Journal of Optimization in Soft Computing (JOSC)

Vol. 3, Issue 1, pp: (24-29), Spring-2025

Journal homepage: https://sanad.iau.ir/journal/josc

Paper Type (Research paper)

Using Machine Learning to Discover Traffic Patterns in Software Defined Networks

Abdulrazzaq Mosa Al-Mhanna[1] and Pouya Khosravian Dehkordi[1,*]

1. Department of Computer Engineering, Shahrekord Branch, Islamic Azad University, Shahrekord, Iran.

Article Info

Article History:
Received: 2024/۰۷/۱۷
Revised: 2025/04/14
Accepted: 2025/۰۶/۲۳

DOI: josc.2025.140305271129363

Keywords:
Network Traffic, Software-Defined Networks, SDN, Machine Learning

**Corresponding Author's Email Address: drkhosravian@iau.ac.ir*

Abstract

In this research, we introduce a deep learning model based on Convolutional Neural Networks (CNNs) along with the Bird Swarm Optimization algorithm to identify and discover traffic patterns in Software-Defined Networks (SDNs). The main objective of this study is to investigate the capability of deep learning models in analyzing traffic data and identifying unique patterns present in SDNs. Using a diverse and comprehensive dataset, the proposed model is trained and evaluated. The use of CNNs, due to their layered structure and deep learning capabilities, enables the identification of unique traffic patterns that are prominently visible in SDNs. The proposed model, with high accuracy and good generalization ability, can serve as an effective tool in enhancing the accuracy and efficiency of traffic pattern identification systems in software-defined networks. This research not only demonstrates the superiority of deep learning models in traffic pattern recognition but also provides practical and effective solutions for traffic analysis and management in SDNs. The results of this study indicate that the proposed model achieves an accuracy of 96.5%, suggesting that the proposed method can significantly contribute to the development and improvement of security systems and performance optimization in software-defined networks.

1. Introduction

Management and configuration of computer networks has become a difficult and vital task due to their complexity and dynamics. These networks consist of a collection of switches, routers, firewalls, and other intermediate devices that work simultaneously. Proper implementation of these networks is possible by operators dealing with a limited set of configuration commands in command-line environments and with complex administrative tasks and policies. These policies and complexities are not enough to react to the continuous changes of the network. For this reason, network configuration modifications are done manually to adapt the network to the changes. Operators use external tools to overcome these limitations, and these constant changes may lead to more configuration errors [1, 24].

1.۲. Problem Statement

For network management, service measurement, and network monitoring, traffic classification is an intelligent process that involves categorizing traffic into multiple groups. In addition, traffic classification enables the configuration of access restrictions, quality of service, and other network security features efficiently and allocates resources. Deep packet inspection and port-based methods are popular methods for traffic classification [2]. However, both of these methods are currently less used, as most applications use dynamic ports and the network communication is encrypted. Therefore, it is very important to

develop a new classification method that is more suitable for today's operational environment. The purpose of this research is to discover network traffic patterns with high accuracy. To extract the patterns, a deep learning based approach is proposed.

1.3 innovation in research

This research pushes the boundaries by exploring and applying advanced deep learning architectures such as deep neural networks (DNN), convolutional neural networks (CNN), recurrent neural networks (RNN), and attention mechanisms. By doing so, an attempt is made to provide a pioneering approach to modeling and understanding network traffic patterns. In summary, the innovative aspect of this research lies in its pioneering use of deep learning models to achieve high accuracy in discovering and analyzing traffic patterns in software-defined networks [25]. This approach has the potential to transform network management, security, and performance optimization, making it a cornerstone for further advancements in this field.

2. Software networks

Organizations have invested heavily in virtualization and hybrid clouds, but they still face challenges, including quickly allocating network connections while systems are running. Often these problems arise due to issues related to policy or implementation processes.

These problems can be partially solved by creating virtual network infrastructure. This infrastructure is easily reassigned, such as when a new SAN or server is implemented. The idea behind this software-defined network management infrastructure, or SDN, is not that new and has been around for over a decade. One efficient definition of SDN is the separation of data and control functions of routers and other layer 2 infrastructure of conventional networks using a programming interface.

2.1 PSO algorithm

The PSO algorithm is an optimization method based on probability rules that was first invented in 1995 by Kennedy and Aberhart [3] inspired by the behavior of birds when searching for food. In this algorithm, first a set of initial answers is generated. Then, to find the optimal answer in the space of possible answers, or to time the generations, the answer search is done. Each particle is defined multidimensionally with two values of position and velocity, and at each stage of the particle's movement, with two indices of velocity and position, the best answers are determined in terms of merit for all particles.

$$\overrightarrow{X_i}(t) = \left(x_i^1(t).x_i^2(t).\ldots.x_i^d(t).\ldots.x_i^n(t)\right).$$

$$\overrightarrow{V_i}(t) = \left(v_i^1(t).v_i^2(t).\ldots.v_i^d(t).\ldots.v_i^n(t)\right).$$

Along with their descriptions are presented in Table 1-3. Meta-heuristic algorithms are known as effective techniques to improve the performance of CNN architectures by optimizing their meta-parameters.

2.2Related works

Basic machine learning methods that enable traffic classification in SDN are reviewed in this section. Through the use of artificial intelligence (AI), machine learning enables computers to recognize complex patterns from massive data sets on their own. Operationally, machine learning is divided into two steps: 1) training, which involves providing the machine learning algorithms with a subset of the data set (called the training set) so that the system model can learn from it, and 2) decision making, which is capable of The system is trained to predict the result of the new input using the model. Supervised, unsupervised, semi-supervised and reinforcement learning categories are used to group machine learning algorithms [4], [5] and [6].

Numerous machine learning methods have been developed over the years as a result of research efforts. For problems with large data sets, machine learning is often the most effective approach. Considering that machine learning techniques are designed for pattern recognition and data identification, they are suitable for solving problems in SDNs.

3. Preposed method

Optimizing the parameters of convolutional neural networks includes determining the appropriate parameters, which results in significant accuracy in each task. However, the task of optimizing a large number of parameters is very difficult and computationally expensive. Therefore, it is necessary to implement optimization algorithms that reduce the number of iterations. The present study is based on the Particle Swarm Optimization (PSO) technique to find the CNN model with the highest accuracy for breast cancer detection. The development of a convolutional neural network (CNN) involves the optimization of several parameters and the precise choice of architecture. Choosing the optimal parameters is very important to obtain accurate results when using Convolutional Neural Networks (CNN). Therefore, it is a challenging task that requires a considerable level of expertise.

The effectiveness of a CNN model depends on its meta-parametric parameters, so some researchers emphasize the necessity of fine-tuning these meta-parameters to obtain positive results. Hence, it is a challenging task that requires a considerable amount of skill. The meta-parameters of the CNN architecture

3.2 Implementation of neural network algorithm optimized with particle swarm optimization algorithm

In this section, we build a CNN from scratch (a new model) to train a Convolutional Neural Network (CNN). It consists of three convolution layers with three maximum localization layers, one dropout layer, one flattening layer, and two fully connected (FC) layers. The activation function for each layer is the ReLU function, except for the last layer which is output and uses the sigmoid function. The output layer uses a sigmoid function that maps the output value to the interval [0, 1]

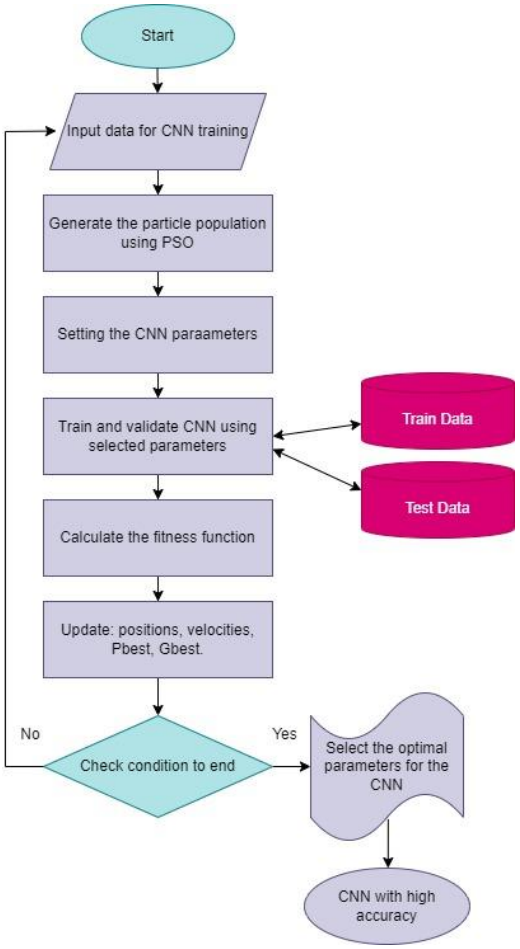

Figure 1-3: Flowchart of neural network optimized by PSO

4.1 evaluation criteria

In this section, various evaluation criteria used to measure the performance of machine learning models in discovering traffic patterns in software-centric networks are described in detail. These criteria include accuracy, recall, and F1 score. Each of these metrics evaluates different aspects of model performance and provides a deeper understanding of model performance.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (1)$$

$$Sensitivity\,(Recall) = \frac{TP}{TP+FN} \quad (2)$$

$$F1\ score = \frac{2*(Recall*Precision)}{(Recall+Precision)} \quad (3)$$

4.2 Data collection

Software-oriented networks (SDN) are one of the leading technologies in network management and control, which enable centralized control and higher flexibility by separating the control layer from the data layer. In the field of SDN, multiple datasets are used for various purposes, including network traffic analysis, attack detection, and network performance optimization.

The NSL-KDD dataset is one of the most popular and comprehensive datasets in the field of network security and intrusion detection. This dataset is an improved dataset of KDD Cup 99 and is designed to address its problems and limitations. The KDD Cup 99 dataset was introduced as one of the first and most comprehensive datasets in the field of intrusion detection.

• Duplicate data: The presence of a large number of duplicate samples in the dataset, which caused the machine learning models to mistakenly perform very well.

• Imbalance in the data: unbalanced distribution of different samples in the data set, which caused the models to tend to oversampled classes.

4.3 The size of the parameters

In this project, our main goal is to use Convolutional Neural Networks (CNN) and Particle Swarm Optimization (PSO) algorithm to discover traffic patterns in software-oriented networks. For this purpose, a set of parameters for convolutional neural network and PSO algorithm are considered.

4.4Training settings:

1. Initial learning rate for updating network weights. A low value of this parameter allows the network to be trained with smaller and more accurate steps.

2. The number of complete training iterations on the training data. Increasing this value helps the model to reach higher accuracy.

3. The number of examples in each small training package. This parameter helps balance between training speed and stability.

4.5. Simulation results

To evaluate the performance of the proposed method, accuracy, recall and F1-Score criteria have been used. 60% of the data was used for training and 40% for testing. The results obtained from the evaluation of the proposed method and its comparison with two other references are as follows:

Table 4-1: Comparison of the proposed method with other methods

	evaluation criteria		
	accuracy	recall	F1-score
Preposed method	۹۶/۵	۹۴/۸۶	۹۵/۸۵
[۳۸]	۸۷/۶	۸۹/۶	۹۰/۰۱

[۴۰]	۹۲/۳۸	۹۳/۱۱	۹۴/۸۷

The results show that the proposed method has the highest accuracy and F1 criterion compared to the other two references, and it is close to the highest value in readout. This shows that the proposed method has been able to establish a good balance between correctly identifying attack samples and preventing false positive samples.

The proposed method using convolutional neural networks and particle swarm optimization algorithm has been able to show better performance than the previous methods in detecting penetration and traffic analysis of SDN networks.

In this section, the results of the evaluation of different machine learning algorithms to discover traffic patterns in software-oriented networks (SDN) are analyzed. The following table shows the accuracy results of different algorithms:

Table 4-2: Comparison of the proposed method with other algorithms

Algorithm	Accuracy
proposed method	۹۶/۵
KNN	۷۱/۴۷
DT	۹۵/۷۶
SVM	۹۵/۷۴

The proposed algorithm, which uses convolutional neural networks (CNN), has the best performance among the investigated algorithms with an accuracy of 96.5%. This result shows that CNN, with its capabilities in extracting complex features and deep learning, has been able to identify traffic patterns well and achieve higher accuracy than other algorithms.

The K-Nearest Neighbor (KNN) algorithm with 71.47% accuracy has the lowest accuracy among the investigated algorithms. This result shows that KNN may perform poorly when dealing with complex and high-dimensional data. Due to the simplicity of this algorithm and the inability to extract complex features, it provides less accuracy.

The decision tree (DT) algorithm has performed very well with an accuracy of 95.76% and is known as one of the efficient algorithms in identifying traffic patterns. Decision tree using tree structure and decision rules has been able to achieve high accuracy and work well with traffic data.

The support vector machine (SVM) algorithm has also performed well with 95.74% accuracy. SVM has been able to detect traffic patterns with high accuracy by using feature spaces and optimal separators. Although the accuracy of SVM is slightly lower than decision tree, it is still in the high performance range.

The results show that the proposed method using convolutional neural networks (CNN) has been able to achieve the best accuracy among the investigated algorithms. This shows the high power of CNN in identifying and learning complex patterns in traffic data. On the other hand, more traditional algorithms such as KNN, DT and SVM have also performed significantly, but could not reach the accuracy of the proposed method.

According to these results, the use of convolutional neural networks (CNN) as the proposed method in this research is a suitable choice and can help improve the accuracy and efficiency of traffic pattern detection systems in software-based networks. This method has many capabilities in analyzing complex data and extracting important features, which has made it a powerful tool in the field of machine learning.

5. Conclusion

the data set used in this research included various network traffics, including normal and abnormal traffics. The data has been collected from various sources to have high diversity and realism. The data pre-processing process has included cleaning, normalization and extraction of important features, which has greatly helped to improve the quality of the data and the accuracy of the models.

In this research, four main algorithms have been evaluated: k-nearest neighbor (KNN), decision tree (DT), support vector machine (SVM) and convolutional neural networks (CNN).

In addition to accuracy, other criteria such as recall and F1 criteria have also been used to evaluate the performance of models.

The proposed algorithm, which uses convolutional neural networks (CNN), has the best performance among the investigated algorithms with an accuracy of 96.5%. This result shows that CNN, with its capabilities in extracting complex features and deep learning, has been able to identify traffic patterns well and achieve higher accuracy than other algorithms. The readability of 94.86% and the F1 criterion equal to 95.85% also indicate the high ability of this algorithm to correctly detect positive samples and reduce positive and negative errors.

The K-Nearest Neighbor (KNN) algorithm with 71.47% accuracy has the lowest accuracy among the investigated algorithms. This result shows that KNN may perform poorly when dealing with complex and high-dimensional data. Due to the simplicity of this algorithm and the inability to extract complex features, it provides less accuracy.

The decision tree (DT) algorithm has performed very well with an accuracy of 95.76% and is known as one of the efficient algorithms in identifying traffic patterns. The decision tree has been able to categorize the traffic data well and achieve high accuracy by using the tree structure and decision rules. This algorithm is one of the popular methods in traffic data analysis due to its simplicity and high efficiency. But it has a weaker performance than the proposed method.

One of the main advantages of a decision tree is that it is naturally interpretable. This feature allows network administrators and analysts to easily understand the

reasons behind the decisions and classifications made by the model. This interpretability is especially valuable in cases where there is a need to explain the results to non-technical managers.

Decision tree can also work well with data with different features and incomplete data. However, one of the weaknesses of this algorithm may be the creation of overly complex trees and overfitting in the training data. To reduce this problem, techniques such as pruning are used to reduce the complexity of the tree and improve the model.

The support vector machine (SVM) algorithm has also performed well with an accuracy of 95.74%. Using feature spaces and optimal separators, SVM has been able to recognize traffic patterns with high accuracy. Although the accuracy of SVM is slightly lower than decision tree, it is still in the high performance range. The recall and F1 criterion for SVM are not available in this table, but it can be expected that this algorithm also performs well in these criteria.

One of the main advantages of SVM is its ability to work with high-dimensional data and determine optimal decision boundaries for separating classes. This feature makes SVM perform very well, especially in cases where the data is not linearly separable. However, one of the challenges of using SVM is the need to fine-tune its various parameters, such as the tuning parameter (C) and choosing the appropriate kernel type.

The results obtained from sources [38] and [40] have also been used as a comparative measure. These results show that other algorithms with accuracy, recall and F1 criterion have had acceptable performance of 87.6%, 89.6% and 90.01% respectively in [38] and 92.38%, 93.11% and 94.87% in [40], but still their performance it was less than the proposed method (CNN).

The results show that the proposed method using convolutional neural networks (CNN) has been able to obtain the best accuracy, readability and F1 criterion among the investigated algorithms. This shows the high power of CNN in identifying and learning complex patterns in traffic data. On the other hand, more traditional algorithms such as KNN, DT and SVM have also performed significantly, but could not reach the accuracy of the proposed method.

According to these results, the use of convolutional neural networks (CNN) as the proposed method in this research is a suitable choice and can help improve the accuracy and efficiency of traffic pattern detection systems in software-based networks. This method has many capabilities in analyzing complex data and extracting important features, which has made it a powerful tool in the field of machine learning.

Finally, the results of this research can pave the way for future research in the field of improving machine learning models to analyze and manage software-based network traffic. Using more advanced deep learning techniques, combining different models and improving data preprocessing processes can lead to achieving higher accuracy and efficiency in identifying traffic patterns. In this way, the security and efficiency of SDN networks will be significantly improved.

5.2 Future works

can lead to the development and improvement of current methods and open new horizons in the field of using machine learning and deep learning in software-based network traffic analysis and management. Therefore, continuing research in this field and applying new techniques can have positive effects on the security and efficiency of SDN networks.

In general, the findings of this research show the importance and efficiency of using advanced deep learning models, especially convolutional neural networks (CNN) in analyzing and identifying traffic patterns in software-based networks (SDN). In this section, research limitations and suggestions for future research are discussed.

Reference

[1] J. Zhang, C. Chen, Y. Xiang, W. Zhou, and A. V. Vasilakos, “An effective network traffic classification method with unknown flow detection,” IEEE transactions on network and service management, vol. 10, no. 2, pp. 133-147, 2013. 10.1109/TNSM.2013.022713.120250

[2] J. Frank, "Artificial intelligence and intrusion detection: Current and future directions." pp. 1-12. http://dx.doi.org/10.14514/BYK.m.26515393.2022.10/1.78-87

[3] Cotton, Michelle, et al. Internet assigned numbers authority (IANA) procedures for the management of the service name and transport protocol port number registry. No. rfc6335. 2011.

[4] J. V. Gomes, P. R. Inácio, M. Pereira, M. M. Freire, and P. P. Monteiro, “Detection and classification of peer-to-peer traffic: A survey,” ACM Computing Surveys (CSUR), vol. 45, no. 3, pp. 1-40, 2013. https://doi.org/10.1145/2480741.2480747

[5] M. Shafiq, X. Yu, A. A. Laghari, L. Yao, N. K. Karn, and F. Abdessamia, "Network traffic classification techniques and comparative analysis using machine learning algorithms." pp. 2451-2455. 10.1109/CompComm.2016.7925139

[6] S. Katal, and A. P. H. Singh, “A Survey of Machine Learning Algorithm in Network Traffic Classification,” International Journal of Computer Trends and Technology (IJCTT), vol. 9, no. 6, 2014. 10.14445/22312803/IJCTT-V9P157

[7] Y. D. Goli, and R. Ambika, "Network traffic classification techniques-a review." pp. 219-222. https://doi.org/10.1007/978-981-19-8493-8_29

[8] T. T. Nguyen, and G. Armitage, “A survey of techniques for internet traffic classification using machine learning,” IEEE communications surveys & tutorials, vol. 10, no. 4, pp. 56-76, 2008. 10.1109/SURV.2008.080406

[9] J. Kennedy, and R. Eberhart, "Particle swarm optimization." pp. 1942-1948. 10.1109/ICNN.1995.488968

[10] M. Kubat, An introduction to machine learning: Springer, 2017. https://doi.org/10.1007/978-3-319-63913-0
[11] H. Wu, and S. Prasad, "Semi-supervised deep learning using pseudo labels for hyperspectral image classification," IEEE Transactions on Image Processing, vol. 27, no. 3, pp. 1259-1270, 2017. 10.1109/TIP.2017.2772836
[12] H.-K. Lim, J.-B. Kim, K. Kim, Y.-G. Hong, and Y.-H. Han, "Payload-based traffic classification using multi-layer lstm in software defined networks," Applied Sciences, vol. 9, no. 12, pp. 2550, 2019. **https://doi.org/10.3390/app9122550**
[13] C. Xu, H. Lin, Y. Wu, X. Guo, and W. Lin, "An SDNFV-based DDoS defense technology for smart cities," IEEE Access, vol. 7, pp. 137856-137874, 2019. 10.1109/ACCESS.2019.2943146
[14] B. Park, J. W.-K. Hong, and Y. J. Won, "Toward fine-grained traffic classification," IEEE Communications Magazine, vol. 49, no. 7, pp. 104-111, 2011. 10.1109/MCOM.2011.5936162
[15] X. Chang, F. Nie, Y. Yang, and H. Huang, "A convex formulation for semi-supervised multi-label feature selection." https://doi.org/10.1609/aaai.v28i1.8922
[16] P. Wang, F. Ye, X. Chen, and Y. Qian, "Datanet: Deep learning based encrypted network traffic classification in sdn home gateway," IEEE Access, vol. 6, pp. 55380-55391, 2018. 10.1109/ACCESS.2018.2872430
[17] C. Zhang, X. Wang, F. Li, Q. He, and M. Huang, "Deep learning–based network application classification for SDN," Transactions on Emerging Telecommunications Technologies, vol. 29, no. 5, pp. e3302, 2018. https://doi.org/10.1002/ett.3302
[18] Z. Fan, and R. Liu, "Investigation of machine learning based network traffic classification." pp. 1-6. https://doi.org/10.1109/ISWCS.2017.8108090
[19] M. M. Raikar, S. Meena, M. M. Mulla, N. S. Shetti, and M. Karanandi, "Data traffic classification in software defined networks (SDN) using supervised-learning," Procedia Computer Science, vol. 171, pp. 2750-2759, 2020. https://doi.org/10.1016/j.procs.2020.04.299
[20] F. A. M. Zaki, and T. S. Chin, "FWFS: Selecting robust features towards reliable and stable traffic classifier in SDN," IEEE Access, vol. 7, pp. 166011-166020, 2019. https://doi.org/10.1109/ACCESS.2019.2953565
[21] A. Malik, R. de Fréin, M. Al-Zeyadi, and J. Andreu-Perez, "Intelligent SDN traffic classification using deep learning: Deep-SDN." pp. 184-189. https://doi.org/10.1109/ICCCI49374.2020.9145971
[22] P. Wang, S.-C. Lin, and M. Luo, "A framework for QoS-aware traffic classification using semi-supervised machine learning in SDNs." pp. 760-765. https://doi.org/10.1109/SCC.2016.133
[23] M. Amiri, H. Al Osman, and S. Shirmohammadi, "Game-aware and sdn-assisted bandwidth allocation for data center networks." pp. 86-91. https://doi.org/10.1109/MIPR.2018.00023
[24] F. Kiyoumarsi , B. Zamani ,Extending the Lifetime of Wireless Sensor Networks Using Fuzzy Clustering Algorithm Based on Trust Model, Journal of Optimization in Soft Computing , Issue 1 , 2023.https://doi.org/10.82553/josc.2023.140207147833 32
[25] P. Khosravian Dehkordi A Comprehensive Review on Service Function Chaining in Network Environments Journal of Optimization in Soft Computing , Issue 3, 2024.https://doi.org/10.82553/josc.2024.140212161104 657