Ain Shams Engineering Journal 15 (2024) 102361

Contents lists available at ScienceDirect

Ain Shams Engineering Journal

journal homepage: www.sciencedirect.com

Encrypted network traffic classification based on machine learning

Reham T. Elmaghraby [a,b,*], Nada M. Abdel Aziem [b], Mohammed A. Sobh [a], Ayman M. Bahaa-Eldin [c]

[a] Faculty of Engineering, Ain Shams University Cairo, Egypt
[b] Arab Academy for Science, Technology and Maritime Transport-Arab League, Egypt
[c] Misr International University, On Leave from Ain Shams University, Egypt

ARTICLE INFO

Article history:
Received 19 April 2023
Revised 22 May 2023
Accepted 18 June 2023
Available online 06 July 2023

ABSTRACT

Encrypted traffic is an essential part of maintaining the security and privacy of data transmission. It plays an important role in keeping our networks secure by preventing attackers from intercepting confidential information, which they may access without authorization; However, its effectiveness relies heavily on accurate classification techniques being applied correctly, so we can differentiate between legitimate users' activities versus those attempting malicious activity within the networks' boundaries. Encrypted network traffic is becoming increasingly common in modern communication systems, presenting a challenge for effective network management and security. To address this challenge, machine learning models have been employed to classify encrypted traffic but with limited success due to the lack of clear visibility into packet contents and an inability to inspect their content. For the sake of tackling this issue, more effective research has begun on developing machine learning models for classifying encrypted payloads without relying on inspecting their contents directly. This research will investigate how features like packet length, time stamps or transport layer security (TLS) and encrypted payload information can be used as input features when attempting classification tasks, instead of analyzing unencrypted content directly from packets themselves which would otherwise be impossible given the current technology constraints. The evaluation process will focus on assessing different model architectures, as well as feature selection techniques that yield improved results over the existing approaches. In this paper, we proposed three approaches to identify encrypted traffic and classify different applications such as browsing, VOIP, file transfer and video streaming. The first two techniques consist of two stages: the first stage is either a neural network or a bi-directional LSTM, and the second stage is a selection of different classification techniques, namely Random Forest, Support vector machine, Linear regression, and K-nearest neighbor. The final result is achieved using an ensemble voting technique. As for the third technique, the network packets are grouped together by Source IP, destination IP and session time before feeding them into three different combinations of LSTM networks; either coupled with convolution 1D or 2D layers, or without. Like the first two techniques, the final result is achieved by means of ensemble voting. Through extensive comparison between the three approaches, The first approach yielded the highest accuracy. However, the performance of the second and third techniques in terms of time complexity was superior. The achieved accuracies were 96.8%, 95.2% and 96.5% for the proposed techniques, respectively.

1. Introduction

More technologies are developed due to the rapid increase in the needs of societies. IOT [1] and network traffic classifications are examples of these innovative technologies that researchers focus on them.

Network traffic classification and recognition is an important task for maintaining cyberspace security. With the rapid development of the Internet, network applications and protocols emerge in an endless stream, making the types of network traffic more complex and diverse [2]. This poses certain obstacles to effective

* Corresponding author.
E-mail address: rehamelmaghraby@aast.edu (R.T. Elmaghraby).
Peer review under responsibility of Ain Shams University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.asej.2023.102361
2090-4479/

management of such networks. To address this issue, advanced techniques are needed to accurately classify different types of network traffic so that appropriate measures can be taken for managing it effectively.

Advanced technologies used for classifying various kinds of network traffic have been developed over time with improved level of accuracies, as well as better classification into distinct classes [3].

Network traffic classification has become more challenging because of the emergence of secure networks that have encrypted payloads, across all digital platforms today [4]. These security features provide a robust foundation upon which other systems can be established in order to guarantee reliable performance while minimizing any potential threats posed by malicious adversaries. Therefore, organizations are increasingly investing in this technology to ensure optimal safety conditions within their respective infrastructures, which renders the classification of such traffic more challenging. On the other hand, these security features mask the configuration of these networks and make it harder to provide different services, such as firewall, QoS, Access control, etc.

To address this issue, researchers have developed techniques for analyzing encrypted traffic without decrypting it. These techniques vary from statistical analysis and machine learning to protocol-specific analysis, which allow them to classify different types of encrypted communication based on different parameters such as packet size or frequency patterns [4].

Classifying encrypted traffic is a complex process that requires a thorough investigation and deep analysis of various methods. Statistical analysis, machine learning, and deep learning are three common methods used to classify encrypted traffic. When it comes to the accurate classification of encrypted network traffics, each of the aforementioned method has its own advantages and disadvantages.

Statistical analysis involves analyzing the statistical properties of the data to identify various patterns that can be used for classification purposes. This method assumes that different types of traffic have different statistical properties (e.g. entropy) which can be identified by inspecting packet size or other features associated with the different types of communication protocols being analyzed [5].

Supervised machine learning techniques [6] use labeled datasets consisting of both plaintext (non-encrypted) and ciphertext (encrypted) samples that are fed into a trained algorithm to distinguish the different characteristics between them, for classification purposes. This allows for greater accuracy compared with the aforementioned traditional statistical approaches, especially since it does not require prior knowledge about the employed encryption algorithms.

Deep learning [7] is a powerful tool for identifying patterns and classifying traffic. This method relies on the assumption that deep neural networks can learn complex patterns in data. Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) are two other types of deep learning architectures that are commonly used, according to the literature, to classify encrypted traffic. These techniques demonstrated the superiority of these methods in recognizing hidden patterns within data sets. Traditional methods may fail or struggle due to lack of information or complexity involved in encryption algorithms used by attackers/hackers.

LSTM [8] is a type of recurrent neural network designed to handle sequential data, such as time series analysis, speech recognition, and natural language processing tasks, and CNNs [9] are another type of neural network specifically designed to work with image [10] or video [11] data by detecting spatial features in the input information.

Overall, these techniques demonstrate how effective deep learning techniques can be at accurately recognizing patterns within different types of online activities, such as web browsing, email exchange, video streaming, etc. This allows us to better understand user actions while also helping protect against malicious activities taking place online.

This research might introduce an ensemble voting technique between combination of existing machine learning algorithms, and capture new dataset that includes a new features and network conditions than previous datasets. Alternatively, the research might present a comprehensive evaluation of existing techniques for encrypted traffic classification, highlighting their strengths and weaknesses and proposing new directions for future research.

The objective of this research is to provide reliable and efficient methods for encrypted payload traffic classification that can enhance network security and management.

This paper is organized into five sections that highlight our contribution. The first section presents an introduction to secured network traffic and the most common challenges in classifying them. It also presents different techniques that are used to overcome prevalent issues and successfully classify them. In the second section, a literature review that covers the history of machine learning is presented. In the third section, data preprocessing methods are illustrated by means data flow diagrams that provide an overview of the procedures of classifying encrypted traffic. The proposed algorithm is explained in the fourth section. Finally, the experimental results are discussed in the fifth section.

2. Literature survey

In [12] K. Ali, A. Tariq, and H. Abbas present an overview of various deep learning techniques used for encrypted network traffic classification. They also propose a new architecture that uses convolutional neural networks (CNNs) to classify encrypted traffic. The proposed architecture achieved a classification accuracy of 97% on the benchmark dataset.

Lu et al. [13] presented a combination of Bidirectional LSTM and RF achieved an accuracy of 96.7%.

In Zhou et al. [14] presented a combination of Bidirectional LSTM and KNN achieved an accuracy of 96.4%.

Wang et al. [15] using a combination of LSTM and CNN2D achieved an accuracy of 92.2%.

In [16] authors S. Ali, M. Arfeen, and S. Rehman explore the use of machine learning techniques for encrypted traffic classification. They compare the performance of various models such as decision trees, random forests, and support vector machines (SVMs). Their results showed that SVMs outperformed other models with an accuracy of 99%.

In [17] authors A. Al Rawi, A. Al Saffar, and M. Al-Adhami propose a hybrid approach that uses statistical analysis and machine learning techniques for encrypted traffic classification. Their approach achieved an accuracy of 98% on the benchmark dataset.

In [18] authors M. Amin, X. Liu, and A. Al-Dhelaan investigate the use of deep learning techniques for encrypted traffic classification. They evaluate the performance of several deep learning models, including CNNs and long short-term memory (LSTM) networks. Their results show that the LSTM network outperforms other models with an accuracy of 99.5%.

In [19] F. Hussain, M. Ismail, and A. Khan explore the use of machine learning techniques for encrypted traffic classification. They compare the performance of various models such as decision trees, k-nearest neighbors, and SVMs. Their results show that the SVM model outperforms other models with an accuracy of 97%.

Zhaolei et al. [20] proposed an algorithm for encrypted traffic called Byte Feature Convolution Network (BFCN) based on BERT and CNN. This model consists of two modules: the packet encoder module and the CNN module, which takes global traffic features through its attention mechanism, as well as byte-level local features in traffic through convolutional operations respectively. The authors used the ISCX dataset to evaluate their model's performance, achieving a high recall, precision accuracy and F1 score compared to other deep packet models using 1D CNN only.

Similarly Xinyi Hu et al. [21], presented a CLD Net model that combines convolutional neural network with LSTM layers in order to enhance classification performance over unknown encrypted network traffics such as VPNs or Skype applications etc..The authors also tested their approach against ISCX public dataset obtaining 98% accuracy for VPN detection while 92% accuracy was achieved when recognizing Skype's specific application layer protocol.

Wei Wang et al. [22] proposed an end-to-end encrypted traffic using 1D convolution neural networks. This method merges feature selection, feature extraction and classifier into a single framework, allowing for more efficient classification of network traffic data. The paper uses the public ISCX non-VPN – VPN dataset which includes multiple types of traffic such as email, chat, streaming VOIP file transfer and P2P protocols. The experiments conducted in this paper concluded that the four methods performed better than existing state-of-the art methods when it comes to encryption detection accuracy rate with false positives being reduced by up to 17%. Furthermore, these results were obtained without any preprocessing or manual feature engineering steps which further enhances its efficiency as well as cost effectiveness due to its automated nature. Moreover, this model is able to detect different types of encrypted flows simultaneously while still maintaining high accuracy rates across all categories making it suitable for real world applications where various type of flows are expected concurrently.

Kun Zhoe [23] et al. proposed a method that combined entropy estimation with artificial neural networks to evaluate the performances of random forests, support vector machine, logistic regression and naïve Bayes for traffic classification. Network traffic is classified as plaintext or encrypted using entropy estimation on a dataset from the Canadian Institute for cybersecurity. The results showed an improved accuracy compared to traditional methods when using this combined approach.

Lulu Guo et al. [24] further developed two models based on deep learning; convolutional autoencoders (CAE) and convolutional neural networks (CNN), which are used to classify non-VPN and VPN traffic in real time while also identifying VPN encrypted traffic accurately. To test their methods, they utilized public ISCXVPN 2016 data set as input into both CAE model and CNN model respectively. The overall identification accuracy rate was 99.8% for CAE model and 92.92% for CNN Model, demonstrating that these approaches were effective at classifying network traffics into different categories accurately in real time applications.

Madushi H. Pathmaperuma et al. [25] recently developed a convolutional neural network (CNN) to detect user activities on mobile applications such as YouTube, Facebook, Messenger and Gmail. This model uses a time window-based approach to divide the encrypted traffic flow activity into segments for more accurate analysis of user behavior on these apps. CNN can differentiate between trained and untrained in-app activities beyond what is known about the type of activity taking place within an app itself. After filtering out unknown traffic, this method has achieved an average accuracy rate of 88% while achieving 92% classification accuracy after filtering is complete - making it one of the most reliable methods available for detecting user behaviors across multiple mobile applications simultaneously from encrypted data sources.

3. Data flow and preprocessing

3.1. Data flow

Encrypted traffic classification is an important tool for network security and monitoring. By analyzing encrypted network traffic, organizations can determine the type of application or service being used on their networks. This helps them to better understand how their networks are being utilized, identify potential threats, and take appropriate action if necessary.

The data flow of encrypted traffic classification begins with capturing the encrypted packets using a capture tool such as Wireshark. The captured packets are then examined to identify any protocol or port numbers associated with encryption services like HTTPS (port 443) and SSH (port 22). Once identified, various techniques such as deep packet inspection statistical analysis machine learning algorithms signature-based matching can be used to classify the type of application or service that is in use on a given network segment. Finally, results from this process can be reported for further analysis and may even trigger automated actions such as blocking malicious activity when necessary.

3.2. Data preprocessing

Data preprocessing is an important step in any machine learning algorithm. It ensures that the data set is properly formatted and ready for analysis. Before entering a data set into proposed algorithms, it must first go through a process of data preprocessing to ensure accuracy and reliability of the results.

The first step in this process involves dividing the dataset into two parts: packets part and information part. The encrypted packets are then converted from hexadecimal form to decimal form so that they can be used by machine learning networks more easily. In addition, some features such as flags style (UDP or TCP) as well as source IP address and destination IP address may consist of strings which need to be categorized accordingly before further processing can take place. Furthermore, these addresses must also be converted from string format to integer values so they can be used by machines effectively during calculations later on down the line.. Finally, both parts are concatenated together along with four class types which have been added after categorization has taken place (as shown in Fig. 1).

In conclusion, proper data preprocessing is essential when using proposed algorithms for machine learning purposes since it helps prepare datasets for further analysis while ensuring accuracy throughout all stages involved in its preparation stage. This allows us to gain better insights about our datasets without having any issues related with incorrect formats or unreliable results due too poor initial formatting done beforehand.

4. Proposed model

This section will discuss the details of the proposed techniques and how they are designed for achieving maximum effectiveness. It is divided into three techniques that each consist of multiple stages. Fig. 2 shows the flowchart of the proposed model with the three techniques.

4.1. First technique

It is an effective method for preprocessing data before entering it into a machine-learning model. The first stage in this technique

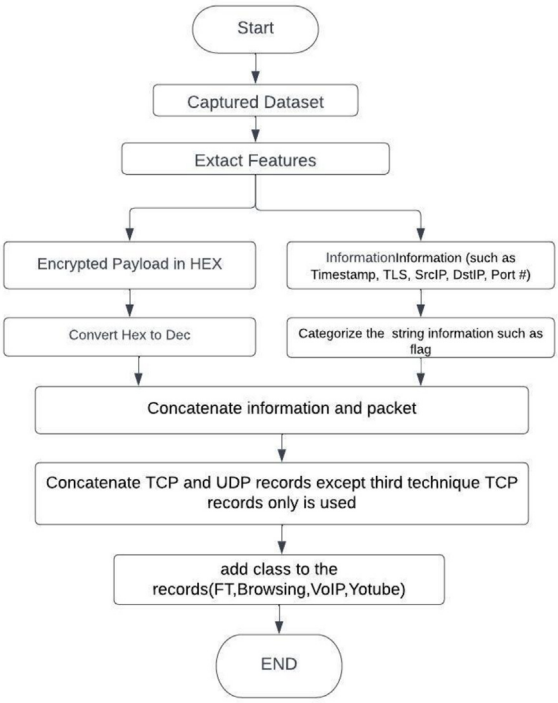

Fig. 1. Data preprocessing.

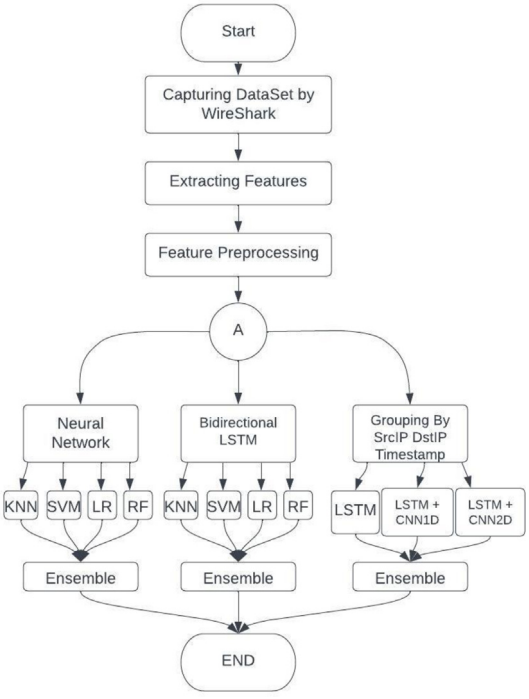

Fig. 2. Proposed model flow chart.

is to shuffle the table, as sequence does not play a role during training. The table then needs to be divided into two parts: encrypted packets and information. The encrypted packet part entered into a neural network consisting of four layers- an input layer, two hidden layers and an output layer of four class types. The last layer before the output layer of the neural network is captured. Then captured layer is concatenated with the information part before it is entered into four different classification algorithms: Random Forest (RF), Logistic Regression (LR), K-Nearest Neighbor (KNN) and Support Vector Machine (SVM). In the last stage, the result is achieved by means of ensemble voting as shown in Fig. 3.

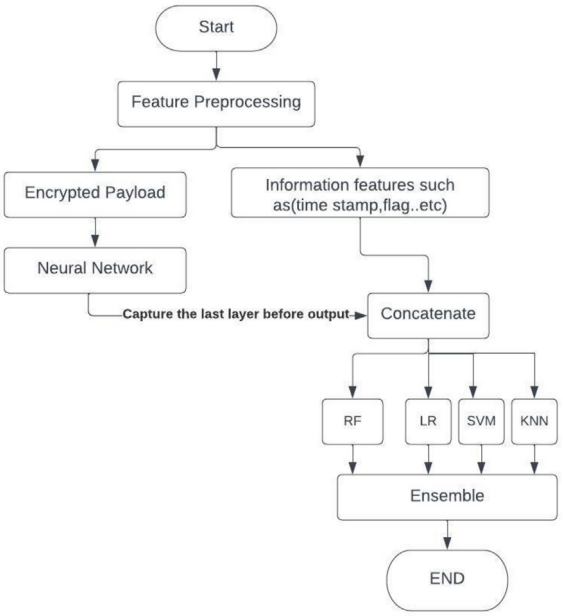

Fig. 3. First Technique: Ensemble of neural Network and Different Classifications.

4.2. Second technique

It follows largely similar stages of first technique but differs slightly by replacing Bidirectional-LSTM instead of neural network in the first stage. The last layer before the output of a neural network is an important part of the process. This layer captures a lot of information about what it has learned from training data, which can be used to make predictions. By concatenating this captured layer with other information parts, four different classification algorithms - Random Forest (RF), Logistic Regression (LR), K-Nearest Neighbor (KNN) and Support Vector Machine (SVM) – can be used for further analysis. The final result is achieved by

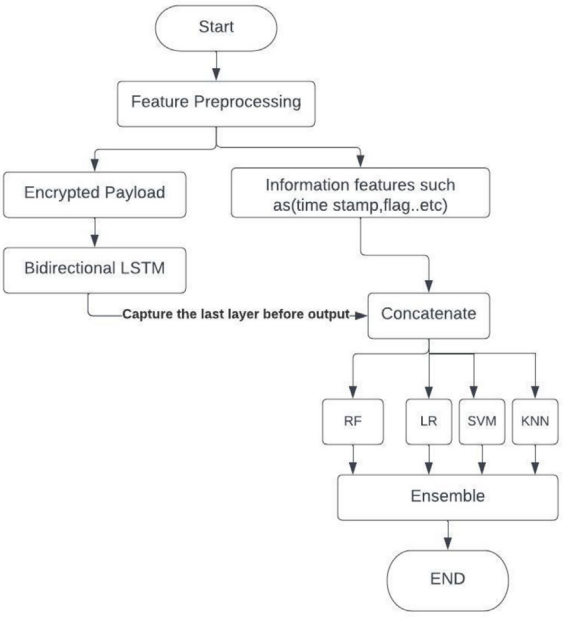

Fig. 4. Second Technique: Ensemble of Bidirectional LSTM and Different Classifications.

means of ensemble voting, which combines all four models' outputs into one overall prediction or decision outcome, as shown in Fig. 4. This technique has been found useful when dealing with large datasets or complex problems where traditional methods may fail due to their limited capacity or capability of processing data effectively. Additionally, using Long Short Term Memory networks can lead more accurate results than conventional models because they have memory cells that allow them store previous inputs better which helps them understand patterns from experience thus leading more reliable predictions.

4.3. Third technique

In the last technique, data preprocessing tables are processed in a different way, TCP records are selected only without using UDP records due to its high reliability and containing time stamp feature that indicates the begun and end of each session. Packets and information are divided into groups depending on source IP, destination IP, time stamp and class type. For each group of packets, the maximum value (max), minimum value (min) and mean standard deviation (SD) is calculated. This grouped information along with the grouped encrypted packets is then concatenated to feed into three types of machine learning networks - Bidirectional LSTM consisting of two layers; CNN 1D followed by an LSTM layer; and CNN 2D followed by an LSTM layer.

To ensure that all groups have equal sized packet sizes for training purposes padding will be applied to reshape the packets accordingly before entering them into these networks for further processing. The concatenated features will also be shuffled beforehand so as to create randomness in order to prevent overfitting during the training phase. The final result is achieved by means of ensemble voting, which combines three network's outputs into one overall prediction or decision outcome as shown in Fig. 5.

All results obtained from these tests can then help us assess how well our network performs against varied input datasets & allow us make necessary changes if needed so that we obtain optimal performance from our model at any given point in time when deployed on production environment.

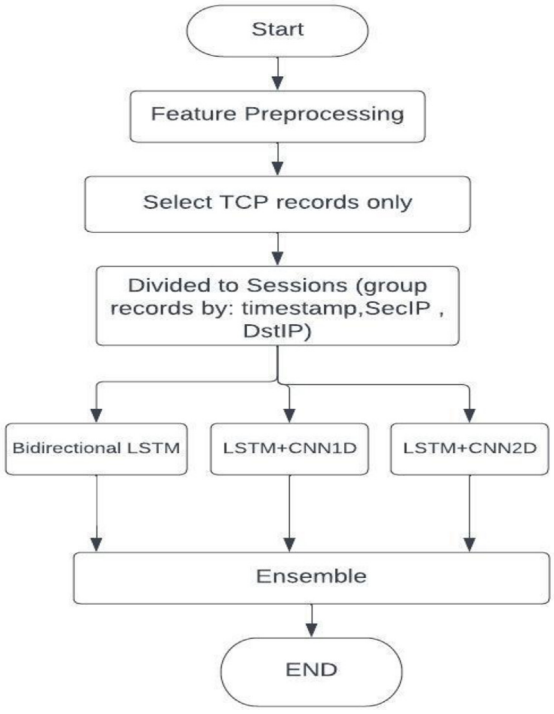

Fig. 5. Third Technique: Ensemble of LSTM and LSTM and conv1D and LSTM and Conv2D.

5. Experimental and results

5.1. DataSet description

Encrypted traffic classification datasets are an important tool in training machine learning models to automatically classify encrypted network traffic. These datasets typically consist of network packet captures, where each packet includes the payload data and metadata such as source and destination IP addresses, protocol, port number, and time stamp. In order to accurately train these models on this data set there must be a variety of features included that can help identify the different types of packets being sent over the network [26].

Common features used in encrypted traffic classification datasets include Protocols (such as TCP/IP or UDP), Port Numbers (which designate which application is sending or receiving a particular message), Packet Lengths (in bytes), Time Stamp for when each packet was captured, Source/Destination IP Addresses so that messages can be routed properly across networks; TLS Version information for any packets using Transport Layer Security protocols; and finally Class Labels which indicate what type of content is contained within a given packet.

Dataset captured divided into four classes with total number of 33,286 packets could have 29,957 packets for training and 3329 for testing number of packets in each class shown in Table 1.

In the above table, we can observe that there are almost no UDP Packets and only TCP packets when it comes to file transfer and VOIP. This is because TCP provides a reliable delivery mechanism for data packets by ensuring they are delivered in the same order as they were sent. The downside of this approach, however, is that it requires additional header information which leads to higher overhead compared to UDP.

Overall, we can see how employing TCP instead of UDP makes sense especially when dealing with critical applications like File Transfer And VoIP where reliability takes precedence over efficiency. By utilizing TCP's ability To ensure Data Is Delivered In The Same Order As It Was Sent, We Can Ensure That All Necessary Information Is Transferred Correctly Without Any Loss Of Quality Or Accuracy During Transmission Across Networks. That's why in approach 3 while grouping the packets and information as mentioned TCP file only are used.

5.2. Evaluation

Evaluating and validating two-class problems typically involves the formulation of four cases: true positive (TP), false positive (FP), true negative (TN) and false negative (FN). TP is when a given sample that should be classified as a positive result is correctly identified as such, whereas FP occurs when an expectedly negative sample is incorrectly labeled as being positive. TN corresponds to the successful identification of a negatively labeled sample, while

Table 1
Dataset Classes and Number of Packets.

Label in Confusion Matrix	Class	Number of packets (TCP)	Number of packets (UDP)
0.0	Browsing	8635	3040
1.0	File Transfer (FT)	13,599	34
2.0	Voice over IP (VOIP)	1173	None
3.0	YouTube	1573	5232

FN refers to mislabeling an expectedly positively classified example.

To assess performance in these scenarios, four metrics are utilized: accuracy, precision, recall and F1 score. Accuracy [27] measures how many samples were correctly classified out of all tested examples; it can be used for both two-class problem evaluation or multi class problem assessment with macro accuracy representing the proportion correct relative to total samples evaluated as shown in equation (1):

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Precision [28] evaluates what fraction of predicted positives were correct; this metric works best for imbalanced datasets where one class dominates another numerically speaking as shown in equation (2):

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Recall [29] looks at what percentage of actual positives was successfully identified by our model; this measure helps us understand how well our algorithm performs on more balanced datasets with equal numbers from each classification group present in training data sets as shown in equation (3).

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

Finally, F1 score [30] combines both precision & recall into one single value which gives us insight into overall performance across multiple classes simultaneously as shown in equation (4):

$$F1 - score = 2x\frac{recall * precesion}{recall + precision} = \frac{2TP}{2TP + FP + FN} \tag{4}$$

5.3. Results and analysis

The results of our analysis regarding deep learning algorithms for classifying encrypted traffic are promising. Our algorithm achieved an average accuracy of 95%, indicating that its ability to classify the underlying applications even when the network traffic is encrypted. We evaluated its performance using a dataset containing 33,286 packets from various applications such as web browsing, file transfer and VoIP. By analyzing the data, we were able to determine that our algorithm had a good precision of 0.96, a recall of 0.94 and an F1 score of 0.95. This suggests that deep learning can be used to successfully classify different types of network traffic even when they are encrypted.

The confusion matrices of the four classification models for the first technique are shown in Figs. 6, 7, 8 and 9 and the performance metrics are presented in Table 2. The matrices show the number of true positives, true negatives, false positives, and false negatives for each class in the classification.

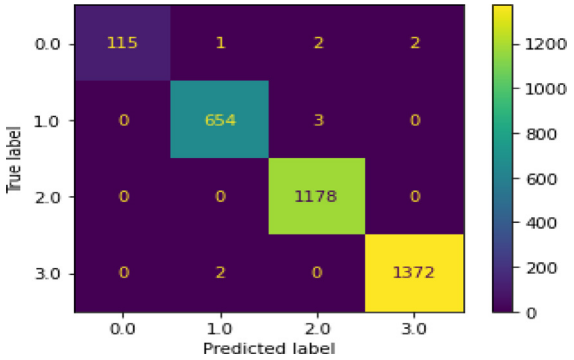

Fig. 6. Confusion Matrix RF first technique.

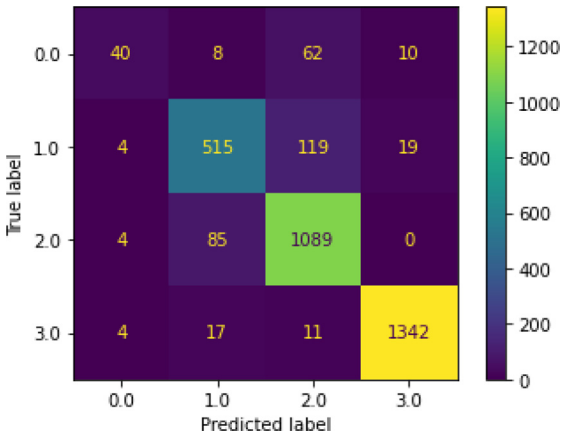

Fig. 7. Confusion Matrix Logistic Regression first technique.

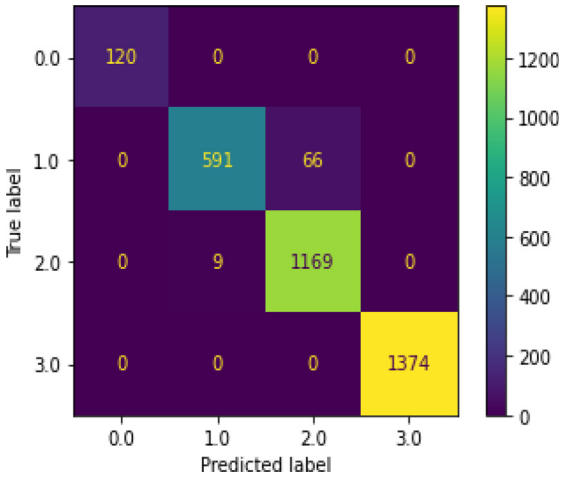

Fig. 8. Confusion Matrix KNN first technique.

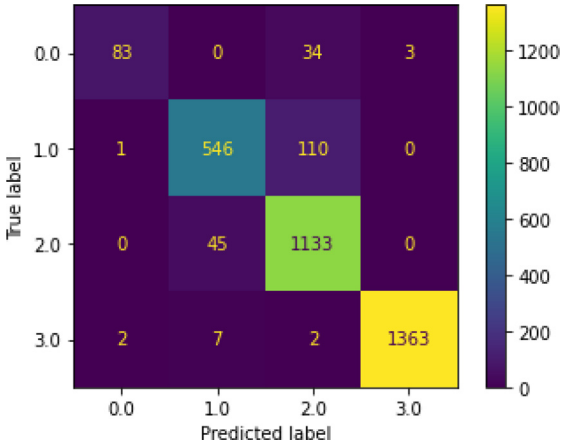

Fig. 9. Confusion Matrix SVM first technique.

Table 2 demonstrates the effectiveness of the suggested machine learning techniques for predicting the four different classes that represent different network applications. The table shows that the random forest model outperformed all other models with an accuracy score of 99.6%, along with average values of

Table 2
First Technique Results Metrics.

Classification Technique	Accuracy	Class	precision	recall	F1-Score
RF	99.60%	Browsing	1	0.94	0.97
		FT	1	0.99	1
		VOIP	0.99	1	1
		YouTube	1	1	1
Average			1	0.96	0.98
LR	90.20%	Browsing	0.77	0.33	0.47
		FT	0.82	0.78	0.8
		VOIP	0.85	0.92	0.89
		YouTube	0.98	0.98	0.98
Average			0.85	0.75	0.78
KNN	97.60%	Browsing	1	1	1
		FT	0.98	0.9	0.94
		VOIP	0.95	0.99	0.97
		YouTube	1	1	1
Average			0.98	0.97	0.97
SVM	94%	Browsing	0.97	0.63	0.81
		FT	0.91	0.83	0.87
		VOIP	0.89	0.96	0.92
		YouTube	1	0.99	0.99
Average			0.94	0.86	0.89

1.00, 0.96 and 0.98 for precision, recall and F1 score, respectively. In addition to the results achieved from Random Forest (RF), Support Vector Machine (SVM) and K-Nearest Neighbor (KNN) also performed well, with accuracy scores of 94% and 97.6%, respectively. As for the Logistic Regression (LR) Classifier, it yielded the lowest accuracy of 90.2%, along with average values for the precision, recall and F1 of 0.85, 0.75 and 0.78, respectively. Despite its lower performance, it is worth noting that LR still provided some useful insights into our data set which can be further explored in future experiments or research projects related to network classification tasks using ML algorithms. Finally, after performing the ensemble voting, the calculated overall accuracy was 98%.

As for the second classification technique, employing the bi-directional LSTM, its performance metrics are shown in Table 3, and the confusion matrices are illustrated in Figs. 10, 11, 12 and 13. The results show that from among all techniques, the KNN technique was able to achieve the highest accuracy of 100%, along with an average precision, recall and F1-score values of 1.00, 1.00 and 1.00, respectively. The results reflect its superiority over the other techniques such as Support Vector Machine (SVM), Random Forest (RF) and Logistic Regression (LR) which had accuracies of 85.8%, 97.5% and 72%, respectively. The performance metrics of the LR technique had a precision, recall and F1-score of 0.49, 0.48 and 0.46, respectively. The results of the ensemble voting stage gave an accuracy of 95.2%.

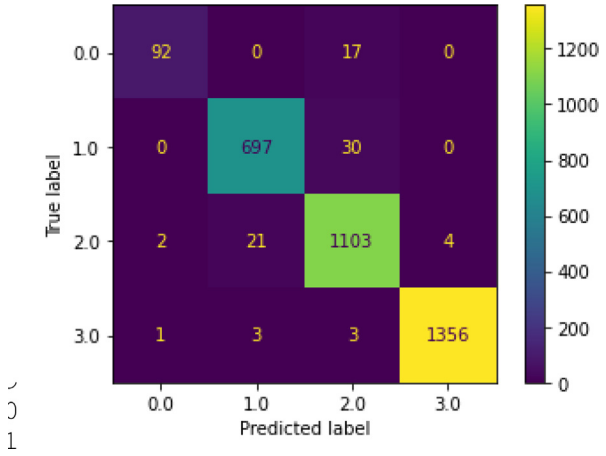

Fig. 10. Confusion Matrix RF Second technique.

Table 3
Second Technique Results Metrics.

Classify network	Accuracy	Class	Precision	recall	F1-Score
RF	97.5%	Browsing	0.97	0.84	0.90
		FT	0.97	0.96	0.96
		VOIP	0.96	0.98	0.97
		YouTube	0.97	0.84	0.90
Average			0.97	0.94	0.95
LR	72%	Browsing	0.00	0.00	0.00
		FT	0.53	0.16	0.25
		VOIP	0.75	0.79	0.77
		YouTube	0.71	1.00	0.83
Average			0.49	0.48	0.46
KNN	100%	Browsing	1.00	1.00	1.00
		FT	1.00	1.00	1.00
		VOIP	1.00	1.00	1.00
		YouTube	1.00	1.00	1.00
Average			1	1	1
SVM	85.8%	Browsing	0.00	0.00	0.00
		FT	0.89	0.65	0.75
		VOIP	0.74	0.91	0.81
		YouTube	0.96	0.99	0.98
Average			0.647	0.637	0.63

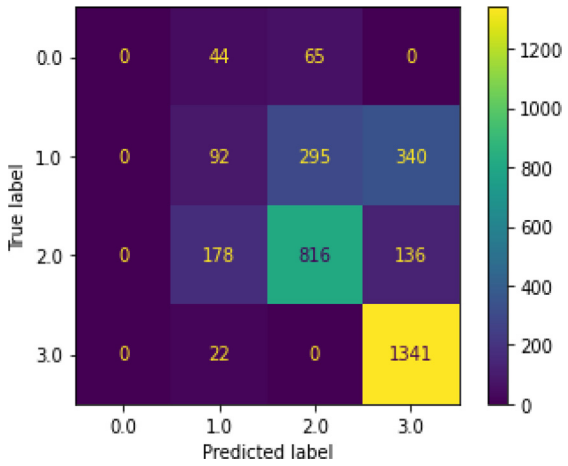

Fig. 11. Confusion Matrix Logistic Regression Second technique.

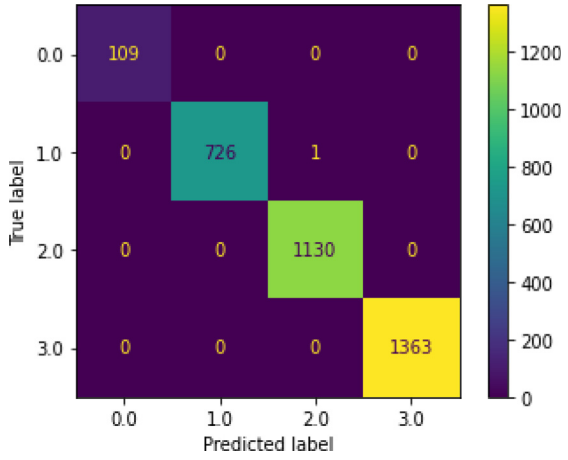

Fig. 12. Confusion Matrix KNN Second technique.

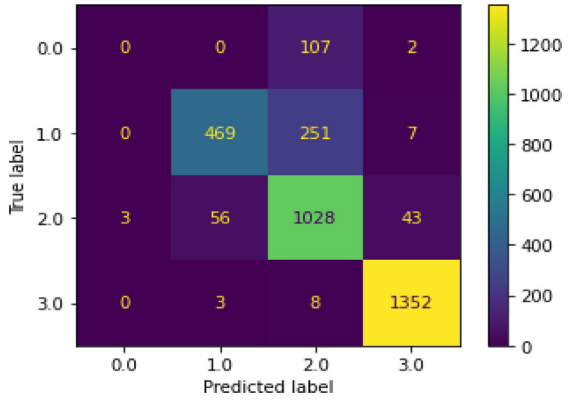

Fig. 13. Confusion Matrix SVM Second technique.

Finally, for the third technique, the dataset was divided to different sessions by grouping the records depending on various features such as time stamp, SrcIP and DstIP as mentioned in the section above. This technique uses a combination of different deep learning networks to classify traffic at various levels, such as Bidirectional LSTM, and LSTM concatenated with CNN-2D and CNN-1D. The results obtained from these methods had accuracies of 94%, 98% and 71%, respectively. The confusion matrices for this technique are shown in Figs. 14, 15 and 16. Moreover, these techniques were also able to accurately classify web browsing and video streaming applications with average precision and recall scores of 0.97and 0.99, respectively. The results confirm that they are also very effective at classifying traffics on application level.

On the other hand, when using a combination of an LSTM network along with conv1D, it was observed that it could accurately classify VoIP application with precision, recall and F1-score of 0.75, 0.8 and 0.78 respectively. However, its performance dropped significantly when trying to identify file transfer or web browsing activities – resulting in much lower accuracy rates than what was previously achieved through either Bidirectional LSTMs or those combined with CNN2D as shown in Table 4. For this technique, the ensemble voting resulted in an overall accuracy of 97%. Ultimately, it can be concluded that all three techniques have their own advantages according to the targeted application (e.g. VOIP, Web Browsing, etc), therefore, selection of the appropriate technique must be done carefully before employing it in any project.

Table 5 presented the comparison between the accuracy of some techniques in the proposed model with some previous related works.

6. Discussion

This section discussed the reasons for using proposed machine learning techniques and the importance of combining them with classification networks.

First, the neural network and Bidirectional-LSTM network used to learn high-level features from the encrypted traffic data, while the SVM or RF or KNN or LR used to make the final classification decision based on these features. The combination of LSTM with other machine learning models such as SVM, KNN, LR, and RF has been shown to be effective for encrypted traffic classification. Encrypted traffic datasets often suffer from class imbalance, with some classes having many more samples than others. This can make it difficult for a single model to learn to distinguish between the different classes. By combining multiple models, it can balance the strengths and weaknesses of each model to better handle this imbalance. It also improves robustness, by combining multiple models, it can be able to handle changes in network conditions or the introduction of new applications and protocols.

The use of Bidirectional LSTMs alone in encrypted traffic classification has several advantages, including the ability to capture long-term dependencies, robustness to noise and variability, ability to handle variable-length input sequences, automatic feature extraction, and ability to handle complex and nonlinear relation-

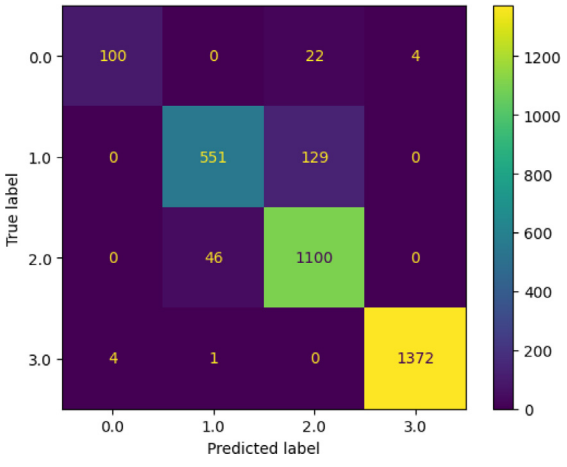

Fig. 14. Confusion Matrix Bidirectional LSTM.

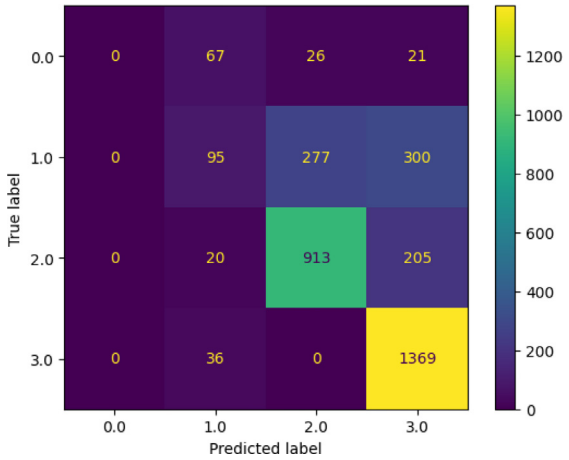

Fig. 15. Confusion Matrix LSTM and CNN-1D.

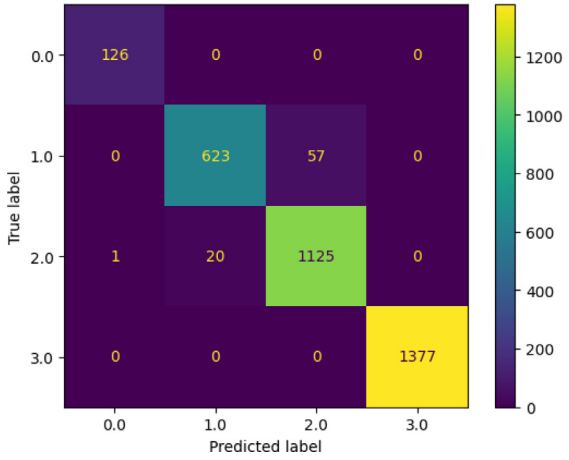

Fig. 16. Confusion Matrix LSTM CNN −2D.

ships. They can also be combined with other types of neural networks, such as CNNs, to further improve their performance. But it may also be prone to overfitting, especially if the training data is imbalanced or noisy.

LSTM combined with CNN2D In the context of encrypted traffic classification, it can be used to capture the spatial patterns in the encrypted traffic, such as packet size, packet timing, and packet direction, which can be indicative of different applications and protocols. As in the third technique the dataset records are divided into sessions depending on packet timing (timestamp) and packet direction (SrcIP and DstIP). On the other hand, when LSTM combined with CNN1D may not be as effective for encrypted traffic classification as other architectures, such as LSTM combined with CNN2D.

At Last, more than machine learning are tried on the captured dataset and the chosen techniques achieved higher accuracy of classifying the network applications.

Table 5
Comparison of accuracy between proposed model and previous related work.

	Technique	Accuracy	Proposed model
[11]	Bidirectional LSTM combined with RF	96.7%	97.5%
[12]	Bidirectional LSTM combined with KNN	96.4%	100%
[13]	LSTM combined with CNN2D	92.2%	98%

7. Conclusion

This research has demonstrated the feasibility of using machine learning models for encrypted payload traffic classification. It has shown that various features, such as packet length, time stamps, TLS information and encrypted payload can effectively contribute to accurate classification. This research results combining different deep learning techniques such as neural network and LSTM with other traditional algorithms like RF, KNN LR and SVM, and combining LSTM with CNN can significantly improve the accuracy of classification models.

Some proposed models require minimal computational resources while achieving high accuracy rates compared to state of art methods. Furthermore, results indicate that by taking into account different features able to achieve better performance.

The results of this research have shown that the proposed models achieved high accuracy rates in classifying encrypted payload traffic. The ensemble of the first technique achieved an accuracy rate of 96.7%, while the second and third techniques both achieved 95.2% and 96% respectively. This is a significant finding as it provides new opportunities to improve network security and management by developing more accurate classification systems for this type of data traffic.

In addition, this research has also provided valuable insights into combining different machine learning techniques to develop better classification systems for encrypted payloads with higher accuracy rates than traditional. Further investigation should be made into using other features that could help strengthen these types of networks even further, such as incorporating deep learning techniques or exploring different feature selection strategies

Table 4
Third Technique Results Metrics.

Classify network	Accuracy	Class	precision	recall	F1-Score
Bidirectional LSTM	94%	Browsing	0.96	0.79	0.87
		FT	0.92	0.81	0.86
		VOIP	0.88	0.96	0.92
		YouTube	1	1	1
Average			0.94	0.89	0.912
LSTM + CNN1D	71%	Browsing	0.00	0.00	0.00
		FT	0.44	0.14	0.21
		VOIP	0.75	0.80	0.78
		YouTube	0.72	0.97	0.83
Average			0.47	0.477	0.455
LSTM + CNN2D	98%	Browsing	0.99	1.00	1.00
		FT	0.97	0.92	0.94
		VOIP	0.95	0.98	0.97
		YouTube	1.00	1.00	1.00
Average			0.978	0.975	0.977

which may lead to improved performance metrics compared to existing solutions.

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.

References

[1] Tahmasseby S. The implementation of smart mobility for smart cities: a case study in Qatar. J Urban Technol 2019;26(3):87–103.
[2] Dainotti A, Amato A, Pescape A, Ventre G. Characterization of encrypted and VPN traffic using time-related features. Comput Netw 2014;64:19–31.
[3] Wang S, Zhang Y, Jiang X, Li Y, Li G. Deep flow: deep learning-based encrypted traffic classification with internet of things applications. IEEE Trans Ind Inf 2019;15(2):764–72.
[4] Singh K, Kumar N, Garg S. Encrypted traffic classification using deep packet inspection and machine learning. J Netw Comput Appl 2019;131:1–14.
[5] Alshammari R, Zincir-Heywood AN, Heywood MI. Statistical analysis of encrypted network traffic for the purpose of classification. J Netw Comput Appl 2017;84:11–22.
[6] Wang J, Liu Y, Chen Z, Yang J. Encrypted traffic classification using machine learning techniques. IEEE Access 2017;5:21017–27.
[7] Chen X, Gao X, Chen J, Zhang Y. Encrypted traffic classification using deep learning techniques. IEEE Access 2018;6:52145–55.
[8] Zou Y, Zhang H, Zhao W. Encrypted traffic classification based on long short-term memory. J Ambient Intell Hum Comput 2020;11(12):5171–81.
[9] Wu J, Sun X, Zhang Y, Huang T. Encrypted traffic classification based on convolutional neural network. IEEE Access 2020;8:171304–13.
[10] Kandel J, Avraham T, Cohen-Or D. Brightness as an augmentation technique for image classification. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops 2016;2016:9–16. doi: https://doi.org/10.1109/CVPRW.2016.5.
[11] Razak MA, Alias NA, Yusoff YM, Ahmad SA. Physiological-based driver monitoring systems: a scoping review. IEEE Access 2021;9:9192–210. doi: https://doi.org/10.1109/ACCESS.2021.3052424.
[12] Ali K, Tariq A, Abbas H. Encrypted traffic classification using deep learning techniques. In: 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS). IEEE; 2019. p. 1–6.
[13] Lu W, Zhang Y, Jiang X, Li Y, Li G. Encrypted traffic classification based on deep bidirectional LSTM and random forest. IEEE Access 2020;8:36571–81.
[14] Zhou Y, Liu X, Hu Q, Zhong Z. Encrypted traffic classification with bidirectional LSTM networks. In: Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM). Waikoloa, HI, USA: IEEE; 2019. p. 1–6.
[15] Wang J, Jia L, Liu X, Zhou Y. Encrypted traffic classification using convolutional neural networks with attention mechanism. IEEE Access 2019;7:24513–25.
[16] Ali S, Arfeen M, Rehman S. Classification of encrypted and unencrypted traffic using machine learning techniques. In; 2020: International Journal of Computer Science and Network Security (IJCSNS), vol. 20(2), p. 47–55.
[17] Al Rawi A, Al Saffar A, Al-Adhami M. Encrypted traffic classification using statistical analysis and machine learning techniques. In: Proceedings of the 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData); 2017. p. 538–543. doi: 10.1109/iThings-GreenCom-CPSCom-SmartData.2017.96.
[18] Amin M, Liu X, Al-Dhelaan A. Using deep learning for encrypted traffic classification: an evaluation study. In: 2019 international conference on computational science and computational intelligence (CSCI); December 2019. p. 129–34. doi: 10.1109/CSCI49375.2019.00030.
[19] F.Hussain, M. Ismail, and A. Khan, "Encrypted Traffic Classification Using Machine Learning Techniques," in 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Sept. 2018, pp. 101-105. DOI: 10.1109/ICACCCN.2018.8685265.
[20] Shi Z, Luktarhan N, Song Y, Tian G. BFCN: A Novel Classification Method of Encrypted Traffic Based on BERT and CNN. Electronics 2023.
[21] Xinyi Hu, Chunxiang Gu, Wei F. CLD-net: a network combining CNN and LSTM for internet encrypted traffic classification. Machine learning for security and communication Networks 2021.
[22] Wang Wei, Zhu Ming, Wang Jinlin, Zeng Xuewen, Yang Zhongzhen. End to end encrypted traffic classification with one-dimensional convolution neural networks. In: 2017 IEEE International Conference on Intelligence and Security Informatics (ISI); 2017.
[23] Zhou K, Wang W, Chenhuang Wu, Teng Hu. Practical evaluation of encrypted traffic classification based on a combined method of entropy estimation and neural networks. ETRI J 2019.
[24] Guo L, Qianqiong Wu, Liu S, Duan M, Li H, Sun J. Deep learning-based real-time VPN encrypted traffic identification methods. J Real Time Image Process 2020.
[25] Pathmaperuma MH, Rahulamathavan Yogachandran, Dogan Safak, Kondoz Ahmed. CNN for user activity detection using encrypted in-app mobile data. Future Internet 2022.
[26] Cai Y, Zhang W, Zhang F, Wang X. A machine learning approach to traffic classification in encrypted communication. In: 2019 IEEE 19th International Conference on Communication Technology (ICCT). Chengdu, China: IEEE; 2019. p. 212–6.
[27] Huang Y, Li Y, Qiang B. Internet traffic classification based on min-max ensemble feature selection. In: Proceedings of the 2016 international joint conference on neural networks (IJCNN). Vancouver, BC, Canada; July 2016. p. 3485–3492.
[28] Lotfollahi M, Zade RSH, Jafari Siavoshani M, Saberian M. Deep packet: a novel approach for encrypted traffic classification using deep learning. Soft Comput 2020;24:1999–2012.
[29] Shi H, Li H, Zhang D, Cheng C, Cao X. An efficient feature generation approach based on deep learning and feature selection techniques for traffic classification. Comput Netw 2018;132:81–98.
[30] Zhang Z, Kang C, Fu P, Cao Z, Li Z, Xiong G. Metrie learning with statistical features for network traffic classification. In: Proceedings of the 2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC), San Diego, CA, USA; December 2017. p. 1–7.

Reham Taher Elmaghraby: Graduated from Arab academy for science technology and maritime transport- Cairo branch in 2003. She achieved her masters at Arab Academy for Science and Technology in 2009. She liked reading and attending conferences. Her field of interests is Cryptography, IOT, Machine Learning and Network Security

Nada M. Abdel Aziem: Graduated from Arab academy for science technology and maritime transport- Cairo branch in 2010. She achieved her PhD at Ain Shams University in 2021. She liked reading and attending conferences. Her field of interests is computer networks, IOT, Machine Learning and Data Security

Mohamed A. Sobh: Mohamed A. Sobh is a Professor of Computer and System Engineering, Ain Shams University. He received his B.Sc., M.Sc. and PhD. in Computer Engineering from Ain Shams University in 1996, 2001, and 2007 respectively. His fields of research are Simulation and Modeling - Computer Programming - and Computer Security.

Ayman M. Bahaa-Eldin: Misr International University, On Leave from Ain Shams University, Egypt, ayman.bahaa@eng.asu.edu.eg, Ayman M. Bahaa-Eldin is a professor, Computer and Systems Eng. Dept. Ain Shams University. Prof. Bahaa-Eldin received his B.Sc., M.Sc. and PhD. in Computer Engineering from Ain Shams University in 1995, 1999, and 2004 respectively. He was a visiting professor in several local and international universities. He was appointed as the managing director of the Egyptian Universities Network (EUN), The Egyptian National Research and Education Network (NREN) from 2010 until 2014. His fields of research are Cryptography, Computer Networks, and Computer and Network Security. He published more than 80 papers in refereed international journals and conferences, managed and participated in several national and international research projects, and participated in the reviewing process of many international journals and conferences.