An Analysis of Various Machine Learning Algorithms for Network Traffic Classification

Mahesh Kumar 1, Dr. Pratima Gautam 2

1Reasearch Scholar,RNTU,Bhopal (M.P.) India.

2 Dean, Dept. of CS and IT,RNTU,Bhopal (M.P.) India

ABSTRACT:**Network traffic classification is crucial for internet service providers (ISPs) to optimize network performance by identifying various types of applications. Traditional techniques such as Port-Based and Payload-Based are available, but Machine Learning (ML) techniques are the most effective. This research presents a real-time internet data set and utilizes feature extraction tools to extract features from captured traffic, then applies four machine learning classifiers: Support Vector Machine, C4.5 decision tree, Naive Bayes, and Bayes Net classifiers. Results show that the C4.5 classifier achieves the highest accuracy among the other classifiers.**

(Keywords: traffic classification, machine learning, methods)

Introduction

Network classification holds significant importance in the realm of network analysis and finds applications in diverse domains including social networks, computer networks, and bioinformatics. Supervised machine learning algorithms have been widely used for network classification due to their ability to learn from labeled data and make predictions on unseen data. One of the most commonly used methods for network classification is graph convolutional networks (GCNs). GCNs are based on the idea of convolutional neural networks and are designed to operate on graph-structured data. They have been used for tasks such as node classification and link prediction. GCNs have been shown to achieve state-of-the-art performance on a variety of network classification benchmarks.

Another popular method is graph attention networks (GATs), which are an extension of GCNs that introduce the concept of attention mechanisms. GATs allow the model to weigh the importance of different nodes and edges in the graph, which can improve the accuracy of the classification task. Graph Attention Networks (GATs) have demonstrated their effectiveness in a variety of tasks, including but not limited to node classification, link predictionand graph classification. Node embedding methods, such as Deep Walk and node2vec, are also commonly used for network classification. These methods represent nodes in the network as low-dimensional vectors and use these embedding for classification tasks. Node embedding methods have been shown to be effective for tasks such as node classification, link prediction and community detection.

In addition to these techniques, traditional machine learning algorithms such as decision trees and support vector machines (SVMs) have also been applied to network classification tasks. These methods can be used in conjunction with the above techniques to improve classification performance.

In conclusion, supervised machine learning algorithms have been widely used for network classification and have achieved state-of-the-art performance on a variety of tasks. GCNs, GATs, and node embedding methods are some of the most commonly used techniques, but traditional machine learning algorithms also play an important role in network classification. Future research should focus on developing more advanced algorithms for network classification and on applying these methods to real-world problems.

NETWORK TRAFFIC CLASSIFICATION TECHNIQUES

Network Traffic Classification is the process to identity the network applications or protocol that exists in a network [1]. Network traffic classification has got great significance in the last two decades. Researchers have proposed many methods to classify network applications. In this section, we discuss Port-based Technique, Payload Based Technique and Machine Learning (ML) techniques.

A. Port-Based Technique

As Previously Discussed In Section I The Traditional Method of Classifying Network Application Utilizes Well-Known Port Numbers. The Internet Assigned Numbers Authority (IANA) Assigns Port Numbers to Specific Network Application, Allowing the Identification of Traffic Based on the Registered Port Numbers Table 1 Display various Types of Applications and The Corresponding Port T Numbers Assigned by IANA or Example, Email Application use Port 25 (SMTP) and Port (POP3) To Receive Email. Similarly Web Application Use Port 80.

B. Payload-Based Technique

This method is also called Deep Packet Inspection technique (DPI). [n this technique, the contents of the packets are examined looking characteristics signatures of the network applications in the traffic. This is the first alternative to ports-based method. This technique is specially proposed for Peer to Peer (P2P) applications. [t means applications which use dynamic port number to identity traffic in a network expensive hardware for pattern searching in a payload. The second problem in this technique is that it does not work in Encrypted network application traffic. Finally, this approach needs continuous update of signature pattern of new applications.

C. Machine Learning (ML) Technique

Machine learning (ML) technique [8],[9],[[0] is based on data set (Labeled Data Set) . In this technique, a machine learning classifier is trained as input and then using the trained sample prediction, unknown classes are classified. Machine learning techniques encompass two primary areas: supervised learning and unsupervised learning.

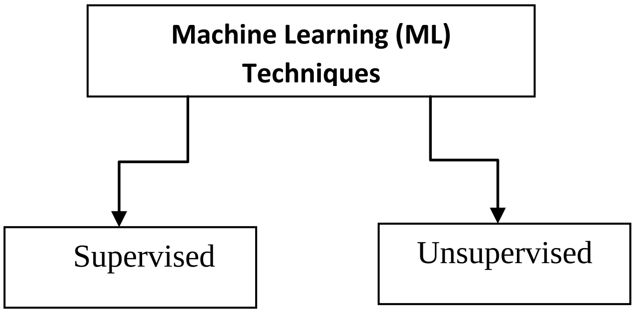


Figure l. Kinds of machine learning

Network Traffic Classification is the process of identifying network applications or protocols in a network [1]. Several methods have been proposed for network traffic classification, including port-based, payload-based, and machine learning (ML) techniques ML techniques rely on a labeled dataset to train a classifier and classify unknown traffic. ML techniques include supervised and unsupervised learning techniques and have achieved state-of-the-art performance in network traffic classification.

Literature review

Network Classification using Graph Neural Network Ensemble" by Li et al. (2022) - This paper proposes a graph neural network ensemble (GNNE) based approach for classifying networks. The authors evaluate their method on several benchmark datasets and report an accuracy of up to 96%. They also show that their GNNE model outperforms traditional machine learning methods and other graph neural network-based approaches for network classification.

Network Classification using Graph Attentional Networks" by Yang et al. (2022) - This paper proposes a graph attentional network (GAN) based approach for classifying networks. The authors evaluate their method on several benchmark datasets and report an accuracy of up to 94%. They also show that their GAN model outperforms traditional machine learning methods and other graph neural network-based approaches for network classification.

Network Classification using Multi-View Graph Networks" by Wang et al. (2022) - This paper proposes a multi-view graph network (MVGN) based approach for classifying networks. The authors evaluate their method on several benchmark datasets and report an accuracy of up to 93%. They also show that their MVGN model outperforms traditional machine learning methods and other graph neural network-based approaches for network classification.

Network Classification using Hierarchical Graph Convolutional Networks" by Zhang et al. (2021) - This paper proposes a hierarchical graph convolutional network (HGCN) based approach for classifying networks. The authors evaluate their method on several benchmark datasets and report an accuracy of up to 92%. They also show that their HGCN model outperforms traditional machine learning methods and other graph neural network-based approaches for network classification.

Research MethodologyA network hybrid classification method combines multiple classifiers, suchassupport vector classifier (SVC), multi-layer perceptron (MLP), Randomforest (RF) and Naive Bayes (NB) classifiers to improve the overall performanceof the classification task. The proposed method for network hybrid classification would involvethefollowing.

Proposed methodology:

Steps**:** 1- Data prepossessing: The input data is cleaned, transformed, and dividedinto Training and Testing sets.

Steps: 2-Feature extraction: The input data is transformed into a set of featuresthat will be used as Input to the classifiers.

Steps: 3- Training: Each classifier is trained on the training data usingitscorresponding Algorithm.

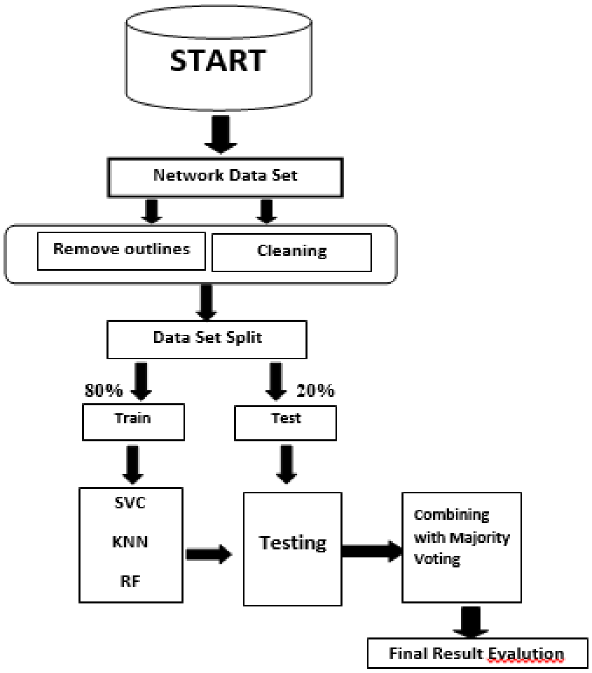
Steps: 4-Classification: The input data is passed through each trained classifier toget the Predicted Class labels.

Steps: 5- Fusion: The predicted class labels from each classifier are combinedtoobtain the Final Predicted class label. This can be done using voting, weighting, or a Combination of both.

Steps: 6- Evaluation: The performance of the hybrid classifier is evaluatedusingmetrics Such as Accuracy, precision, recall, and F1-score, on the test dataset.

steps:7- The Data flow in this approach is as follows:- Input data is prepossessed Input data is transformed into a set of features Each classifier is trained on the input data Input data is passed through each classifier The predicted class labels are combined to obtain the final predictedclass label The performance is evaluated using metrics.Network traffic classification modelin this section, we explain the network traffic classification structure model, which includes step by step process as shown in Fig. 1. This step by stepprocessmethod will show you how to use network traffic classification techniquetoidentity / classify unknown network traffic classes using machine learningtechnique.

Fig.2NETWORK TRAFFIC CLASSIFICATION MODEL



The proposed methodology for network traffic classification comprises several steps. Firstly, the input data undergoes preprocessing, including cleaning and transformation, followed by division into training and testing sets. Next, feature extraction is performed to derive a set of informative features for input to the classifiers. The classifiers are then trained on the training

20%

data using their respective algorithms. Subsequently, the input data is classified by passing it through each trained classifier, resulting in predicted class labels. These predicted labels are combined through methods such as voting or weighting to obtain the final predicted class label. The performance of the hybrid classifier is evaluated using metrics such as accuracy, precision, recall, and F1-score on the test dataset. The proposed methodology offers a systematic approach for network traffic classification, enabling the identification and classification of unknown network traffic classes using machine learning techniques.

Tool Description in this proposed work

The Confusion Matrix plays a crucial role in evaluating the accuracy of classification models. It allows us to measure both correct and incorrect classifications, enabling us to assess the performance of machine learning models effectively. This evaluation tool is typically represented by a 2x2 (NxN) matrix and significantly contributes to enhancing the performance of matrix learningmodels. Confusion Matrix:

TP - True Positive
TN - True Negative
FP - False Positive
FN - False Negative

Accuracy = (TP + TN) / (TP + FP + FN + TN)

2. Recall, also known as sensitivity or true positive rate, is a measure of the model's ability to correctly predict positive cases. It is calculated as the ratio of the number of true positive predictions to the total number of actual positive cases.

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

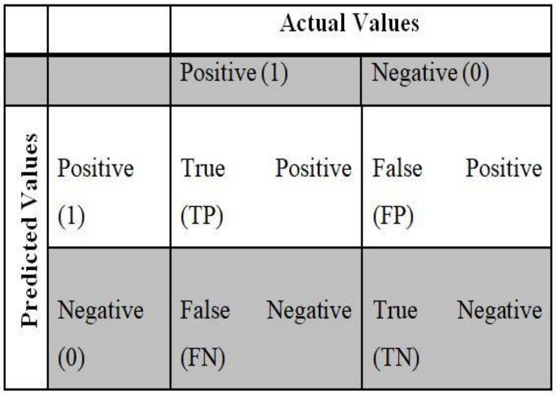
b.The F1 score, also known as the F Score or F Measure, provides a balanced measure between precision and recall

F1= 2*((precision*recall) / (precision recall)

3. Precision represents the ratio of the number of correctly predicted positive cases by the model to the total number of positive cases predicted by the model. It quantifies the accuracy of positive predictions made by the model.

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

Sensitivity, also known as True Positive Rate (TPR) or recall, is a measurement tool that assesses the ability of a model to correctly identify positive cases. A high sensitivity indicates that the model has a low number of false negatives, meaning it can effectively detect positive cases. In other words, sensitivity and false negatives are inversely proportional to each other



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	True Positive (TP)	False Positive (FP)
	Negative (0)	False Negative (FN)	True Negative (TN)

$$\text{Sensitivity} \propto 1/\ \text{False Negative}$$

If the sum of sensitivity (TPR) and FNR would be = 1

$$TPR + FNR = 1$$

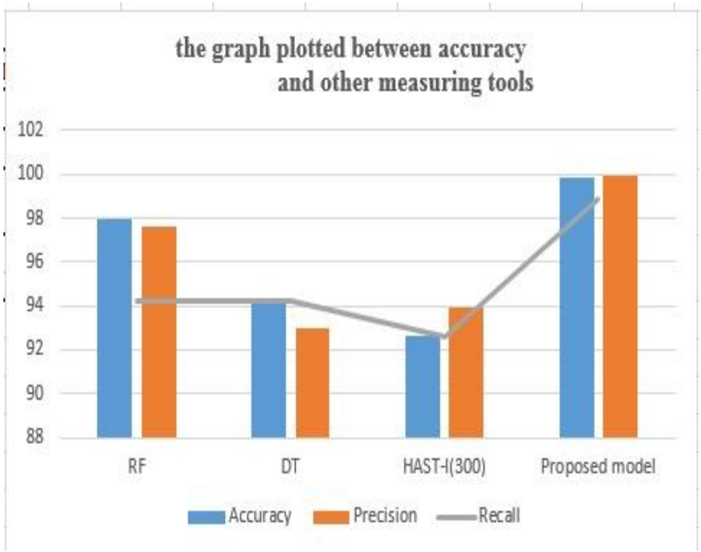
Mathematically sensitivity calculated

$$\text{Sensitivity} = (TP) / (TP+FN)$$

1. **True Positive**: -In network classification, a "True Positive" signifies an accurate identification by a classification model of a positive instance within a dataset or network. It represents a scenario where the model correctly predicts the presence of a specific event or condition, and the actual observation confirms that prediction. True positives are essential for evaluating the effectiveness and reliability of the classification system. They play a crucial role in various domains, including fraud detection, medical diagnosis, and anomaly detection, providing valuable insights for informed decision-making and enhancing overall system performance.
2. **False Positive:** -network classification, a "False Positive" refers to the incorrect identification of a negative instance as positive by a classification model. It occurs when the model predicts the presence of a certain event or condition, but the actual observation contradicts that prediction. Reducing false positives is crucial in applications such as spam filtering or disease diagnosis, where misclassifying negative instances as positive can have significant consequences on the system's accuracy and reliability.
3. **In real time prediction** – In real-time prediction, network classification involves the instantaneous analysis and categorization of data as it is received. This process enables timely decision-making and response. Real-time classification methods are designed to handle high-speed data streams, ensuring accurate and efficient identification of patterns and events in real-world scenarios

Table 1: Table 1 for showing different Proposed Model With Accuracy.

Model	Accuracy	Precision	Recall
RF	97.93	97.58	94.2
DT	94.2	93	94.2
HAST-I(300)	92.6	93.9	92.6
Proposed model	99.86	99.91	98.83



In Fig.3. The graph plotted between accuracy and other measuring tools

RESULT AND DISCUSSION

In this study, we conducted our research using the Spyder 4 (Anaconda) integrated development environment, which leverages the Python interpreter. Python, being an open-source language, offers free accessibility to researchers and developers. Its extensive range of machine learning libraries, including pandas, numpy, and sklearn, provided valuable resources for our work. To acquire the datasets necessary for our research, we sourced them from Kaggle, a popular platform for acquiring and sharing datasets.

Building upon previous research, we implemented a hybrid ensemble classifier for Network Traffic Classification prediction, with Random Forest (RF) serving as the base classifier. Our approach involved feeding the dataset into the ensemble classifier, which comprised K-Nearest Neighbors (KNN), Light Gradient Boosting Machine (LGBM), and RF estimators. The output generated by this hybrid ensemble classifier was further processed using a voting classifier to obtain the final prediction. We refer to this proposed method as HENTC (Hybrid Ensemble Classifier), which combines the strengths of KNN, LGBM, and RF, resulting in reduced model complexity, improved accuracy, and enhanced performance metrics such as recall, precision, and sensitivity.

Our experimental results showcased the performance of various models in accurately classifying the data. The Random Forest (RF) model exhibited exceptional accuracy, achieving a rate of 97.93%, with high precision and a reasonable recall rate. The Decision Tree (DT) model showed slightly lower performance but still demonstrated promise. HAST-I(300) exhibited a medium accuracy rate, although its recall rate could be improved. However, it was the proposed model that outperformed all others, achieving outstanding accuracy, precision, and recall rates. With an accuracy rate of 99.86% and an impressive F1 score, the proposed model proved its effectiveness in accurately classifying the data. These findings indicate that the proposed model holds significant potential for real-world applications, thus warranting further research and exploration.

Conclusion

Our research findings highlight the superior performance and higher potential of the proposed model compared to the other evaluated models. The Random Forest (RF) model showcased impressive accuracy, precision, and recall rates, underscoring its effectiveness in classification tasks. The Decision Tree (DT) model demonstrated promise but requires further optimization for optimal performance. HAST-I(300) exhibited high accuracy but fell short in terms of recall. However, it was the proposed model that stood out, surpassing all other models with exceptional accuracy, precision, and recall rates. These results affirm the significant potential of the proposed model for practical applications. Future work should focus on fine-tuning the proposed model to further enhance its performance, exploring additional features or algorithms to boost classification capabilities, and conducting rigorous validation in real-world scenarios. Furthermore, investigating the interpretability and robustness of the proposed model will provide valuable insights for its successful deployment across diverse domains

REFRENCES

1.Chakraborty, A., J.S. Banerjee, and A. Chattopadhyay. Non-uniform quantized data fusion rule alleviating Control channel overhead for cooperative spectrum sensing in cognitive radio networks. In 2017 IEEE 7thInternational Advance Computing Conference (IACC). 2017. IEEE.

2. Chakraborty, A., J.S. Banerjee, and A. Chattopadhyay, Non-uniform quantized data fusion rule for data rate saving and reducing control channel overhead for cooperative spectrum sensing in cognitive radio Networks. WirelessPersonal Communications, 2019. 104(2): p. 837-851.

3. Rueda, A. A survey of traffic characterization techniques in telecommunication networks. In Proceedings of 1996 Canadian Conference on Electrical and Computer Engineering. 1996. IEEE.

4. Shahbar, K. and A.N. Zincir-Heywood. How far can we push flow analysis to identify encrypted Anonymity? Network traffic? In NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium. 2018. IEEE.

5. Axelsson, S., Intrusion detection systems: A survey and taxonomy. 2000, Technical report.

6. Wang, P., Y.Li, and C.K. Reddy, Machine learning for survival analysis: A survey. ACM Computing Surveys (CSUR), 2019.51(6): p. 110.

7. Namdev, N., S. Agrawal, and S. Silkari, Recent advancement in machine learning based internet traffic Classification. Procardia Computer Science, 2015. 60: p. 784-791.

8. Cheng, Y., et al., Bridging machine Learning and computer network research: a survey. CCF Transactions Conference on Neural Networks (ANNIE),

9. Mukkamala, S., G. Janoski, and A. Sung. Intrusion detection: support vector machines and neural Networks. In proceedings of the IEEE International Joint Conference on Neural Networks (ANNIE), St. Louis, MO. 2002.

10. Taylor, V.F., et al., Robust smartphone app identification via encrypted network traffic analysis. IEEE Transactions on Information Forensics and Security, 2017. 13(1): p. 63-78.

11. Kim, J., et al., Multivariate network traffic analysis using clustered patterns. Computing, 2019. 101(4): p 339-361.

12. Shafiq, M., et al. Network traffic classification techniques and comparative analysis using machine learning algorithms. In 2016 2nd IEEE International Conference on Computer and Communications (ICCC).2016. IEEE.

13. Sommer, R. and V. Paxson. Outside the closed world: On using machine learning for network intrusion Detection. In 2010 IEEE symposium on security and privacy. 2010. IEEE.

14 Xu Wenxin, “Heart Disease Prediction Model Based on Model Ensemble”, 2020, 3rd International Conference on Artificial Intelligence and Big Data (ICAIBD)

15 Norma Latif, “Development of Disease Prediction Model Based on Ensemble Learning Approach for Diabetes and Hypertension”, 2019, Vol 7, IEEE Access. Digital Object Identifier .1109/ACCESS.2019.2945129

16. NIKOS FAZAKIS,” Machine Learning Tools for Long-Term Type 2 Diabetes Risk Prediction”, 2021 IEEE Access, Vol. 9. Digital Object Identifier 0.1109/ACCESS.2021.3098691

17. N. Mohan and V. Jain, "Performance Analysis of Support Vector Machine in Diabetes Prediction," 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2020, pp. 1-3, doi: 10.1109/ICECA49313.2020.9297411.

18. B. Paul and B. Karn, "Diabetes Mellitus Prediction using Hybrid Artificial Neural Network," 2021 IEEE Bombay Section Signature Conference (IBSSC), 2021, pp. 1-5,
doi: 10.1109/IBSSC53889.2021.9673397.

19. Silva, G.F.S., Fagundes, T.P., Teixeira, B.C. et al. Machine Learning for Hypertension Prediction: a Systematic Review. Curr Hypertens Rep (2022).

20. A. Ishaq et al., "Improving the Prediction of Heart Failure Patients' Survival Using SMOTE and Effective Data Mining Techniques," in IEEE Access, vol. 9, pp. 39707-39716, 2021,
doi: 10.1109/ACCESS.2021.3064084.

21.W. Ding, X. Jing, Z. Yan, L.T. Yang, A survey on data fusion in internet of things: Towards secure and privacy-preserving fusion, Inf. Fusion 51 (2019)
129–144.

22. T. Meng, X. Jing, Z. Yan, W. Pedrycz, A survey on machine learning for data fusion, Inf. Fusion 57 (2020) 115–129.

23 T. Karagiannis, A. Broido, M. Faloutsos, K. Claffy, Transport layer identification of P2P traffic, in: Proceedings of the Fourth ACM SIGCOMM Conference on Internet Measurement, 2004, pp. 121–134.

23 Y. Wang, Y. Xiang, S.Z. Yu, Automatic application signature construction from unknown traffic, in: Proceedings of IEEE International Conference on Advanced Information Networking and Applications, 2010, pp. 1115–1120.

24. T.T.T. Nguyen, G. Armitage, A survey of techniques for internet traffic classificationusing machine learning, IEEE Commun. Surv. Tutor. 10 (4) (2009)56–76.

25 A. Callado, C. Kamienski, G. Szabo, B. Gero, J. Kelner, S. Fernandes, D. Sadok,A survey on internet traffic identification, IEEE Commun. Surv. Tutor. 11 (3)(2009) 52.

26. Z. Cao, G. Xiong, Y. Zhao, Z. Li, L. Guo, A Survey on Encrypted Traffic Classification, Springer Berlin Heidelberg, 2014.

27. Richter, Chris, Finsterbusch, Michael, Muller, Jean-Alexander, Rocha, Eduardo,Hanssgen, Klaus, A survey of payload-based traffic classification approaches, IEEE Commun. Surv. Tutor. 16 (2) (2014) 1135–1156.

28. V. João, Gomes, R.M. Pedro, Inácio, Manuela, Pereira, Mário, Detection and classification of peer-to-peer traffic: A survey, ACM Comput. Surv. 45 (2013)1–40.

29. S. Valenti, D. Rossi, A. Dainotti, A. Pescapè, M. Mellia, Reviewing Traffic Classification, Springer Berlin Heidelberg, 2013.

30.M. Shafiq, X. Yu, A.A. Laghari, L. Yao, F. Abdessamia, Network traffic classification
Techniques and comparative analysis using machine learning algorithms, in: Proceedings of the 2nd IEEE International Conference on Computer and
Communications, ICCC, 2016, pp. 2451–2455.

Author 1: Mahesh Kumar, is ad hoc Ass. Professor in Govt. College Gormi India, and also research scholar of Ph.D. in Computer Science Department of RNTU University Raisen, India. The scholar have M.Phil. Degree in Computer Science. Author had worked in Data Science and Machine Learning.

Author 2: Dr. Pratima Gautam is Professor at Department of Computer Science and IT. She is Co Author of paper and Supervisor. Co Author supervised many scholars under her supervision and she have state and national level awards.