EasyChair Preprint

№ 9690

Encrypted Network Traffic Classification and Feature Selection by Ensemble of CNN and TLBO Meta-Heuristic Algorithm

Kr Harinath and G Kishorekumar

February 9, 2023

Encrypted Network Traffic Classification and Feature Selection by Ensemble of CNN and TLBO Meta-Heuristic Algorithm

Harinath K R
Research Scholar
Department of CSE
JNTUA
Anantapuramu, India
harirooba007@gmail.com

Kishorekumar G
Associate Professor
Department of CSE
RGM College of Engineering and Technology,
Nandyal, India
kishorgulla@yahoo.co.in

***Abstract*--The network traffic and its classification are crucial to network administration and monitoring. The extensive use of encryption techniques and the dynamic ports policy make it difficult for standard traffic classification algorithms to classify encrypted data. Deep learning techniques have lately been the subject of in-depth research for network traffic categorization. Unfortunately, a lot of training data is needed for these models. The fact that the characteristics for most traffic categorization algorithms must be retrieved by a specialist presents another difficulty. Finding the required elements that contribute to a better categorization using these approaches is highly laborious and time-consuming. In order to construct a traffic classification model that properly identifies traffic categories, this study combines the convolutional neural network (CNN), Teaching-Learning Based Optimization Algorithm (TLBO), and Self-Organizing Maps (SOM) methods. The suggested approach is a blend of CNN and TLBO, where CNN is an image-based technique that can accurately identify encrypted network data and TLBO is a feature selection mechanism that combines Self Organizing Maps (SOM). This approach is highly lightweight and has the ability to automatically extract features, choose features, and categorize encrypted network information.**

Keywords— Deep learning, Encrypted traffic, CNN, TLBO, SOM.

I. INTRODUCTION

Machine learning algorithms are frequently used in statistical feature-based approaches to categorize network traffic. These techniques can successfully categorize encrypted network data, but they necessitate manually designing network traffic characteristics and choosing suitable features through the approach of trial and error. Since these approaches heavily rely on the expertise of specialists, they not only necessitate a lot of time and effort to examine the data samples [1]. Even if these algorithms choose a small subset of the many features produced from the traffic data in order to classify encrypted information, this entails a high computational cost and the features derived from the data is constrained. As a result, the chosen feature subset for encrypted traffic categorization may not always be the best. A typical machine learning method, deep learning algorithms yield outstanding results in a wide range of domains, including computer vision and natural language processing. They can automatically learn characteristics from raw data. Therefore, figuring out how to use the deep learning algorithm to classify encrypted communication is a challenging task. The majority of end-to-end studies to far have ignored the efficiency of encrypted traffic in favor of alternative deep learning methods for traffic classification [2], [3] [4], [5].

In this research, we suggest an end-to-end strategy that uses an image-based approach to address these issues. In order to achieve the goal of categorizing encrypted network data, this technique turns the first few non-zero payload sizes of a session into grey pictures and uses the one dimensional convolution neural network (1D-CNN) deep learning architecture to automatically extract and pick features. The suggested technique merely uses the initial non-zero payload values of every session to categorize encrypted network data; therefore it can be implemented quickly and at little computing cost [6].

Deep network models that allow for the extraction and learning of features have greatly enhanced the efficiency of traffic categorization. The two biggest issues are the selection of reliable features and having enough data available for some traffic. The accuracy of network traffic categorization is typically hampered by these issues. Deep learning algorithms primarily employ an end-to-end deep learning-based classification approach to categorize network traffic, which eliminates the need for human feature extraction and eliminates the possibility of additional algorithm optimization. Meta-heuristic techniques were therefore employed in order to improve the algorithm and get around some of the issues and difficulties associated with deep learning. They were used for this reason: after a series of iterations, they employ simple procedures and operations to eventually arrive at an appropriate and ideal answer. When exploring the feature space for an ideal subset of features, the TLB optimization approach and one of the meta-heuristic algorithms work well [7]. The TLBO produces extremely ambitious outcomes in terms of exploitation, enhanced exploration, local optimum avoidance, and convergence in many optimization function forms. As a result, it is frequently employed to address issues in a variety of fields.

The method described in this study can recognize network traffic, automatically extract characteristics, and address the issue of data availability. Different forms of traffic may be classified using our suggested approach with high accuracy and acceptable performance. Additionally, it may resolve issues with encrypted packets, dynamic protocols like P2P (Peer to Peer), and virtual private network-based protocols. As a result, the approach described in this research, which combines CNN, ATLBO, and fuzzy-SOM neural network-based clustering, can identify the types of applications (protocols) with high accuracy.

The following are the key elements and advantages of the suggested method:

• Making use of entropy variance and entropy to pre-process traffic data.
• Automated feature extraction from the one-dimensional convolutional neural network's hidden layers (1D-CNN).
• High classification accuracy and effective feature selection utilizing TLBO.
• Clustering that uses a fuzzy-SOM to classify novel instances.

The remaining sections of this paper discussed in the following order. A survey of the literature is conducted in Section 2, and Section 3 introduces the fundamental ideas of CNN and meta-heuristic algorithms. The suggested approach is described in Section 4 and Experimental results by comparing the approach with previous approaches in Section 5. The paper is concluded in Section 6.

II. RELATED WORK

Gil et al. [8] developed a technique to enhance the performance of network traffic classification using deep learning techniques. They employed the BP-based model for comparison under the same circumstances and developed a network applications classifying model using the Deep Belief Network (DBN). The results demonstrated that DBN-based network traffic classification had a higher level of accuracy. Yamansavascilar et al. [9] utilizing the "ISCX VPN-non VPN" dataset and 111 flow characteristics with 12 application types, 94% accuracy was attained. The primary drawback of these approaches is that the extraction of features and selection operations are carried out by specialists. As a result, such techniques are time-consuming, costly, and subject to human error. Rezaei et al. [10] established a broad framework for deep learning-based network traffic classification. Lotfollahi et al. [11] proposed a method for categorizing encrypted communications based on CNN and Stacked Auto-Encoder (SAE). They employed a similar number of classes and data instances to those in the current investigation. When using the CNN model to classify traffic, they achieved an accuracy of 94%. Wang et al. [12] suggested a 1D-CNN-based technique for categorizing encrypted communications. In order to provide a final, cohesive framework for automatic learning of the non-linear relationship between the row input and the anticipated output, this technique includes feature extraction, feature selection, and the classifier. The accuracy of their solution, which used the "ISCX VPN-non VPN" dataset, was 92% for 12 types of applications. They didn't utilize enough examples of the dataset just a few to be able to make any firm conclusions.

The performance of most IDSs in terms of accuracy in classification and training time has been impacted by the rise in the amount of audit data characteristics. This research suggests using the TLBO approach to resolve this problem through a quick and precise optimization procedure that can enhance IDS's capacity for locating the best detecting model based on Machine Learning. Rao and Patel [13]. The mechanical design problems posed by the TLBO approach do not require any user-defined parameters during the optimization phase. The results showed that TLBO performed better than Particle Evolutionary Swarm Optimization, Artificial Bee Colony (ABC), and cultural Differential Evolution when this unique approach was evaluated on several benchmark functions (DE). Das and Padhy [14] examined the potential for combining data from several commodities futures indexes derived through multi-cut crossover to apply an unique TLBO method to the selection of appropriate data points for an SVM regression model of financial time-series data (MCX). Compared to the standard SVM, the suggested hybrid SVM-TLBO model seems to have been more successful in locating the ideal parameters based on the experimental findings. Nayak *et al*. [15] built a matrix of solutions for each objective of a multi-objective TLBO. The best answer offered in the optimal solution which is the primary criterion used in TLBO to pick teachers, and learners are only instructed to maximize that goal. To create a set of ideal solutions, all the potential answers in the solution space are sorted.
When combined with the 1D-CNN, TLBO, and SMO algorithms, our suggested strategy significantly increased accuracy on the same dataset when compared to previous work on the "ISCX VPN-non VPN" dataset.

III. CONVOLUTIONAL NEURAL NETWORK AND TEACHING-LEARNING BASED OPTIMIZATION ALGORITHM

Convolutional neural networks (CNN) and the Teaching-Learning Based Optimization algorithm (TLBO) principles are discussed in this part.

A. Convolutional Network Theory

The convolutional neural networks algorithm is one of the most widely used deep learning techniques (CNNs). An input layer, an output layer, and a number of hidden layers make up a CNN. Convolutional, pooling, and fully connected layers are the three main types of layers that make up a CNN. In general, CNNs are tiered neural networks in which numerous fully connected layers are built after the convolutional and pooling layers are alternatively organized. There are three benefits to doing a convolution process. [16] Every feature map that uses a weight sharing technique has a significantly smaller number of parameters, a local connection that learns the relationships between nearby pixels, and immutability and stability over object substitution. The amount of parameters and the size of the feature map may both be decreased by using a pooling layer, which is often placed after the convolutional layers. The network's output may then be represented as a vector with a specific size thanks to completely linked layers. We think that this approach is the best choice for categorizing network traffic. The findings supported this assertion and demonstrated the effectiveness of 1D-CNN for extracting features from network traffic categorization.

B. Teaching Learning Based Optimization Concept

The teacher (the best solution) communicates his knowledge with the students (the population of solutions), and his effectiveness as a teacher has an impact on the students' grades in TLBO [17], which replicates the influence of instructors on students in a traditional school learning process (fitness values). There are two primary phases to the TLBO learning process: The best answer is chosen as the teacher in the first phase, and the mean of the student positions is computed and moved in favor of the teacher's position. The new student position is then determined as follows:

$$a_i^{(t+1)} = a_i^t + r(a^* - TS.\ \bar{a}) \quad (1)$$

Where a_i^t is the position of the previous i^{th} student, r is a random number between [0, 1], a^* is the location of the current instructor, ā is the mean of the current population, and TF is the teaching factor that was randomly determined. In the second phase, the student i interacts with student j, who was chosen at random, to broaden his knowledge.

Three criteria are used to compare students i and j:

- The option with the higher fitness value is chosen if both are technically possible.
- The viable solution is chosen if one is possible while the other is impractical.
- The option with the least amount of the feasibility restrictions violated is chosen if both are impractical.

If the chosen student's knowledge was superior, the knowledge of student i is changed. The knowledge of the learner is modified in the following ways:

$$a_i^{(t+1)} = a_i^t + r(a_i^t - a_j^t) \text{ if } f(a_i^t) < f(a_i^t) \quad (2)$$

$$a_i^{(t+1)} = a_i^t + r(a_j^t - a_i^t) \quad (3)$$

By requiring no particular parameters, the authors demonstrated that TLBO is an algorithm-specific parameter-less algorithm. The authors used many constrained benchmark functions and mechanical design problems to demonstrate the effectiveness of TLBO by comparing it to Multi-membered Evolutionary Strategy (M-ES) [18], Particle Evolutionary Swarm Optimization (PESO) [19], Cultural Differential Evolution (CDE) [20], Co-evolutionary Differential Evolution (CoDE) [21], and Artificial Bee Colony (ABC) [22]. The TLBO results generally exceeded the opposition.

```
TLBO Algorithm
Create initial students;
Calculate count of the students;
Set the teacher;
Compute the average of all students;
while(termination criterion is not satisfied) do
// Teacher phase
        for every student
                change student position according to teacher position;
        end
        Evaluate new students;
        if updated solution is better than old one
                Accept the new solutions better than the old ones;
        Otherwise
                keep the old solutions
// Learner phase
        for every student
                Select different student randomly;
                if the selected student is better
                        change student position;
                end
        end
        calculate new students;
        Accept the new solutions better than the old ones;
        change the teacher and the average;
end
```

Fig. 1. TLBO Pseudo code

IV. PROPOSED METHOD

Deep learning and a meta-heuristic algorithm are used in a novel method for categorizing network traffic that is described. Figure depicts the suggested technique in broad terms. The four steps of the suggested technique are listed below. 1-The first step is to preprocess the communication flows using entropy and entropy variance.2-Secondly 1D-CNN-based automated feature extraction. 3- Use of the TLBO meta-heuristic method for effective feature selection. 4- Classification of new instances using a hybrid fuzzy-SOM-based clustering. Each step is thoroughly described.

In order to normalize and standardize the data, preprocessing is done to the dataset. By computing the entropy of the collected statistical feature values, one may identify the data in a flow that is prone to error. Over time, a flow with a low entropy characteristic is likely to have redundant patterns. In other words, this flow's pattern is more consistent and is an excellent choice for the training phase.

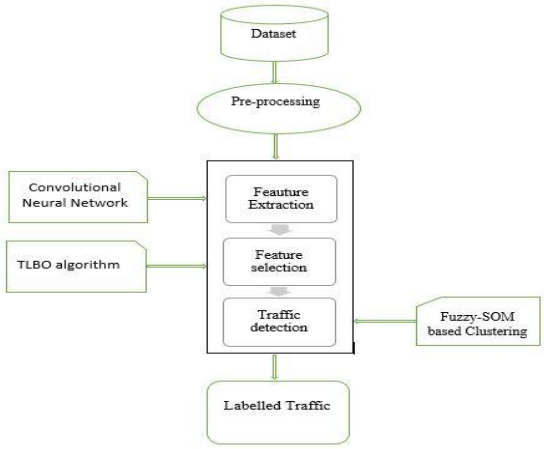

Fig. 2. Framework of Proposed Traffic Classification Mechanism

One of the most widely used deep learning feature extraction methods is the CNN. The processed data are then passed to a CNN in the following phase to extract features for this reason. The suggested approach works using an estimating model that applies the grid search algorithm to calculate estimates of the amount of neurons, the number of training iterations, and other CNN parameters. Two convolutional layers, two pooling layers, and five fully linked layers make up the CNN model used in this work. The final classification layers are fully connected layers, in which the output features are determined by dividing the input feature vector by the completely connected weight matrix. The output of this stage is made up of a number of new combinational features that need to be retrieved from the convolutional network structure's hidden layers.

Relevant characteristics must be eliminated during the feature selection step in order to improve classification performance. In order to maximize classification performance and locate the ideal feature set, TLBO was employed in this study as the search method. In order to choose the best features and get around the data problem, it looked helpful to apply Teaching Learning based optimization meta-heuristic methods. After the features were extracted, a high-accuracy optimum set of the feature sets was discovered using the TLBO meta-heuristic method.

This step results in a feature set that is highly accurate in classifying traffic.

Unsupervised neural network with few parameters and neurons grouped in a same network structure. An n-dimensional weight vector exists for each neuron. The input layer's weight vectors are connected to the output layer, also known as the map or competition layer, by weight vectors (synapses). The proximity function links the neurons together. Each input layer activates a winning cell neuron inside the output layer based on which input layers share the greatest similarities. The most significant distinction between the SOM optimization technique and other vector measuring techniques is that, in addition to updating the weight of the winner cell's neighbor cells, the SOM training algorithm also updates the weight of the specific transmission weight with the highest adaptation (the winner cell). The values of the retrieved features are initially sorted into k groups using clustering in the SOM network, an unsupervised learning network, depending on the training data. In accordance with protocol settings, each flow's label is simultaneously known (UDP, HTTPS, etc.). The initial value of k is set to k = 6, but when more clusters are added, the partitioning of flow instance based on feature similarity improves. As a result, it is anticipated that the suggested approach will become more accurate.

V. EXPERIMENTAL RESULTS

The suggested method's accuracy was compared to that of previous approaches [8], [9], and [11] using the "ISCX VPN-non VPN" dataset.

TABLE I. COMPARISON OF PROPOSED MECHANISM WITH PREVIOUS APPROACHES

S.No	Method	Accuracy
1	Decision tree[14]	92
2	KNN[15]	94
3	CNN,SAE[17]	94
4	CNN-TLBO(Proposed)	97

Table 1 compares the suggested approach to various methods that target the "ISCX VPN-non VPN" dataset in broad terms. Comparing the suggested strategy to both ML and DL approaches, it achieved a greater accuracy. Since the deep learning network performs feature extraction automatically, numerous features may be produced that are worthless in accuracy and potentially increase computational cost and time.

VI. CONCLUSION

To automatically choose attributes and precisely categorize network traffic, a deep learning and evolutionary algorithm-based approach was suggested. The suggested approach makes use of genetic algorithms to choose the best characteristics and identify the best solution, cutting down on both execution time and computing cost. The goal of this approach, which combines CNN, TLBO, and SOM, is to properly categorize network traffic irrespective of the size of the training data. The suggested technique may identify an optimum training strategy to categorize network traffic with great accuracy by fusing an evolutionary algorithm with neural networks. On the "ISCX VPN-non-VPN" dataset, the findings demonstrated that the recommended strategy performed better than all comparable strategies.

REFERENCES

[1] Wang, Z. (2015). The applications of deep learning on traffic identification. *BlackHat USA*, *24*(11), 1-10.

[2] Wang, W., Zhu, M., Zeng, X., Ye, X., & Sheng, Y. (2017, January). Malware traffic classification using convolutional neural network for representation learning. In *2017 International conference on information networking (ICOIN)* (pp. 712-717). IEEE.

[3] Chen, Z., He, K., Li, J., & Geng, Y. (2017, December). Seq2img: A sequence-to-image based approach towards ip traffic classification using convolutional neural networks. In *2017 IEEE International conference on big data (big data)* (pp. 1271-1276). IEEE.

[4] Wang, W., Zhu, M., Wang, J., Zeng, X., & Yang, Z. (2017, July). End-to-end encrypted traffic classification with one-dimensional convolution neural networks. In *2017 IEEE international conference on intelligence and security informatics (ISI)* (pp. 43-48). IEEE.

[5] Lotfollahi, M., Jafari Siavoshani, M., Shirali Hossein Zade, R., & Saberian, M. (2020). Deep packet: A novel approach for encrypted traffic classification using deep learning. *Soft Computing*, *24*(3), 1999-2012.

[6] Shim, K. S., Ham, J. H., Sija, B. D., & Kim, M. S. (2017). Application traffic classification using payload size sequence signature. *International Journal of Network Management*, *27*(5), e1981.

[7] Rao, R. V., & Patel, V. (2013). An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems. *Scientia Iranica*, *20*(3), 710-720.

[8] Draper-Gil, G., Lashkari, A. H., Mamun, M. S. I., & Ghorbani, A. A. (2016, February). Characterization of encrypted and vpn traffic using time-related. In *Proceedings of the 2nd international conference on information systems security and privacy (ICISSP)* (pp. 407-414).

[9] Yamansavascilar, B., Guvensan, M. A., Yavuz, A. G., & Karsligil, M. E. (2017, January). Application identification via network traffic classification. In *2017 International Conference on Computing, Networking and Communications (ICNC)* (pp. 843-848). IEEE.

[10] Rezaei, S., & Liu, X. (2019). Deep learning for encrypted traffic classification: An overview. *IEEE communications magazine*, *57*(5), 76-81.

[11] Lotfollahi, M., Jafari Siavoshani, M., Shirali Hossein Zade, R., & Saberian, M. (2020). Deep packet: A novel approach for encrypted traffic classification using deep learning. *Soft Computing*, *24*(3), 1999-2012.

[12] Wang, W., Zhu, M., Wang, J., Zeng, X., & Yang, Z. (2017, July). End-to-end encrypted traffic classification with one-dimensional convolution neural networks. In *2017 IEEE international conference on intelligence and security informatics (ISI)* (pp. 43-48). IEEE.

[13] Rao, R. V., & Patel, V. (2013). An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems. *Scientia Iranica*, *20*(3), 710-720.

[14] Das, S. P., & Padhy, S. (2018). A novel hybrid model using teaching–learning-based optimization and a support vector machine for commodity futures index forecasting. *International Journal of Machine Learning and Cybernetics*, *9*(1), 97-111.

[15] Nayak, M. R., Nayak, C. K., & Rout, P. K. (2012). Application of multi-objective teaching learning based optimization algorithm to optimal power flow problem. *Procedia Technology*, *6*, 255-264.

[16] Zeiler, M. D. (2013). *Hierarchical convolutional deep learning in computer vision* (Doctoral dissertation, New York University).

[17] Rao, R. V., Savsani, V. J., & Vakharia, D. P. (2011). Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. *Computer-aided design*, *43*(3), 303-315.

[18] Mezura-Montes, E., & Coello, C. A. C. (2005). A simple multimembered evolution strategy to solve constrained optimization problems. *IEEE Transactions on Evolutionary computation*, *9*(1), 1-17.

[19] Muñoz Zavala, A. E., Aguirre, A. H., & Villa Diharce, E. R. (2005, June). Constrained optimization via particle evolutionary swarm optimization algorithm (PESO). In *Proceedings of the 7th annual conference on Genetic and evolutionary computation* (pp. 209-216).

[20] Becerra, R. L., & Coello, C. A. C. (2006). Cultured differential evolution for constrained optimization. *Computer Methods in Applied Mechanics and Engineering*, *195*(33-36), 4303-4322.