification has more security vulnerabilities compared to traditional computer networks.

On the other hand, security vulnerabilities in network traffic classification can be eliminated by taking measures such as data rate limitation for the network, packet sending limitation, authentication techniques, and the installation of firewalls and intrusion detection and prevention systems in order to prevent attacks. This study has several advantages.

A. simplify and cheapen network equipment,

B. facilitate network management,

C. The ability to quickly change the behaviour of network equipment regardless of the manufacturer, to assist in the research and testing of new mechanisms.

In future works, we advise the researchers to apply various machine learning techniques in developing network traffic classification methods. Furthermore, applying unsupervised learning to various security problems leads to finding which of these techniques are suitable for computer security problems.

REFERENCES

[1] V. Mitra, H. Franco, M. Graciarena and D. Vergyri, "Medium-duration modulation cepstral feature for robust speech recognition," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Florence,, pp. 1749-1753, doi: 10.1109/ICASSP.2014.6853898.2014.

[2] O. Hazrati, S. Ghaffarzadegan and J. H. L. Hansen, "Leveraging automatic speech recognition in cochlear implants for improved speech intelligibility under reverberation," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brisbane, QLDpp. 5093-5097, doi: 10.1109/ICASSP.2015.7178941.2015.

[3] R. Lotfidereshgi and P. Gournay, "Biologically inspired speech emotion recognition," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, New Orleans, LA, 27, pp. 5135-5139, doi: 10.1109/ICASSP.2017.7953135. 2017.

[4] Y. Suh *et al.*, "Development of distant multi-channel speech and noise databases for speech recognition by in-door conversational robots," *20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)*, Seoul, , pp. 1-4, doi: 10.1109/ICSDA.2017.8384419.2017.

[5] P. S. Nidadavolu, V. Iglesias, J. Villalba and N. Dehak, "Investigation on Neural Bandwidth Extension of Telephone Speech for Improved Speaker Recognition," *ICASSP IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, , pp. 6111-6115, doi: 10.1109/ICASSP.8682992.2019.

[6] L. Li, W. Xu, J. Wu, S. He and X. Li, "The Hokkien isolated word recognition system based on FPGA," *International Conference on Anti-Counterfeiting, Security and Identification (ASID)*, Macao, , pp. 1-5, doi: 10.1109/ICASID.2014.7064971.2014.

[7] W. Shu-Guang, Z. Xiang-Yang and W. Qiang, "Isolated word recognition in reverberant environments," *IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, Xi'an, , pp. 1-4, doi: 10.1109/ICSPCC.2011.6061575.2011.

[8] S. Sawant and M. Deshpande, "Isolated Spoken Marathi Words Recognition Using HMM," *8 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, Pune, India, , pp. 1-4, doi: 10.1109/ICCUBEA.2018.8697457.2018.

[9] F. Ang, H. Tsutsui and Y. Miyanaga, "Incorporation of time-varying LP cepstral features in HMM-based isolated word speech recognition," *International Symposium on Signals, Circuits and Systems (ISSCS)*, Iasi, , pp. 1-4, doi: 10.1109/ISSCS.2015.7204030.2015.

[10] S. C. Sajjan and C. Vijaya, "Comparison of DTW and HMM for isolated word recognition," *International Conference on Pattern Recognition, Informatics and Medical Engineering*, Salem, Tamilnadu, , pp. 466-470, doi: 10.1109/ICPRIME.2012.6208391.2012.

[11] S. Masood, M. Mehta, Namrata and D. R. Rizvi, "Isolated word recognition using neural network," *Annual IEEE India Conference (INDICON)*, New Delhi, , pp. 1-5, doi: 10.1109/INDICON.2015.7443697.2015.

[12] M. Sher, N. Ahmad and M. Sher, "TESPAR feature based isolated word speaker recognition system," *18th International Conference on Automation and Computing (ICAC)*, Loughborough, , pp. 1-4.2012.

[13] C. Wei and Y. Yang, "Mandarin isolated words recognition method based on pitch contour," *IEEE 2nd International Conference on Cloud Computing and Intelligence Systems*, Hangzhou, , pp. 143-147, doi: 10.1109/CCIS.2012.6664385.2012.

[14] G. Wang, Y. Zhang, M. Sun, X. Wang and Y. Zhang, "Speech signal feature parameters extraction algorithm based on PCNN for isolated word recognition," *International Conference on Audio, Language and Image Processing (ICALIP)*, Shanghai, , pp. 679-682, doi: 10.1109/ICALIP.2016.7846618.2016.

[15] M. Raczynski, "Speech processing algorithm for isolated words recognition," *International Interdisciplinary PhD Workshop (IIPhDW)*, Swinoujście, , pp. 27-31, doi: 10.1109/IIPHDW.2018.8388238.2018.

[16] Preeti Saini, Parneet Kaur, Mohit Dua, “Hindi Automatic Speech Recognition Using HTK,” International Journal of Engineering Trends and Technology (IJETT) – Volume4 Issue6- June 2013.

[17] Gaurav, Devanesamoni Shakina Devi, Gopal Krishna Sharma, Mahua Bhattacharya, “Development of Application Specific Continuous Speech Recognition System in Hindi”, Journal of Signal and Information Processing, 2012, 3, 394-401.

[18] M. K. Linga Murthy, G.L.N. Murthy,"Isolated Word Recognition Using LPC & Vector Quantization", International Journal of Research in Engineering and Technology (IJRET)-Volume1 Issue3- Nov2012.

[19] L. Xu and M. Ke, "Research on isolated word recognition with DTW-based," *International Conference on Computer Science & Education (ICCSE)*, Melbourne, VIC, pp. 139-141, doi: 10.1109/ICCSE.2012.6295044.2012.

[20] Hall T, Beecham S, Bowes D, Gray D, Counsell S. “A systematic literature review on fault prediction performance in software engineering”. IEEE Transactions on Software Engineering, 38(6), 1276-1304, 2012.

[21] Catal C, Diri B. “A systematic review of software fault prediction studies”. Expert Systems with Applications, 36(4), 7346-7354, 2009.

[22] Pal B, Hasan A, Aktar M, Shahdat N. “Cluster ensemble and probabilistic neural network modeling of class ımbalance learning in software defect prediction”. Artificial Intelligence and Applications, In Press.

[23] Shirabad S, Menzies TJ. School of Information Technology and Engineering, University of Ottawa. “The PROMISE repository of software engineering databases”. http://promise.site.uottawa.ca/SERepository (01.10.2017).

[24] Koru A, Liu H. “Building effective defect-prediction models in practice”. IEEE Software, 22(6), 23-29, 2005.

[25] Pelayo L, Dick S. “Applying novel resampling strategies to software defect prediction”. Fuzzy Information Processing Society NAFIPS ’07, San Diego, USA, 24-27 June, 2007.

[26] Frank E, Witten IH. “Generating accurate rule sets without global optimization”. 15th International Conference on Machine Learning, Wisconsin, USA, 24-27 July 1998.

[27] Menzies T, Greenwald J, Frank A. “Data mining static code attributes to learn defect predictors”. IEEE Transactions on Software Engineering, 33(1), 2-13, 2007.

[28] Lessmann S, Baesens B, Mues C, Pietsch S. “Benchmarking classification models for software defect prediction: A proposed framework and novel findings”. IEEE Transactions on Software Engineering, 34(4), 485-496, 2008.

[29] Japkowicz N, Stephen S. “The class imbalance problem: A systematic study”. Intelligent Data Analysis, 6(5), 429-449, 2002.

[30] Batista G, Prati R, Monard M, “A Study of the Behavior of several methods for balancing machine learning training data”. ACM SIGKDD Explorations Special issue on learning from imbalanced datasets, 6(1), 20-29, 2004.

[31] He H, Bai Y, Garcia EA, Li S. “ADASYN: Adaptive synthetic sampling approach for imbalanced learning”. IEEE International Joint Conference on Neural Networks, Hong Kong, China, 1-8 June 2008.

[32] Xiangwen Wang, Xianghong Lin, Xiaochao Dang, Supervised learning in spiking neural networks: A review of algorithms and evaluations, Neural Networks, Volume 125, Pages 258-280, ISSN 0893-6080.2020.

[33] Khalil Moshkbar-Bakhshayesh, Performance study of bayesian regularization based multilayer feed-forward neural network for estimation of the uranium price in comparison with the different supervised learning algorithms, Progress in Nuclear Energy, Volume 127, 103439, ISSN 0149-1970.2020.

[34] Jiabin Zhang, Hu Su, Wei Zou, Xinyi Gong, Zhengtao Zhang, Fei Shen, CADN: A weakly supervised learning-based category-aware object detection network for surface defect detection, Pattern Recognition, Volume 109, 107571, ISSN 0031-3203.2021.

[35]] Lu Han, Wenjun Li, Zhi Su, An assertive reasoning method for emergency response management based on knowledge elements C4.5 decision tree, Expert Systems with

Applications, Volume 122, Pages 65-74, ISSN 0957-4174.2019.

[36] X. Wang, C. Zhou, X. Xu, Application of C4.5 decision tree for scholarship evaluations, Procedia Computer Science, Volume 151, Pages 179-184, ISSN 1877-0509.2019.

[37] Sunanda Gamage, Jagath Samarabandu, Deep learning methods in network intrusion detection: A survey and an objective comparison, Journal of Network and Computer Applications, Volume 169, 102767, ISSN 1084-8045.2020.

[38] Fatsuma Jauro, Haruna Chiroma, Abdulsalam Y. Gital, Mubarak Almutairi, Shafi'i M. Abdulhamid, Jemal H. Abawajy, Deep learning architectures in emerging cloud computing architectures: Recent development, challenges and next research trend, Applied Soft Computing, Volume 96, 106582, ISSN 1568-4946.2020.

[39] Y. Li, B. Sixou, F. Peyrin, A review of the deep learning methods for medical images super resolution problems, IRBM, ISSN 1959-0318.2020.

[40] Xiyun Yang, Yanfeng Zhang, Wei Lv, Dong Wang, Image recognition of wind turbine blade damage based on a deep learning model with transfer learning and an ensemble learning classifier, Renewable Energy, ISSN 0960-1481.2020.

[41] Shaolong Sun, Shouyang Wang, Yunjie Wei, A new ensemble deep learning approach for exchange rates forecasting and trading, Advanced Engineering Informatics, Volume 46, 101160, ISSN 1474-0346.2020.

[42] S. Hernández, Juan L. López, Uncertainty quantification for plant disease detection using Bayesian deep learning, Applied Soft Computing, Volume 96, 106597, ISSN 1568-4946.2020.

[43] Hui Yin, Yuanhao Gong, Guoping Qiu, Fast and efficient implementation of image filtering using a side window convolutional neural network, Signal Processing, Volume 176, 107717, ISSN 0165-1684.2020.

[44] Samira Zare, Moosa Ayati, Simultaneous fault diagnosis of wind turbine using multichannel convolutional neural networks, ISA Transactions, ISSN 0019-0578.2020.

[45] Qi Li, Pengfei Li, Kezhi Mao, Edmond Yat-Man Lo, Improving convolutional neural network for text classification by recursive data pruning, Neurocomputing, Volume 414, Pages 143-152, ISSN 0925-2312.2020.

[46] JongCheol Pyo, Lan Joo Park, Yakov Pachepsky, Sang-Soo Baek, Kyunghyun Kim, Kyung Hwa Cho, Using convolutional neural network for predicting cyanobacteria concentrations in river water, Water Research, Volume 186, 116349, ISSN 0043-1354.2020.

[47] He Li, Xiaohui Li, Feng Yuan, Simon M. Jowitt, Mingming Zhang, Jie Zhou, Taofa Zhou, Xiangling Li, Can Ge, Bangcai Wu, Convolutional neural network and transfer learning based mineral prospectivity modeling for geochemical exploration of Au mineralization within the Guandian–Zhangbaling area, Anhui Province, China, Applied Geochemistry, 104747, ISSN 0883-2927.2020.

[48] Marcus Stoffel, Franz Bamer, Bernd Markert, Deep convolutional neural networks in structural dynamics under consideration of viscoplastic material behaviour, Mechanics Research Communications, Volume 108, 103565, ISSN 0093-6413.2020.

[49] Amin Khatami, Asef Nazari, Abbas Khosravi, Chee Peng Lim, Saeid Nahavandi, A weight perturbation-based regularisation technique for convolutional neural networks and the application in medical imaging, Expert Systems with Applications, Volume 149, 113196, ISSN 0957-4174,.2020.

[50] Kaiyu Zhang, Jinglong Chen, Tianci Zhang, Zitong Zhou, A Compact Convolutional Neural Network Augmented with Multiscale Feature Extraction of Acquired Monitoring Data for Mechanical Intelligent Fault Diagnosis, Journal of Manufacturing Systems, Volume 55, Pages 273-284, ISSN 0278-6125.2020.

[51] Jialin Tang, Qinglang Su, Binghua Su, Simon Fong, Wei Cao, Xueyuan Gong, Parallel ensemble learning of convolutional neural networks and local binary patterns for face recognition, Computer Methods and Programs in Biomedicine, Volume 197, 105622, ISSN 0169-2607.2020.

[52] Zhi-jun Lu, Qi Qin, Hong-yin Shi, Hao Huang, SAR moving target imaging based on convolutional neural network, Digital Signal Processing, Volume 106, 102832, ISSN 1051-

2004,.

[53] M.V. Valueva, N.N. Nagornov, P.A. Lyakhov, G.V. Valuev, N.I. Chervyakov, Application of the residue number system to reduce hardware costs of the convolutional neural network implementation, Mathematics and Computers in Simulation, Volume 177, Pages 232-243, ISSN 0378-4754.2020.

[54] Zhiying Fang, Han Feng, Shuo Huang, Ding-Xuan Zhou, Theory of deep convolutional neural networks II: Spherical analysis, Neural Networks, Volume 131, Pages 154-162, ISSN 0893-6080.2020.

[55] Rachna Jain, Preeti Nagrath, Gaurav Kataria, V. Sirish Kaushik, D. Jude Hemanth, Pneumonia detection in chest X-ray images using convolutional neural networks and transfer learning, Measurement, Volume 165, 108046, ISSN 0263-2241.2020.

[56] Priyadarshiny Dhar, Saibal Dutta, Vivekananda Mukherjee, Cross-wavelet assisted convolution neural network (AlexNet) approach for phonocardiogram signals classification, Biomedical Signal Processing and Control, Volume 63, 102142, ISSN 1746-8094.2021.

[57] Xuliang Liu, Daolun Li, Jinghai Yang, Wenshu Zha, Ziqi Zhou, Liping Gao, Jiahang Han, Automatic well test interpretation based on convolutional neural network for infinite reservoir, Journal of Petroleum Science and Engineering, Volume 195, 107618, ISSN 0920-4105.2020.

[58] Adil Moghar, Mhamed Hamiche, Stock Market Prediction Using LSTM Recurrent Neural Network, Procedia Computer Science, Volume 170, Pages 1168-1173, ISSN 1877-0509.2020.

[59] Refat Khan Pathan, Munmun Biswas, Mayeen Uddin Khandaker, Time series prediction of COVID-19 by mutation rate analysis using recurrent neural network-based LSTM model, Chaos, Solitons & Fractals, Volume 138, 110018, ISSN 0960-0779.2020.

[60] Alex Sherstinsky, Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network, Physica D: Nonlinear Phenomena, Volume 404, 132306, ISSN 0167-2789.2020.

[61] Pritika Bahad, Preeti Saxena, Raj Kamal, Fake News Detection using Bi-directional LSTM-Recurrent Neural Network, Procedia Computer Science, Volume 165, Pages 74-82, ISSN 1877-0509.2019.

[62] Zhiyong Cui, Ruimin Ke, Ziyuan Pu, Yinhai Wang, Stacked bidirectional and unidirectional LSTM recurrent neural network for forecasting network-wide traffic state with missing values, Transportation Research Part C: Emerging Technologies, Volume 118, 102674, ISSN 0968-090X.2020.

[63] Ying Chen, Voltages prediction algorithm based on LSTM recurrent neural network, Optik, Volume 220, 164869, ISSN 0030-4026.2020.

[64] Neculai Andrei, A Dai–Yuan conjugate gradient algorithm with sufficient descent and conjugacy conditions for unconstrained optimization, Applied Mathematics Letters, Volume 21, Issue 2, Pages 165-171, ISSN 0893-9659.2008.

[65] Muhammed Maruf Öztürk, İbrahim Arda Cankaya, Deniz İpekçi, Optimizing echo state network through a novel fisher maximization based stochastic gradient descent, Neurocomputing, Volume 415, Pages 215-224, ISSN 0925-2312.2020

[66] Andreea Koreanschi, Oliviu Sugar Gabor, Joran Acotto, Guillaume Brianchon, Gregoire Portier, Ruxandra Mihaela Botez, Mahmoud Mamou, Youssef Mebarki, Optimization and design of an aircraft's morphing wing-tip demonstrator for drag reduction at low speed, Part I – Aerodynamic optimization using genetic, bee colony and gradient descent algorithms, Chinese Journal of Aeronautics, Volume 30, Issue 1, Pages 149-163, ISSN 1000-9361.2017.

[67] Arnulf Jentzen, Philippe von Wurstemberger, Lower error bounds for the stochastic gradient descent optimization algorithm: Sharp convergence rates for slowly and fast decaying learning rates, Journal of Complexity, Volume 57, 101438, ISSN 0885-064X, https://doi.org/10.1016/j.jco.2019.101438..2020.

[68] Xuemei Dong, Ding-Xuan Zhou, Learning gradients by a gradient descent algorithm, Journal of Mathematical Analysis and Applications, Volume 341, Issue 2, Pages 1018-1027, ISSN 0022-247X, https://doi.org/10.1016/j.jmaa.2007.10.044..2008.

[69] Rui Ye, Qun Dai, Implementing transfer learning across different datasets for time series forecasting, Pattern Recognition, Volume 109, 107617, ISSN 0031-3203,.2021.

[70] Yu Liu, Ao Li, Xing-Ming Zhao, Minghui Wang, DeepTL-Ubi: A novel deep transfer learning method for effectively predicting ubiquitination sites of multiple species, Methods, ISSN 1046-2023,..2020.

[71] Jingyao Wu, Zhibin Zhao, Chuang Sun, Ruqiang Yan, Xuefeng Chen, Few-shot transfer learning for intelligent fault diagnosis of machine, Measurement, Volume 166, 108202, ISSN 0263-2241,..2020.

[72] Shanshan Wang, Lei Zhang, Jingru Fu, Adversarial transfer learning for cross-domain visual recognition, Knowledge-Based Systems, Volume 204, 106258, ISSN 0950-7051,..2020

[73] Xiyun Yang, Yanfeng Zhang, Wei Lv, Dong Wang, Image recognition of wind turbine blade damage based on a deep learning model with transfer learning and an ensemble learning classifier, Renewable Energy, ISSN 0960-1481,..2020

[74] Chuan Li, Shaohui Zhang, Yi Qin, Edgar Estupinan, A systematic review of deep transfer learning for machinery fault diagnosis, Neurocomputing, Volume 407, Pages 121-135, ISSN 0925-2312,..2020.

[75] Yue Wang, Yuting Liu, Wei Chen, Zhi-Ming Ma, Tie-Yan Liu, Target transfer Q-learning and its convergence analysis, Neurocomputing, Volume 392, Pages 11-22, ISSN 0925-2312,..2020.

[76] Zhengjun Qiu, Shutao Zhao, Xuping Feng, Yong He, Transfer learning method for plastic pollution evaluation in soil using NIR sensor, Science of The Total Environment, Volume 740, 140118, ISSN 0048-9697,..2020.

[77] Santi Kumari Behera, Amiya Kumar Rath, Prabira Kumar Sethy, Maturity status classification of papaya fruits based on machine learning and transfer learning approach, Information Processing in Agriculture, ISSN 2214-3173,..2020.

[78] Seunghyeon Kim, Yung-Kyun Noh, Frank C. Park, Efficient neural network compression

via transfer learning for machine vision inspection, Neurocomputing, Volume 413, Pages 294-304, ISSN 0925-2312,..2020.

[79] Xiang Li, Wei Zhang, Hui Ma, Zhong Luo, Xu Li, Partial transfer learning in machinery cross-domain fault diagnostics using class-weighted adversarial networks, Neural Networks, Volume 129, Pages 313-322, ISSN 0893-6080,..2020.

[80] Piyush Kant, Shahedul Haque Laskar, Jupitara Hazarika, Rupesh Mahamune, CWT Based Transfer Learning for Motor Imagery Classification for Brain computer Interfaces, Journal of Neuroscience Methods, Volume 345, 108886, ISSN 0165-0270,..2020.

[81] Xinghua Li, Zhongyuan Hu, Mengfan Xu, Yunwei Wang, Jianfeng Ma, Transfer learning based intrusion detection scheme for Internet of vehicles, Information Sciences, Volume 547, Pages 119-135, ISSN 0020-0255,..2021.

[82] Sheikh Saud, Basharat Jamil, Yogesh Upadhyay, Kashif Irshad, Performance improvement of empirical models for estimation of global solar radiation in India: A k-fold cross-validation approach, Sustainable Energy Technologies and Assessments, Volume 40, 100768, ISSN 2213-1388,..2020.

[83] Jie Wei, Hui Chen, Determining the number of factors in approximate factor models by twice K-fold cross validation, Economics Letters, Volume 191, 109149, ISSN 0165-1765,..2020.

[84] Tzu-Tsung Wong, Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation, Pattern Recognition, Volume 48, Issue 9, Pages 2839-2846, ISSN 0031-3203,..2015.

[85] Gaoxia Jiang, Wenjian Wang, Error estimation based on variance analysis of k-fold cross-validation, Pattern Recognition, Volume 69, Pages 94-106, ISSN 0031-320..2017.

[86] Nima Shiri Harzevili, Sasan H. Alizadeh, Analysis and modeling conditional mutual dependency of metrics in software defect prediction using latent variables, Neurocomputing, Volume 460, Pages 309-330, ISSN 0925-2312, https://doi.org/10.1016/j.neucom.05.043..2021

[87] Chao Ni, Xiang Chen, Fangfang Wu, Yuxiang Shen, Qing Gu, An empirical study on pareto based multi-objective feature selection for software defect prediction, Journal of Systems and Software, Volume 152, Pages 215-238, ISSN 0164-1212.2019.

[88] Ruchika Malhotra, Shine Kamal, An empirical study to investigate oversampling methods for improving software defect prediction using imbalanced data, Neurocomputing, Volume 343, Pages 120-140, ISSN 0925-2312. 2019.

[89] Yerima, S. Y., Alzaylaee, M. K., & Sezer, S. . Machine learning-based dynamic analysis of Android apps with improved code coverage. *EURASIP Journal on Information Security*, 1-24.2019.

[90] Siddiqui, M., Wang, M. C., & Lee, J. Data mining methods for malware detection using instruction sequences. In *Artificial Intelligence and Applications* (pp. 358-363).2008.

[91] N. Milosevic, A. Dehghantanha, K-K. R. Choo. "Machine learning aided Android malware classification", Computers & Electrical Engineering, 61, 266-274, 2017.

[92] A. Pektaş, M. Çavdar, T. Acarman, "Android malware classification by applying online machine learning", (International Symposium on Computer and Information Sciences, Kraków, Poland, 72-80, October 27–28, 2016.

[93] L. Onwuzurike, M. Almeida, E. Mariconti, J. Blackburn, G. Stringhini, E. D. Cristofaro, "A family of droids: Analyzing behavioral model based Android malware detection via static and dynamic analysis", *arXiv:1803.03448*, https://arxiv.org/abs/1803. 03448, 2018.