Egyptian Informatics Journal 23 (2022) 57–68

Contents lists available at ScienceDirect

Egyptian Informatics Journal

journal homepage: www.sciencedirect.com

Classification of Virtual Private networks encrypted traffic using ensemble learning algorithms

Ammar Almomani [a,b]

[a] IT-department- Al-Huson University College, AL-Balqa Applied University, P. O. Box 50, Irbid, Jordan
[b] Research and Innovation Department, Skyline University College, University City of Sharjah, P.O. Box 1797, Sharjah, United Arab Emirates

ARTICLE INFO

Article history:
Received 11 February 2022
Revised 18 June 2022
Accepted 22 June 2022
Available online 25 July 2022

Keywords:
Ensemble Learning
Machine learning
(VPN (and Non-VPN traffic analysis
Encrypted traffic

ABSTRACT

Virtual Private Networks (VPNs) are one example of encrypted communication services commonly used to bypass censorship and access geographically locked services. This study performed VPN and non-VPN traffic analysis and developed a classification system based on the new techniques of machine learning classifiers known as stacking ensemble learning. The methods used for VPN and Non-VPN classification use three machine learning techniques: random forest, neural network, and support vector machine. To assess the proposed method's performance, we tested it on a dataset containing 61 features. The experiment results accurately prove the study's classifiers to differentiate between VPN and Non-VPN traffic. The accuracy level was approximately 99% in the training and testing phase. The study's classifiers also show the best standard deviation, with a 100% accuracy rate compared to other A.I. classifier methods.

1. Introduction

Classification of network traffic has become popular recently due to breakthroughs in technology. The increasing growth of high-speed internet traffic demands has increased the need for such tasks. Network traffic classification facilitates the efficient management of network resources, which is also essential in managing and securing the network [1]. Accurate Classification of network traffic will assure that customers are provided with a quality of Service (QoS) and that the network is managed efficiently.

Various encryption methods are available today, and among the commonly used ones include HTTPS, SSH, SSL, and programs such as Tor and TrueCrypt. These methods are used in internet traffic encryption to safeguard the privacy of internet users from breaches. Thus, there is a need to classify these network traffics that are generated by the applications. The Classification will facilitate network management while preventing malware and maximizing network intrusion detection. Also, Classification will facilitate ISPs (Internet Service Providers) in carrying out a troubleshooting process for the network. However, classifying those encrypted network traffics has led to the emergence of new challenges [2].

Meanwhile, the most commonly used approaches in network traffic categorization are Deep Packets Inspection-based DPI and statistical and behavioural network traffic classification [3]. Notably, the port-based classification method has recently shown low accuracy, which is attributed to the unsystematic use of port or port concealment. As for the method based on DPI, it cannot be employed in classifying the encrypted network traffic because it could infringe users' privacy.

Researchers have hence opted to use the statistical and behavioural methodologies that are based on machine learning. These methodologies involve manually engineering the features associated with the raw traffic flow, which are extracted and applied in the Classification of network traffic [4]. Somehow, it should be noted that this method has a disadvantage, especially concerning the manually generated features, whereby a professional specialized in the networking field and has much domain knowledge is required to run the task. He/she must possess knowledge of the flow-based network traffic's statistical features, which must be considered while designing the classification model.

In Non-VPN encrypted traffic classification, network intrusion and user privacy violations are preventable through an ensemble learning algorithm. This can be exemplified in the Classification

Peer review under responsibility of Faculty of Computers and Information, Cairo University.
E-mail addresses: ammarnav6@bau.edu.jo, ammar.almomani@skylineuniversity.ac.ae

Production and hosting by Elsevier

https://doi.org/10.1016/j.eij.2022.06.006
1110-8665/

of encrypted network traffics so that ISPs could provide high-level service quality to users. The encrypted network traffics is categorized so the network can be debugged and managed while customer information remains safeguarded. In this regard, a system that could predict the category of network traffic data is proposed by using hidden patterns found in transmitted packets, as in specific security protocols, such as HTTPS, which use encryption. This system can significantly impact time-sensitive and essential applications because it prioritizes transaction application traffic over other traffic types. Consequently, the available bandwidth becomes optimized, and the network's performance is increased. This can significantly affect user experience.

Ensemble learning techniques are proposed in this study to classify encrypted network data. The traffic is classified using deep learning and machine learning methods. Deep learning is employed as a learning algorithm, which automatically learns features based on the raw network traffic [5]. Many researchers used different statistical features for developing their models. This has resulted in models with different overall accuracy levels. For instance, the researchers in ref. [6], employed 44 statistical features extracted from the raw PCAP file, while the researchers in ref. [7] employed four (4) statistical features. However, there are some researchers, such as [8,9] they did not provide nor mention the number of statistical features they have used.

The motivation of the research comes in twofold: one is motivated by network uses, and the other is motivated by achieving better tradeoffs to meet the identification needs, ultimately boosting network activities. Another reason is that the number of application types significantly impacts the accuracy of ML-based and ensemble learning classification. The preceding motivates us to develop a lightweight ensemble classification system capable of accurately and quickly identifying VPN traffic.

This study proposes a Virtual Private Networks Traffic analysis and Classification system based on stacking ensemble learning, and this is the first time this approach has been used in VPN and Non -VPN traffic classification problems. Based on the experiment results, the classifiers can distinguish between VPN traffic and non-VPN traffic. The main contribution of the paper includes1) the proposed ensemble model merger between three different machine learning algorithms from distinct categories. 2) the proposed method does not require deep packet inspection features; all used features are based header of packets. 3) the proposed approach achieves acceptable performance compared with the deep learning algorithm, which requires many resources. However, the proposed model can have the unknown threats detected effectively by showing a rate of accuracy greater than 99%. Other parts of this study are mentioned in the paragraph below:

Section 2 discusses the study's background and past works on Ensemble learning. The discussion highlights machine learning and combining various methods to create hybrid approaches. Relevant articles are reviewed, focusing on VPN and NON-VPN traffic analysis and Classification. Section 3 discusses the study methodology. The new VPN and NON-VPN Traffic analysis and classification systems are outlined via the stacking ensemble learning classifiers approach. Section 4 presents data on the experiments and the achieved results. The descriptions of the data used in the study are presented as well. In section 5, the researcher presents the conclusion and the study implications.

2. Related work

Encryption has been regarded as a critical feature in network traffic's cryptography, and it is increasingly pervasive on the web. Definitions for the expression (Classifying I.P. traffic) could be provided based on several functions, for instance, the encryption of traffic (through the use of HTTPS) throughout the process of communication from the source to the concerned destination based on the I.P. address, protocol encapsulation through the use of I.P. Security protocol, and application-specific port numbers, or particular applications, like Gmail. Some of those real-time applications have complex nature because they employ various services. Hence, the applications are usually identified with their associated tasks [1,10].

VPN technologies have been developing rapidly, becoming the preferred network access medium to route the internet traffic between 2 endpoints linked together on the web. The most salient feature of VPN is represented in the tunnelling of I.P. traffic that is already encrypted to ensure secure remote access to the servers. The IPsec protocol governing the VPN tunnelling mechanism shall contribute to maintaining the packet-level encryption. The protocol makes it semi-impossible to detect the application running through the tunnel's endpoints [1].

Countless studies have been conducted on the internet traffic classification and characterization in real- and Non-real-time. Most of these researches used statistical techniques and machine-learning ones to solve problems found in practice [1].

In this section, the researcher reviews some related articles on the VPN and Non-VPN as encrypted network traffic classification, focusing on the strengths and weaknesses of the encrypted network traffic classification scenario.

2.1. Classification of network traffic enabled by deep learning

Salman et al. [7] proposed a multi-level classification framework for the Classification of internet traffic data based on additional security and quality of service network requirements. Involving the use of the deep learning (Convolutional Neural Network) method, the authors classified the traffic into four groups: interactive, data transfer, streaming, and transaction. For each flow and sub-flow, the authors employed an MTU of 784 bytes and 16 packets for offline and online traffic classification from devices and applications of various kinds. Four discrete features were considered. There were two significant weaknesses of this study. Firstly, the online categorization did not consider the required bandwidth amount, and secondly, the study used imbalanced public datasets, which may affect the system's accuracy.

A Deep Packet approach was presented by Lotfollahi et al. [8] in developing an end-user application classification system (s). Deep learning algorithms were used in network traffic classification. Datasets provided by Gil et al. [11] for public ISCX VPN-Non-VPN traffic were used in the application of 1D convolutional neural networks and stack auto-encoding neural networks in testing and forming the proposed solution. A total of 17 application categories were identified. 1D-CNN generated the best results in the trial, specifically in the recognition and Classification of network traffic and application traffic, with 98 and 93% accuracy rates. Two significant limitations were highlighted. Firstly, the researchers did not state their used features in object classification. Hence, their proposed system could not detect traffic from any classes recognized during its training and testing. Also, the system runs on a packet header; therefore, it needs data and the internet to run.

Zeng et al. [12] proposed the Deep Full Range (DPR) system framework. The system used Deep Learning algorithms such as 1D-CNN, LSTM, and SAE-NN to classify encrypted network traffic and detect malware incursions. The proposed system's capacity to accurately classify encrypted network traffic was tested using a dataset. The limitation of this study is that it did not reveal the statistical features and the number of used packets per flow in the encrypted network traffic classification.

A classification system called SPCaps was proposed by *Cui et al.* [9] involving the use of Capsule Neural Network 11 that utilizes

vectors rather than scalars as a neuron for encrypted network traffic classification. The ISCX VPN-Non-VPN dataset by *Gil et al.* [11] was used in testing and training the system. SPCaps learn the spatial properties of network traffic and the location and sequence of fixed strings in packets. There are two significant issues of this study. Firstly, the authors did not consider the surfing and VPN browsing labels as these labels were eliminated from the datasets. Another issue is that the classes were reduced to 12 from 14, which may affect the system's accuracy. The system may fail to categorize new traffic data correctly, and thus, the system may suffer from an underfitting problem.

Shapira and Shavitt [13] proposed a " FlowPic " system to classify encrypted internet traffic. This system employs network flow data comprising packet sizes and arrival times in forming an image. Additionally, the CNN technique was used to classify traffic flow into its corresponding classes and identify applications. The flow-based dataset and tiny packets were used to test the proposed system's accuracy. The model was trained using the Non-VPN traffic dataset and was tested using the VPN traffic dataset. FlowPic showed an accuracy rate of 85% for Non-VPN networks classification and 98% for VPN networks classification. This study had two major weaknesses. Firstly, the authors did not include the time-series features that may increase the system's performance on Non-VPN traffic when used as input during implementation. Furthermore, Deep learning has been used in other studies, for instance, [5,14,15].

2.2. Network traffic classification aided by Machine learning

McGaughey et al. [6] formulated a statistical feature selection method involving selecting features with predictive values from raw network traffic data by applying the Fast Orthogonal Search (FOS) algorithm for encrypted communications classification. Selecting just a subset of features from the raw traffic data in the classification process can minimize errors. The raw traffic data feature is the primary feature set extracted using NetMate. A total of 44 features were accordingly extracted [16]. In this study, the problem that may arise is overfitting which can occur when the classification device is fed with unfamiliar traffic data, particularly in the situation of Dropbox network traffic. Another problem is the dependency of the method on professional knowledge of the features, which does not happen in sequential feature selection and deep learning.

Aouini et al. [17] constructed a system to classify residential encrypted network data using the C5.0 machine learning algorithm. The system employed an extensive French residential aggregation network with over 34,000 users as traffic data input. From the training data, the proposed system could classify seven features. Features with high accuracy rates in positively impacting the categorization process were manually created during extraction; therefore, the authors could not determine the importance of the source of the option. Meanwhile, the turbo mode of C5.0 increased the training time nine-fold, and despite the use of the destination port to improve the speed, there was a risk of overfitting.

Tong et al. [18] proposed a method of encrypted network traffic classification grounded upon CNN. This method classifies traffic in two stages, involving flow-based and packet-based information. Similar features are also being used in Google services of file transfers, voice calls, video streaming, chat, and Google Play Music, through the protocol of QUIC (Quick UDP Internet Connection). QUIC is a transport layer network protocol created by Google, and it provides security comparable to TLS, except that it does not require a devoted server. This method's inability to detect flow-based traffic during first-stage categorization causes the online system to be unfit for offline applications and for the initial traffic flow prediction. In addition, there was no clarification on the obtained features. Also, the use of flow-based features has caused the computational and Classification time to increase.

Still, the problem of classifying a network flow as VPN or Non-VPN encrypted remains unresolved. In the present research, the researcher employed ensemble learning methods for classifying a network flow as VPN or Non-VPN encrypted. The methods require a simple setup, and the results are interpretable. Through deep learning for VPN and Non-VPN Traffic analysis and Classification, fast convergence on large and small datasets could be achieved. The full description of the proposed system is provided below:

The models of the stacking method are developed by employing several learning algorithms. The stacking method creates a combiner algorithm, which is trained to make ultimate predictions using the predictions that have been made using the base algorithms. This combiner algorithm may be any ensemble method, as shown in Fig. 1 [19].

Based on IoT technology, Al-Qurabat et al. [21] used compression and minimum description length technology to send data from sensors to set up a remote monitoring system for smart agriculture. The results showed that this method could significantly slow down the speed of data transmission and provide a better way to monitor in real-time. China was slow to start smart farming, and smart farming based on the Internet of Things is still in its early stages.

Al-Qurabat and Kadhum [22] proposed a lightweight lossless compression algorithm based on Differential Encoding (DE) and Huffman algorithms that is especially advantageous for IoT sensor nodes that monitor the environment's properties, particularly those with limited CPU and memory resources. Instead of developing novel ad hoc algorithms, they demonstrated that, given basic knowledge of the features to be monitored, traditional Huffman coding might be utilized to successfully express the same attributes measured at different times and locations. Although the proposed system does not achieve its theoretical maximum, temperature measurements show that it beats standard methods explicitly created for WSNs.

All works that use machine learning algorithms for network traffic classification must be retrained to remain robust in the face of new and diverse data. Furthermore, most such efforts select a subset of applications or protocols to test the feasibility of Classification. As a result, such solutions are most effective for training applications and protocols, posing scalability and adaptation concerns. In any discipline of traffic analysis, the problem of high false-positive rates is a crucial obstacle. Unfortunately, due to the great diversity and variability presented in recent years, it does

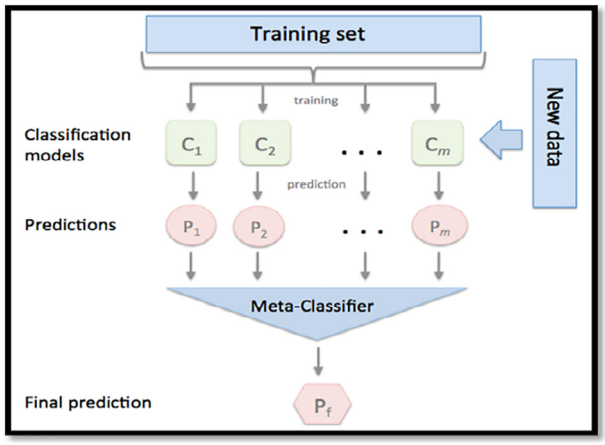

Fig. 1. The Concept Diagram of Stacking Ensemble Learning [20].

not look easy to build a universal solution capable of covering an entire area. So the primary concern of the proposed approach is to build a scalable and robust system using a dataset containing diverse VPN applications and an ensemble to gain the performance of several machine learning in one system.

In the present research, the ensemble model was designed using 3 machine learning from various categories. The researcher used random forest (R.F) from the tree categories, Neural networks as computational machine learning (N.N.), and Support Vector Machines (SVM) as statistical machine learning.

3. The proposed methodology:

Ensemble machine learning techniques to increase the capacity of prediction of current models. Meta-algorithms are thus applied to predict the model's core mechanisms (s). As opposed to the simpler prediction models, these models have the potential to identify anomalies, making them appropriate for network intrusions detection [23].

The proposed system proved superior in VPN and Non-VPN Traffic analysis and Classification. An approach to VPN and Non-VPN Traffic analysis and Classification Systems with ensemble learning adaptive techniques is accordingly proposed in this study. Fig. 2 illustrates Algorithm 1, which encompasses a strategy for implementing the stacking ensemble learning in the proposed system. Meanwhile, the full description of the VPN and Non-VPN Traffic analysis and the classification system based on the stacking ensemble learning is shown in Fig. 3.

3.1. VPN and Non-VPN dataset

The researcher conducted several experiments assessing the proposed system and having it compared with other comparable systems. The VPN-Non-VPN dataset [11,24], utilized in numerous experiments by researchers like [25], was used in this study. The researcher utilized the Intel(R) Core (T.M.) i7-8700 CPU @ 3.20 GHz 3.19 GHz, RAM 32.0 GB, and WINDOWS 10 64-bit operating system to execute the experiments.

The ISCXVPN2016 dataset was used for the generation of Non-VPN and VPN traffic. The researcher listed the details related to VPN and Non-VPN traffic categories in the first table. The researcher also presented the applications employed in producing network traffic. There are 43,191 records, as displayed in Table 2. Meanwhile, Table 1 shows the data on the Encrypted applications. Table 3 presents the information on the quantity of data used in the study's experiments.

3.2. ***Data and features Pre-Processing***:

Data Cleaning Phase:

From the full dataset employed in the present experiments which included about 77 features, the researcher utilized only sixty-one (61) features, because the remaining sixteen (16) features contained many NAN, null, and zero-value records.

In the present phase:

- The researcher handled the missing values and replaced them with the mean value of the respective column.
- The null and NaN values records were deleted, and the zero columns were removed.
- The columns, including infinite ones (without value), were deleted.

3.3. The proposed ensemble learning system:

This section describes the proposed VPN and Non-VPN Traffic analysis and a classification system based on stacking ensemble learning. Based on the empirical results, ensemble learning shows better performance when significant differences among the ensemble models exist. The stacked model involves several learning stages, and the ensemble learning approach has been the most popular. Therefore, the researcher proposed a new system, which possesses 2 levels of structure, namely a base module and a combining module. It aims at employing the base modules and combining ones.

3.3.1. Base module –level 1

The base module involves training, testing, and utilizing a set to have the base classifiers tested and trained. After that, decisions shall be forwarded to the logistic regression algorithm to make decisions in the second level. The present study addresses VPN and Non-VPN Traffic problem detection on the internet. This problem is connected to binary Classification. Therefore, this study uses three (3) base classifiers to create a base module for the proposed system. The latter classifiers involve random forest (R.F.) [33], Artificial Neural Networks (ANNs)[26], and support-vector networks

Input: Train Data $T = [Y_i, Z_i]_{i=1}^{n}$ (Y_i R", $Z_i \epsilon$ Z)
Output: Predictions from the ensemble E

1. Impose cross-validation to prepare a training set for meta-classifier
2. Randomly split T into "n" equal size subsets, Le. T= (T1, T2, T3... Tn)
3. For $n \leftarrow 1$ to N
   Learn base classifiers, namely random forest, neural network, and support vector machine (SVM)
   For $n \leftarrow 1$ to m
   Learn a classifier Pin from T or T,
   End for
4. Formulate a training set for meta-classifier (Logistic regression)
   For each $Y_n \in T_n$
   Extract a new instance ($yi.21$), Where $y;' = (P_{n1}(Y_i), P_{n2}(Y_i), \ldots P_{nm}(Y_m)$
   End for
5. End For
6. Return $Z_i = \{Z_i, Z_2, Z_3 \ldots . Z_m\}$ from ensemble

Fig. 2. Algorithm 1 as a Strategy for Implementing Stacking Ensemble Learning in the Proposed System.

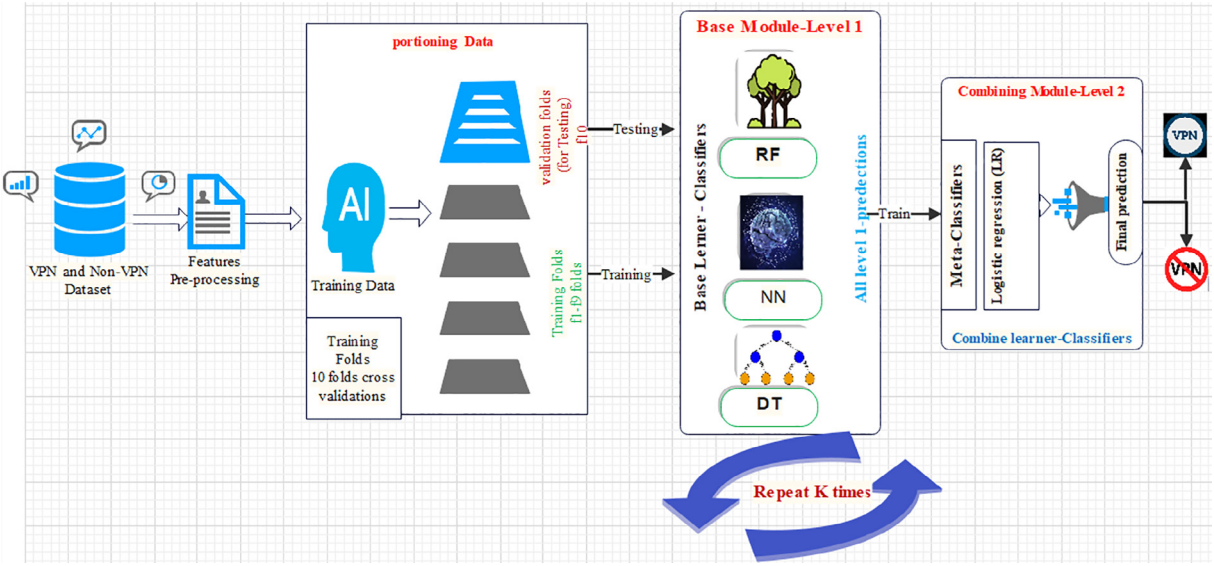

Fig. 3. VPN and Non-VPN Traffic analysis and Classification System based on Stacking Ensemble Learning.

Table 1
Details on the Encrypted Applications.

Traffic	Application
FTP	FTPS, Skype, and SFTP,
Chat	ICQ, AIM, Skype, Facebook, and Hangouts
Audio	Spotify
Video	Vimeo and YouTube
Email	POP3S, IMAPS, and SMTPS

Table 2
Specific Quantity of Data.

Traffic	Chat	FTP	Email	Audio	Video	Class
VPN	4476	2501	569	13,060	1144	1
Non-VPN	65,521	1795	5071	3296	4758	2

[27]. All the algorithms are deemed adequate for their success in solving binary classification problems.

Random forest (R.F.):

Random Forests could be employed for a categorical response variable, called in ref. [28] "classification," or a continuous response called "regression." Similar to that, the predictor variables may be categorical. It may continue. From a computational view, random forests are deemed appealing. That is because they are capable of handling the classification and regression. Random forests are fast relative to the train in predicting and rely on only 2 or 1 tuning parameter. They possess a built-in estimate of the generalization error and can get employed directly for addressing high-dimensional problems.

Random forests can be easily implemented in parallel. From a statistical view, random forests are deemed appealing. This is attributed to the additional features they offer. The features include measures of variable importance, in addition to differential class weighting as well as visualization. They also include missing value imputation and outlier detection. They include unsupervised learning as well [29]. Details are shown in Fig. 4.

Support-vector networks (SVM) is a known classifier. This classifier can classify with a limited set of samples. However, it can have predictions optimized [31]. Researchers in ref. [32] proved that SVM is better than ANNs in terms of detecting intrusions while experimenting with basic security module (BSM) audit data obtained from DARPA intrusion detection dataset. This is because ANN requires much training data. As for SVM, it can show a high level of performance with fewer relatively data. In addition, SVM can carry out the execution process faster. Despite that, it is known that SVM excels in binary classification. However, when SVM is combined with other classifiers, it could also show superior multiclass classification. Details are shown in Fig. 5.

Artificial Neural Networks (ANNs)[34]: ANNs may be called (neural networks), and ANNs comprise an input layer of neurons, and 1, 2, or 3 hidden layers of neurons. ANNs also carry a final layer of output neurons. Fig. 6 accordingly displays a typical architecture. In the figure, lines connecting neurons are shown too. It should be noted that each connection comprises a numeric number named (weight). Meanwhile, the output, *hi*, of neuron I in the hidden layer is as expressed below:

$$h_i = \delta \sum_j^N VijXj + T_i^{hid} \quad (1)$$

From the expression above: σ() is called transfer or activation function, T_i^{hid} denotes the threshold terms of the hidden neurons, V_{ij} *is* the weights, N is the number of input neurons, and X_j is the inputs to the input neurons. The goal sought from the activation function is represented in the introduction of Nonlinearity into the neural network to bound the value of the neuron to ensure that the divergent neurons do not paralyze the neural network. Details are shown in Fig. 6.

The generation methods of the stacking model involve the use of cross-validation. Based on Fig. 2, the researcher selected the cross-validation K-fold method. First, the original dataset is partitioned into a test set D and training set D train. While carrying out the K -fold cross-validation procedure, the D_{train} is split into K disjoint subsets with comparable sizes. It should be noted that each subset is called a fold, and the subset seeks to maintain the same class scale as the dataset is deemed original. The cross-validations aim to have the training phase executed on the D_{train} and the testing phase on the D_{test}. The researcher chose 1 classifier $C_n(1,\ldots, N)$ as an example. Here, N represents the number of base classifiers. During the training phase, the researcher employed one 1 subset as a validation set D_{valid} while the remaining subsets were used as training sets. This procedure was repeated (K) times.

In terms of the prediction results on the validation sets, they were joined into a matrix of prediction P_n (n = 1…, N). The

Table 3
Features details [25].

Feature	Description
IP	[Source IP, destination IP]
Port	[Source port, destination port]
Protocol	The protocol of the flow
Flow duration	The duration of the flow
Packet	The overall packet size in forwarding and backward directions
Length of packet	The size of packets in the forward and backward directions comes in various metrics
Flow packet length	The length of the flow packet comprises the maximum, minimum, average, standard deviation, and variation of the flow
Flow bytes/s	The number of transported bytes per second
Flow packets/s	The number of exchanged packets per second
Packets/s	Total packets per second (forward and reverse included)
Flow IAT	In between-packet IAT (Total, Maximum, Minimum, Mean, and Standard Deviation)
Backward IAT	Total, maximum, minimum, mean, and standard deviation of the time elapsed between two packets in the backward direction
Flags	The number of times packets moving in the 'forward, backward' direction had the 'PSH, URG' flag set (0 for UDP)
Flag count	The number of packets with FIN, SYN, RST, PSH, ACK, URG, CWE, and ECE in them
Header length	Bytes are used in the 'forward, backward' direction for headers
Ratio	The inverse of the inverse
Average packet size	The average package volume
Segment size avg	When observed in a "flow, forward, backward" orientation the average size
Bytes/Bulk avg	The average "forward, backward" bulk rate, bytes-wise
Packet/Bulk avg	Forward and reverse traffic volume averages within one direction
Bulk rate avg	Bulk rates in the "forward, backward" direction.
Subflow packets	Packet count in the 'forward, reverse' direction of a sub-flow.
Subflow bytes	The average amount of bytes moved by subflow forward and backward
Init win bytes	The amount of bytes transported in the 'forward, backward' direction in an initial window
Forward Act data pkts	TCP data payloads in forwarding packets with 1-byte minimum
Forward seg size min	Forward motion in the smallest section size
Active time	The required average, maximum, minimum, and standard deviation of time for flow activation
Idle time	The required average, maximum, minimum, and standard deviation of the time to make the flow idle

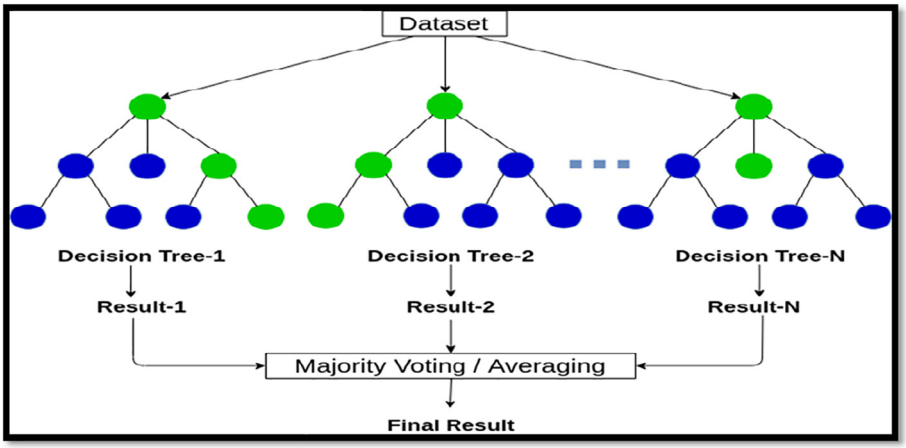

Fig. 4. Random Forest Inference for a Simple Classification Example with N_{tree} = 3 [30].

researcher applied Cn to generate a classification matrix during the testing phase. After repeating this procedure for (K) times, he obtained (K) matrices of Classification and averaged them by rows to generate a matrix A_n(n = 1,. . ., N). This procedure shall be repeated to the N classifiers. Regarding the whole, the matrices of prediction P_n will be combined into a set of training P. Regarding all the A_n. They are averaged for generating a modern test set (A) for making a prediction result representing every classifier in level 1. Then, the result was forwarded to level 2- *meta*-classifier. Through the proposed method, the number of a fold K is 10. That is the size of the real dataset.

3.3.2. Combining module – level2 by logistic regression (Meta Classifier):

Through stack generalization, the ensemble's output will serve as the *meta*-classifier level inputs that build the final decision based on the logistic regression algorithm. Logistic regression [36]: In statistics, the logistic model (or logit model) is employed for modelling the probability of a specific event or class, like win/lose and pass/fail**.**

Logistic regression (L.R.): It is a standard classification model that's probabilistic and statistical. It has been increasingly used in various fields. Such fields include marketing, I.T., and social sciences. On the other hand, the outcome of linear regression is always similar for all samples. In this regard, the probability of a result that is either negative or positive will determine the success of L.R. as an approach. Hence, LR is commonly employed in the classification process.

More formally, for a sample $x_i \in R^p$ whose label is denoted as y_i, the probability of yi being positive is predicted based on the following equation:

$$p_{yi+1} = \frac{1}{1 + e - \beta T x_i} \quad (2)$$

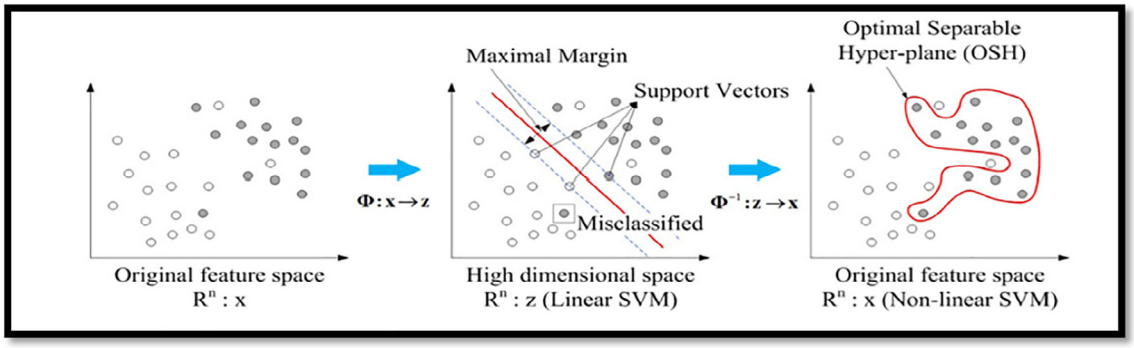

Fig. 5. Support-Vector Networks (SVM) [33].

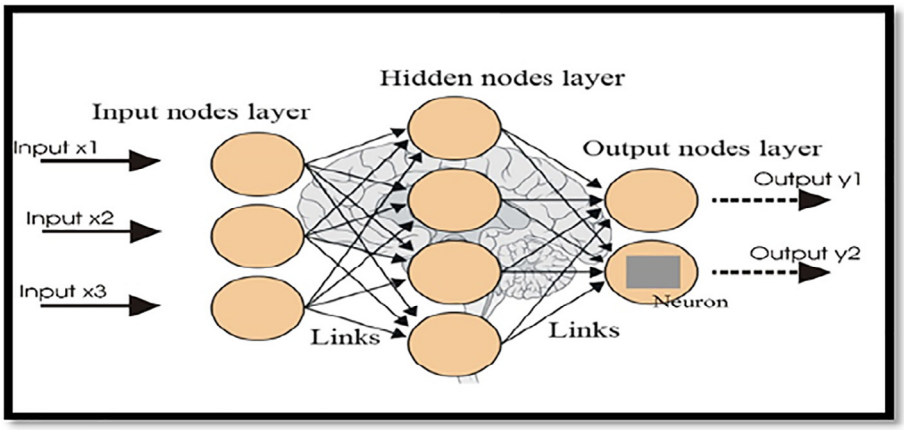

Fig. 6. Neural Networks [35].

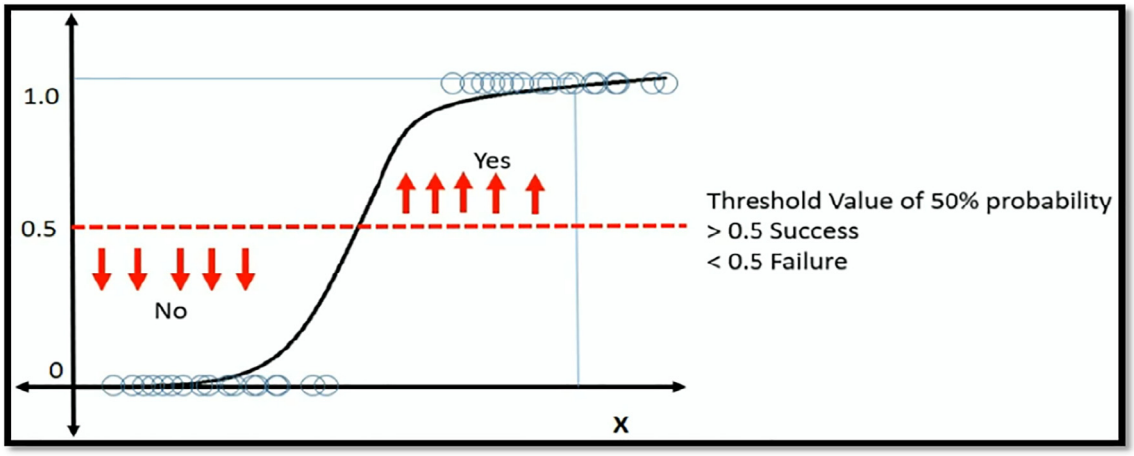

Fig. 7. Classification with Logistic Regression[38].

Given the L.R. model parameter β. For obtaining a parameter with good performance, a set of labelled samples {(x1, y1), …, (x_n, y_n)} is collected for learning the L.R. parameter β, which has the induced likelihood function maximized over the training samples [37]. Despite that, combining the level strategies will improve the final classification capabilities. Details can be observed in Fig. 7. The researcher employed the parameter shown below in the Regression logistic algorithm.

4. Experiments results

In this section, the researcher presents the results obtained from the dataset described in subsection 3.1. The conclusion and the most important results are duly highlighted. The researcher used various measures utilized in regression to assess the performance level of the base methods and ensemble scheme. Those measures were: accuracy, Precision, Recall, AUC, F1_Score,

Table 4
Measurements Used in the Experiments.

Number	Measurement	Equation	Meaning
1	Accuracy	$\frac{\lvert TP\rvert+\lvert TN\rvert}{\lvert TP\rvert+\lvert TN\rvert+\lvert FP\rvert+\lvert FN\rvert}$ $\frac{\lvert TP\rvert+\lvert TN\rvert}{\lvert TP\rvert+\lvert TN\rvert+\lvert FP\rvert+\lvert FN\rvert}$	The percentage of correct predictions
2	Precision	$\frac{\lvert TP\rvert}{\lvert TP\rvert+\lvert FN\rvert}$	The percentage of correct positive predictions
3	Recall	$\frac{\lvert TP\rvert}{\lvert TP\rvert+\lvert FN\rvert}$ $\frac{\lvert TP\rvert}{\lvert TP\rvert+\lvert FN\rvert}$	The percentage of positive labelled instances that were predicted as positive
4	AUC		The AUC (area under the ROC curve) can range from 0.1 to 1. The closer AUC is to 1, the better the performance. In other words, if the AUC is equal to 0.5, then the Classification randomly performs.
5	F1_Score	$2\frac{precision.recall}{precision+recoll}$ $2\frac{precision.recall}{precision+recoll}$	A measure of test accurateness computed by executing the test precision and recall
6	Root Mean Squared Error (RMSE) Or Mean_squared_error (MSE)	$\sqrt{\sum_{i=1}^{N}\frac{\left(y_i-\dot{y}_i\right)^2}{N}}$	A residual standard error of zero (RMSE) signifies that the model output precisely matches what was observed for the given number of input samples (n), i_{th} real output (y_i), and i_{th} framework output (y_i). However: mean squared error (MSE) is used to assess the regression problem's accuracy

mean_squared_error, False Negative rate (FNR), and False Positive rate (FPR). Details are provided in Table 4. Those measures are defined in ref. [39,40]. However, The experimental method was scheduled into the following steps: 1) the dataset is pre-processed. 2) the missing and corrupted record is removed. 3) the dataset is divided into training and testing. 4) the experiment is performed, and the results are estimated. To ensure the quality of the learned neural network agent, K-fold cross-validation method is used to estimate the error rate of classifiers. The data set is randomly divided into N samples in cross-validation, and the evaluation is applied N times. Each time, N-1 samples are chosen for training, and the last sample is utilized to assess the accuracy of the classifier.

Based on Tables 5 and Fig. 8, the study's proposed method has the best accuracy compared with other A.I. classifier methods, with 99.3% accuracy level, while NN. = 30%, LG = 95%, KNN = 97%, SVM = 96%, and KNN = 97. Also, the proposed system shows the best result compared to other measurements such as precision and recall, with an approximately 99.3% accuracy rate. Furthermore, the proposed method shows the lowest error rate compared to other methods, with an MSE of roughly 0.6%. The standard deviation in the training phase represents the stability of the proposed system. Further details are shown in Fig. 9 below.

Based on Table 5 and Fig. 9, the proposed Ensemble System shows the best standard deviation. This means that the proposed system's stability is the best compared to other A.I. classifier methods, as it achieved 100% stability.

Based on Tables 6 and Fig. 10, the proposed method shows the best accuracy compared to other A.I. classifier methods. Specifically, the proposed method achieved about 99% accuracy level, while NN = 30%, LG = 95%, KNN = 97%, SVM = 96%, and KNN = 97%. Also, the proposed system shows the best result compared to other measurements, like precision and recall, with 91% accuracy. Furthermore, the proposed system has the lowest error

Table 5
The Classification Results for Various Methods of Machine Learning - Training Data Results.

Method	Accuracy	AUC	Mean squared error
NN	0.3020 ± 0.0011	0.5310 ± 0.0022	0.6980 ± 0.0011
LG	0.9538 ± 0.0009	0.8087 ± 0.0062	0.0462 ± 0.0009
KNN	0.9765 ± 0.0006	0.9919 ± 0.0002	0.0235 ± 0.0006
SVM	0.9574 ± 0.0005	0.8555 ± 0.0145	0.0426 ± 0.0005
Proposed Ensemble model	0.99314 ± 0.00054	0.999998 ± 0.000002	0.006859 ± 0.000543

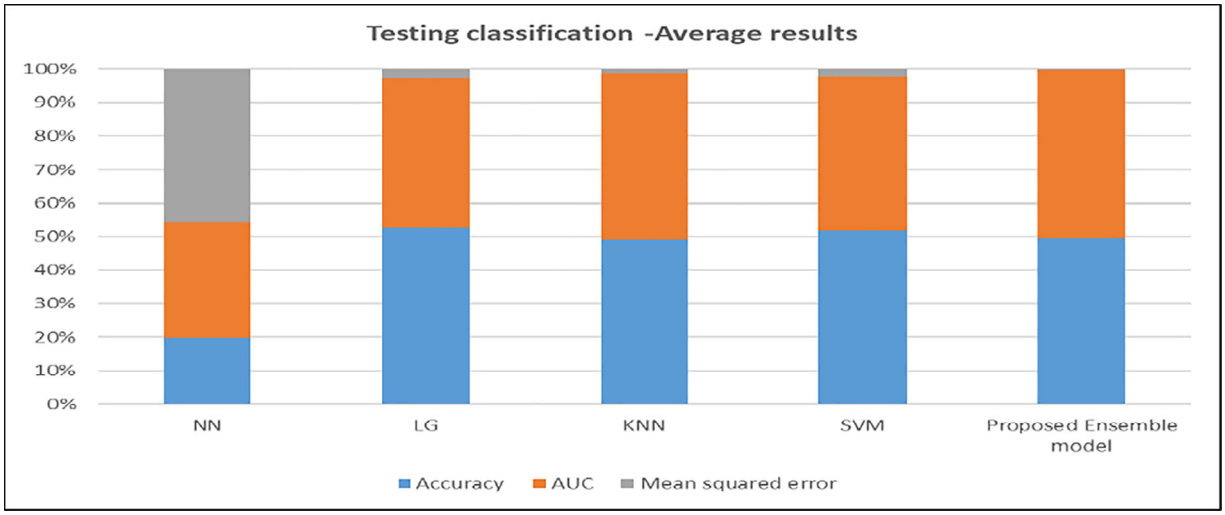

Fig. 8. Training Classification - Average results based on A.I. Classifier Methods compared with Proposed Ensemble System.

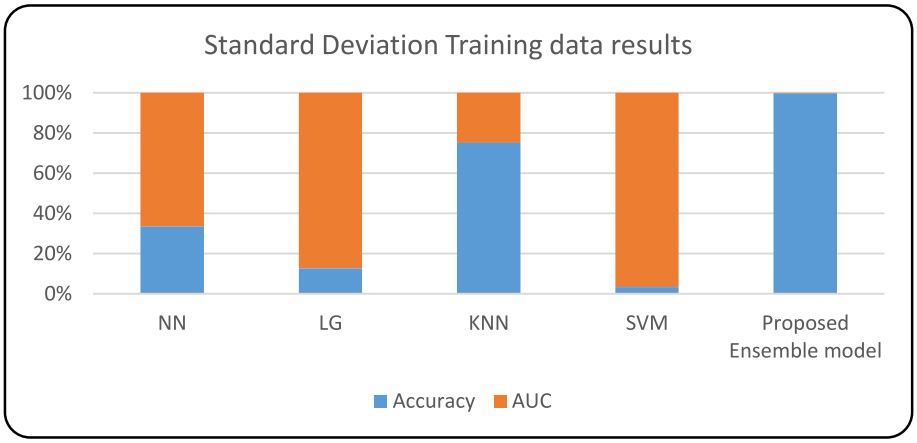

Fig. 9. Training Classification –Standard Deviation Results based on A.I. Classifier Methods Compare with Proposed Ensemble system.

Table 6
The Classification Results for Various Methods of Machine Learning - Testing Data Results.

Method	Accuracy	Precision	Recall	AUC	F1-Score	Mean squared error
NN	0.3020 ± 0.0103	0.0663 ± 0.0044	0.5318 ± 0.0193	0.5314 ± 0.0195	0.3020 ± 0.0103	0.6980 ± 0.0103
LG	0.9536 ± 0.0047	0.6389 ± 0.0483	0.7744 ± 0.0245	0.8082 ± 0.0309	0.9536 ± 0.0047	0.0464 ± 0.0047
KNN	0.9685 ± 0.0034	0.8026 ± 0.0416	0.8189 ± 0.0186	0.9141 ± 0.0158	0.9685 ± 0.0034	0.0315 ± 0.0034
SVM	0.9571 ± 0.0040	0.6794 ± 0.0400	0.7766 ± 0.0216	0.8400 ± 0.0318	0.9571 ± 0.0040	0.0429 ± 0.0040
Proposed Ensemble model	0.98876 ± 0.00174	0.98720 ± 0.01196	0.91356 ± 0.01554	0.97769 ± 0.00723	0.98876 ± 0.00174	0.01124 ± 0.00174

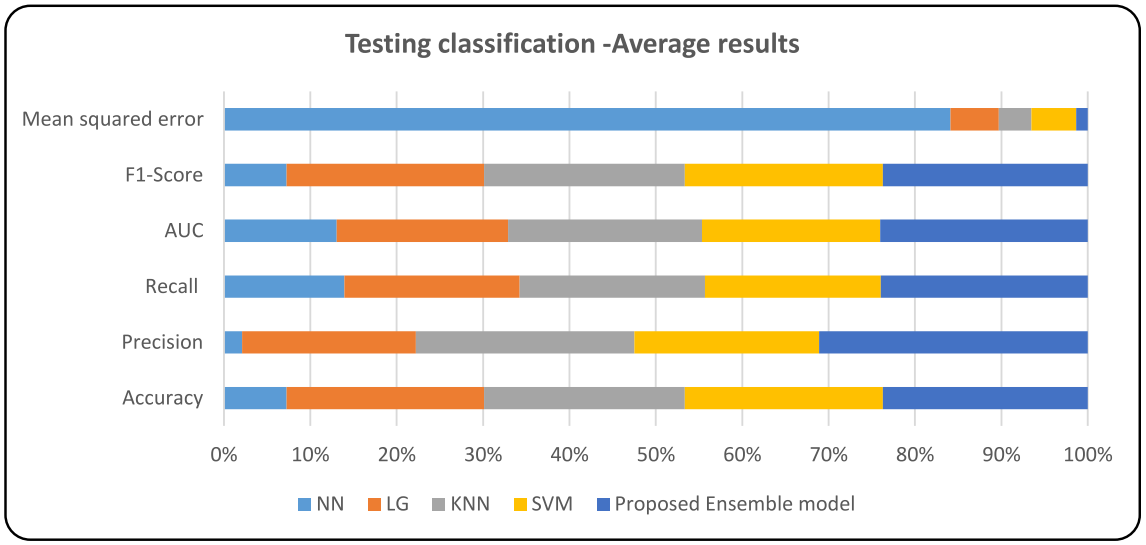

Fig. 10. Training Classification - Average Results based on A.I. Classifier Methods compared with Proposed Ensemble System.

rate in comparison to other methods, with MSE = 1%. The standard deviation in the testing phase shows that the proposed system's stability can flow in Fig. 11 below.

To measure the stability of the results in the testing phase, the standard deviation between cross-validation folds results was estimated. As shown in Table 6 the proposed approach achieved the lowest standard deviation for ACC, F-Score, Recall and AUC at 0.0017, 0.0017, 0.015 and 0.007, respectively.

Based on Fig. 11, the proposed Ensemble System shows the best standard deviation. This means that the stability of the proposed system is the best compared to other A.I. classifier methods, with about 100% stability.

Table 7 introduces the confusion matrix based on the testing experiment.

The choose machine learning and the meat classifier at the final layer contribute to the complexity of the suggested approach. It is essential to select a base classifier with minimal complexity and a wide range of base classifiers. Based on the above two ideas, we consider Random Forest (RF), Neural networks (NN), and support vector machine (SVM) as base classifiers (SVM).

Table 8 compares the build time (training time) of our proposed method. The LG algorithm achieved the best training time results compared with other machine learning in our experiment. However, the maximum time is based on the ensemble model due to its combination with several machine learning. However, the proposed model achieves a long training time, but this approach achieves high performance on all evaluation matrix.

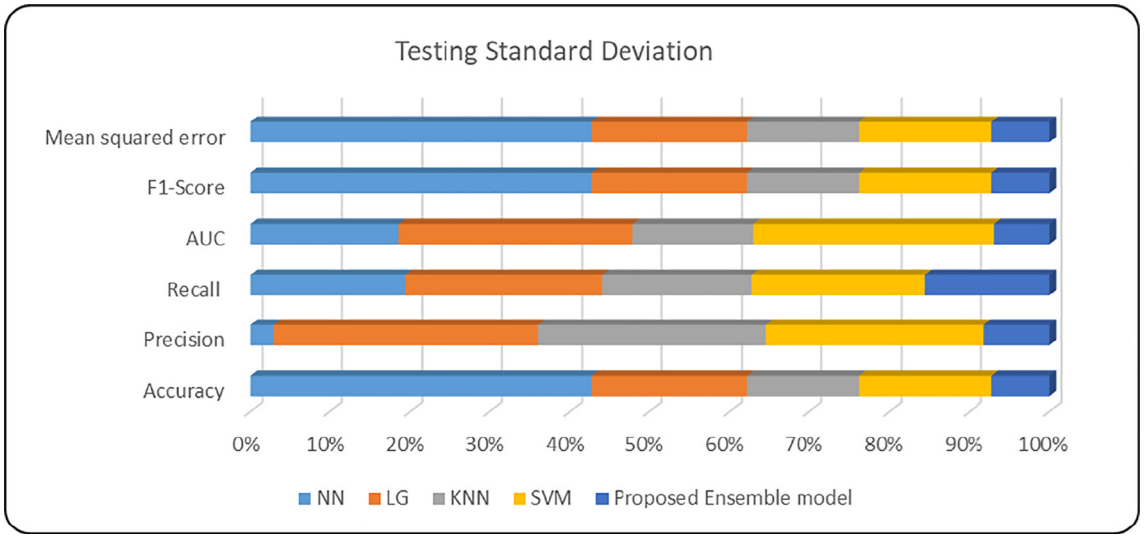

Fig. 11. Standard Deviation Results based on A.I. classifier Methods compared with Proposed Ensemble System.

Table 7
Results of Different Machine Learning Methods confusion matrix (Testing Phase).

Ensemble			SVM		
	True Positive	True Negative		True Positive	True Negative
Predicted Positive	21,530	499	Predicted Positive	21,233	520
Predicted Negative	220	20,941	Predicted Negative	517	20,921
KNN		NN			
	True Positive	True Negative		True Positive	True Negative
Predicted Positive	21,130	820	Predicted Positive	8150	14,400
Predicted Negative	620	20,621	Predicted Negative	13,600	7041
LG					
	True Positive	True Negative			
Predicted Positive	21,140	810			
Predicted Negative	610	20,631			

In our experiment, 10-fold cross-validation was used to estimate the classifier performance. The results showed that the proposed ensemble model performs better than any individual ML classifier. However, to determine if the differences between different model predictions are statistically significant, we used the Wilcoxon signed-rank test to ensure statistical differences between the proposed ensemble model and each machine learning classifier. So, we perform the one-tailed hypothesis test. The null hypothesis (H_0) is that "Two detection methods have the same accuracy performance ($X1 = X2$). If the $p - value \geq 0.05$, it will accept the null hypothesis. If the $p - value \leq 0.05$, the null hypothesis will be rejected at a confidence level of 95%. Table 9 estimates the Wilcoxon statistical results for the accuracy metric. Based on the results, we can conclude that there are no significant statistical differences between SVM and KNN, which means they have the same performance in classifying VPN network traffic. However, when comparing the Ensemble model with the machine learning classifiers, we found significant differences between them, whereas a p-value < 0.05 that mean can reject the H0.

Table 10 shows the results of the comparison of our results with those in other published work based on the analysis of network traffic to classify encrypted VPN traffic. The table also demonstrates that the proposed approach's accuracy rate is better than those gained by previous solutions. Moreover, due to its simple design, our solution can identify VPN traffic with a lighting system and low complexity performance.

Table 8
Training time comparison.

Method	Train time
NN	91 sec
LG	70 sec
KNN	144 sec
SVM	156 sec
Proposed Ensemble model	**356 sec**

Table 9
Comparison of machine learning P-value for Accuracy measures (training phase).

	NN	SVM	LG	KNN
NN				
SVM	$p \leq 0.05$			
LG	$p \leq 0.05$	$p \geq 0.05$		
KNN	$p \leq 0.05$	$p \geq 0.05$	$p \leq 0.05$	
Proposed approach	$p \leq 0.05$	$p \leq 0.05$	$p \leq 0.05$	$p \leq 0.05$

Table 10
Comparison with other published work based on the same dataset.

Paper	**Result ACC (%)**	**Techniques**
Draper-Gil et al.[11]	92 ACC	Decision Tree, KNN
Lotfollahi et al.[41]	86.5 ACC	CNN, SAE
Wang et al.[5]	94 ACC	CNN-1D
Izadi et al.[42]	97	Fusion (CNN, DBN, MLP)
Our proposed ensemble method	98	Stacked ensemble

5. Conclusion

The researcher proposed a VPN and Non-VPN Traffic analysis and Classification system based on new Machine Learning classifiers techniques called stacking ensemble learning, used for the first time in a VPN and Non-VPN attack problem. Ensemble learning permits combining the predictions made by several learning mechanisms to increase the accuracy of the predictions.

The researcher employed an ensemble learning stacking scheme involving the use of 2 levels of learning techniques. The prediction made by the 1st level technique was passed to the top technique that combines them to produce the final forecasting. The researcher employed 3 based learning methods, namely neural networks, random forest, and support vector machines which were used by the top-level technique to make the final predictions based on meta classifiers as logistic regression. Some historical data were employed to attain a prediction. Various measurements were applied, such as accuracy, precision, and RMSE. Comparisons were made between the results obtained from the ensemble technique and those from the single technique. The predictions from the ensemble scheme were superior to those from other techniques. The proposed method proved its ability to detect unknown and known threats effectively by showing an accuracy rate that exceeds 99% in the training phase and an accuracy rate of 99% in the testing phase. However, the main limitation of our research is that unable to perform deep packet inspection due to performance and privacy requirements. For future research, the researcher will investigate other ensemble schemes by employing several methods, including those based on the S.P. Theory of Intelligence and those based on the support vector machines. The effectiveness of the proposed system will be tested using other databases and attack types.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The research reported in this publication was supported by Al-Balqa Applied University in Jordan, Grant Number: **DSR-2021#398**

References.

[1] Bagui S, Fang X, Kalaimannan E, Bagui SC, Sheehan J. Comparison of machine-learning algorithms for classification of VPN network traffic flow using time-related features. Journal of Cyber Security Technology 2017;1(2):108–26.
[2] Cao Z, Xiong G, Zhao Y, Li Z, Guo L. In: A survey on encrypted traffic classification. Springer; 2014. p. 73–81.
[3] Y. Bengio, "Deep learning of representations for unsupervised and transfer learning," in *Proceedings of ICML workshop on unsupervised and transfer learning*, 2012, pp. 17-36: JMLR Workshop and Conference Proceedings.
[4] Wang W, Zhu M, Zeng X, Ye X, Sheng Y. In: Malware traffic classification using convolutional neural network for representation learning. IEEE; 2017. p. 712–7.
[5] Wang W, Zhu M, Wang J, Zeng X, Yang Z. In: End-to-end encrypted traffic classification with one-dimensional convolution neural networks. IEEE; 2017. p. 43–8.
[6] McGaughey D, Semeniuk T, Smith R, Knight S. In: A systematic approach of feature selection for encrypted network traffic classification. IEEE; 2018. p. 1–8.
[7] Salman O, Elhajj IH, Chehab A, Kayssi A. In: A multi-level internet traffic classifier using deep learning. IEEE; 2018. p. 68–75.
[8] Lotfollahi M, Siavoshani MJ, Zade RSH, Saberian M. Deep packet: A novel approach for encrypted traffic classification using deep learning. Soft Comput 2020;24(3):1999–2012.
[9] Cui S, Jiang B, Cai Z, Lu Z, Liu S, Liu J. In: A session-packets-based encrypted traffic classification using capsule neural networks. IEEE; 2019. p. 429–36.
[10] T. C. Obasi, "Encrypted Network Traffic Classification using Ensemble Learning Techniques," Carleton University, 2020.
[11] Draper-Gil G, Lashkari AH, Mamun MSI, Ghorbani AA. Characterization of encrypted and vpn traffic using time-related. In: in *Proceedings of the 2nd international conference on information systems security and privacy (ICISSP)*. p. 407–14.
[12] Zeng Y, Gu H, Wei W, Guo Y. $ Deep-full-range $: A deep learning based network encrypted traffic classification and intrusion detection framework. IEEE Access 2019;7:45182–90.
[13] Shapira T, Shavitt Y, "Flowpic. In: Encrypted internet traffic classification is as easy as image recognition,". IEEE; 2019. p. 680–7.
[14] Lopez-Martin M, Carro B, Sanchez-Esguevillas A, Lloret J. Network traffic classifier with convolutional and recurrent neural networks for Internet of Things. IEEE Access 2017;5:18042–50.
[15] H. Yao, C. Liu, P. Zhang, S. Wu, C. Jiang, and S. Yu, "Identification of encrypted traffic through attention mechanism based long short term memory," *IEEE Transactions on Big Data*, 2019.
[16] C. Schmoll and S. Zander, "NetMate-Version 0.9. 5," ed: Germany: Fraunhofer FOKUS, 2009.
[17] Aouini Z, Kortebi A, Ghamri-Doudane Y, Cherif IL. In: Early classification of residential networks traffic using C5. 0 machine learning algorithm. IEEE; 2018. p. 46–53.
[18] Tong V, Tran HA, Souihi S, Mellouk A. In: A novel QUIC traffic classifier based on convolutional neural networks. IEEE; 2018. p. 1–6.
[19] Divina F, Gilson A, Goméz-Vela F, García Torres M, Torres JF. Stacking ensemble learning for short-term electricity consumption forecasting. Energies 2018;11(4):949.
[20] Raschka S. 28–3-2021). An ensemble-learning meta-classifier for stacking: StackingClassifier; 2020. Available: https://rasbt.github.io/mlxtend/user_guide/classifier/StackingClassifier/.
[21] Al-Qurabat AKM, Mohammed ZA, Hussein ZJJWPC. Data traffic management based on compression and MDL techniques for smart agriculture in IoT 2021;120(3):2227–58.
[22] M. Al-Qurabat and A. Kadhum, "A lightweight Huffman-based differential encoding lossless compression technique in IoT for smart agriculture," *International Journal of Computing Digital System*, 2021.
[23] Lower N, Zhan F. In: A study of ensemble methods for cyber security. IEEE; 2020. p. 1001–9.
[24] A. H. L. Gerard Drapper Gil, Mohammad Mamun, Ali A. Ghorbani. (2016, 13-9-2021). *VPN-nonVPN dataset (ISCXVPN2016)*. Available: https://www.unb.ca/cic/datasets/vpn.html.
[25] Y. Li and Y. Lu, "Multimodality Data Analysis in Information Security ETCC: Encrypted Two-Label Classification Using CNN," *Security and Communication Networks*, vol. 2021, 2021.
[26] Hopfield JJ. Artificial neural networks. IEEE Circuits Devices Mag 1988;4(5):3–10.
[27] Cortes C, Vapnik V. Support-vector networks. Machine learning 1995;20(3):273–97.
[28] Breiman L. Random forests. Machine learning 2001;45(1):5–32.
[29] Cutler A, Cutler DR, Stevens JR. Random forests. In: Ensemble machine learning. Springer; 2012. p. 157–75.
[30] Wood T. Random Forests. Available 2020;1–3). , https://deepai.org/machine-learning-glossary-and-terms/random-forest.
[31] S. Rajagopal, P. P. Kundapur, and K. S. Hareesha, "A stacking ensemble for network intrusion detection using heterogeneous datasets," *Security and Communication Networks*, vol. 2020, 2020.
[32] Chen W-H, Hsu S-H, Shen H-P. Application of SVM and ANN for intrusion detection. Comput Oper Res 2005;32(10):2617–34.
[33] Lee C, Park S. Damage classification of pipelines under water flow operation using multi-mode actuated sensing technology. Smart Mater Struct 2011;20(11):115002.
[34] Wang S-C. "Artificial neural network," in *Interdisciplinary computing in java programming*. Springer 2003:81–100.
[35] K. Jain. (2016, 28-3-2021). The Evolution and Core Concepts of Deep Learning & Neural Networks. Available: https://www.analyticsvidhya.com/blog/2016/08/evolution-core-concepts-deep-learning-neural-networks/.
[36] Tolles J, Meurer WJ. Logistic regression: relating patient characteristics to outcomes. JAMA 2016;316(5):533–4.
[37] Feng J, Xu H, Mannor S, Yan S. Robust logistic regression and classification. Advances in neural information processing systems 2014;27:253–61.
[38] Sharma P. Classification – Logistic Regression. Available 2018;28–3). , https://www.tech-quantum.com/classification-logistic-regression/.
[39] Almomani A. Fast-flux hunter: a system for filtering online fast-flux botnet. Neural Comput Appl 2018;29(7):483–93.
[40] Almomani A, Gupta BB, Atawneh S, Meulenberg A, Almomani E. A survey of phishing email filtering techniques. IEEE Commun Surv Tutorials 2013;15(4):2070–90.
[41] Lotfollahi M, Jafari Siavoshani M, R. Shirali Hossein Zade, and M. J. S. C. Saberian. Deep packet: A novel approach for encrypted traffic classification using deep learning 2020;24:1999–2012.
[42] Izadi S, Ahmadi M, A. J. J. o. N. Rajabzadeh, and S. Management. "Network traffic classification using deep learning networks and. Bayesian data fusion," 2022;30(2):1–21.

Dr. Ammar Almomani received his Ph.D. from University Sains Malaysia (USM) in 2013. He is the author of over 75 research papers in renowned International Journals and Conferences, including IEEE, Elsevier, ACM, Springer, and Inderscience, with countless of international awards, He has toured various nations in presenting his valued research work and has revised 10s Journals in IEEE, Springer, Wiley, and Taylor & Francis. He has lectured for 16 years, covering over 40 subjects in computer science, networks, cybersecurity, and programming language. He holds numerous international certificates and has contributed to various projects and specialized scientific courses. He has been exploring cybersecurity, advanced Internet security, and monitoring as an ardent researcher. Dr. Ammar Almomani is currently serving the post of senior lecturer at Al- Balqa Applied University. Currently, he leads the research and innovation department in SKYLINE university college-SHARJAH-UAE. Link: https://scholar.google.com/citations?user=d_tRtPkAAAAJ&hl=en.