International Scientific Journal of Engineering and Management
Volume: 03 Issue: 04 | April – 2024
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

Internet Traffic Classification Using Machine Learning Techniques

• Author - Debmalya Ray

• Email - debmalya.ray9989@gmail.com

1. Abstract:

Internet traffic classification is a fundamental task for network services and management. There are good machine learning models to identify the class of traffic. However, finding the most discriminating features to have efficient models remains essential. In this paper, we use interpretable machine learning algorithms such as random forest and gradient boosting to find the most discriminating features for internet traffic classification. This paper aims to overcome these challenges by proposing machine learning classification mechanism.

Index Terms:

Supervised Learning, Classification Problems, Feature Engineering, Scaling, Python, Material Type.

2. Introduction:

As a data-driven method, machine learning provides reliable and accurate performance in solving problems in materials science. Research concerning the prediction of various properties of materials by machine learning has continued.

In other words, machines can learn from past data and situations, and improve algorithms to make decisions when encountering different and even unknown situations.

The important feature variables used to determine the 'Action' are as follows:

- 'Source Port'
- 'Destination Port'

ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

- ' NAT Source Port'
- ' NAT Destination Port'
- ' Bytes'
- ' Bytes Sent'
- ' Bytes Received'
- 'Packets'
- 'Elapsed Time (sec)'
- 'pkts_sent'
- 'pkts_received'

Using these data, we will try to use analytical techniques and ML algorithms to determine the 'Action' broadly divided into classes **{ allow, deny, drop, reset-both }.**

3. Related Work

In the last decade, numerous research has been conducted for Internet Traffic Classification, such as port-based, bytes-based, internet packet based and machine-learning-based approaches. Most of the researchers have focussed on building right exploratory analysis with correct feature engineering solutions and identifying the best machine learning algorithm.

Author	Title	Summary
Muhammad Basit Umair1 , Zeshan Iqbal1 , Muhammad Bilal2 , Jamel Nebhen4 , Tarik Adnan Almohamad3 and Raja Majid Mehmood5,*	An Efficient Internet Traffic Classification System Using Deep Learning for IoT	Internet of Things (IoT) defines a network of devices connected to the internet and sharing a massive amount of data between each other and a central location. These IoT devices are connected to a network therefore prone to attacks. Various management tasks and network operations such as security, intrusion detection, Quality-of-Service provisioning, performance monitoring, resource provisioning, and traffic engineering require traffic classification. Due to the ineffectiveness of traditional classification schemes, such as port-based and payload-based methods, researchers proposed machine learning-based traffic classification systems based on shallow neural networks. Furthermore, machine learning-based models incline to misclassify internet traffic due to improper feature selection. In this research, an

ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

		efficient multilayer deep learning based classification system is presented to overcome these challenges that can classify internet traffic. To examine the performance of the proposed technique, Moore-dataset is used for training the classifier. The proposed scheme takes the pre-processed data and extracts the flow features using a deep neural network (DNN). In particular, the maximum entropy classifier is used to classify the internet traffic. The experimental results show that the proposed hybrid deep learning algorithm is effective and achieved high accuracy for internet traffic classification, i.e., 99.23%. Furthermore, the proposed algorithm achieved the highest accuracy compared to the support vector machine (SVM) based classification technique and k-nearest neighbours (KNNs) based classification technique.
Hyunsu Mun and Youngseok Lee *	Internet Traffic Classification with Federated Learning	Traffic classification utilizing flow measurement enables operators to perform essential network management. Flow accounting methods such as NetFlow are, however, considered inadequate for classification requiring additional packet-level information, host behaviour analysis, and specialized hardware limiting their practical adoption. This paper aims to overcome these challenges by proposing two-phased machine learning classification mechanism with NetFlow as input. The individual flow classes are derived per application through k-means and are further used to train a C5.0 decision tree classifier. As part of validation, the initial unsupervised phase used flow records of fifteen popular Internet applications that were collected and independently subjected to k-means clustering to determine unique flow classes generated per application. The derived flow classes were afterwards used to train and test a supervised C5.0 based decision tree. The resulting classifier reported an average accuracy of 92.37% on approximately 3.4

ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

		million test cases increasing to 96.67% with adaptive boosting. The classifier specificity factor which accounted for differentiating content specific from supplementary flows ranged between 98.37% and 99.57%. Furthermore, the computational performance and accuracy of the proposed methodology in comparison with similar machine learning techniques lead us to recommend its extension to other applications in achieving highly granular real-time traffic classification.
Taimur Bakhshi and Bogdan Ghita	On Internet Traffic Classification: A Two-Phased Machine Learning Approach	As Internet traffic classification is a typical problem for ISPs or mobile carriers, there have been a lot of studies based on statistical packet header information, deep packet inspection, or machine learning. Due to recent advances in end-to-end encryption and dynamic port policies, machine or deep learning has been an essential key to improve the accuracy of packet classification. In addition, ISPs or mobile carriers should carefully deal with the privacy issue while collecting user packets for accounting or security. The recent development of distributed machine learning, called federated learning, collaboratively carries out machine learning jobs on the clients without uploading data to a central server. Although federated learning provides an on-device learning framework towards user privacy protection, its feasibility and performance of Internet traffic classification have not been fully examined. In this paper, we propose a federated-learning traffic classification protocol (FLIC), which can achieve an accuracy comparable to centralized deep learning for Internet application identification without privacy leakage.
Hamza Awad Hamza Ibrahim; Omer Radhi Aqeel Al Zuobi; Marwan A. Al-Namari; Gaafer MohamedAli;	Internet traffic classification using machine learning approach: Datasets validation issues	Internet traffic classification is an area of current research interest. The failure of port and payload based classification motivates researchers to head

ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

		towards a machine learning (ML) approach. However, training and testing dataset validation has not been formally addressed. This paper discusses the problem of ML dataset validation and highlights three training issues to be considered in ML classification. The first issue is when training and testing datasets collected from same or different network characteristics. The second issue considers training dataset classes whose real online traffic classes are not presented. The third issue is the geographic place where the network traffic is captured. Real Internet traffic datasets collected from a campus network are used to study the traffic features and classification accuracy for each validation training issue. The experimental results demonstrate that there are differences in some traffic features such as inter-arrival time when training and testing data were collected from different networks. Furthermore, the experiment of the second issue shows that the online classifier achieved the highest accuracy (92.22%) when the ML classifier was trained by dataset classes which have the same ratio of the real online traffic. For the geographic capturing level, the results indicate that there is a difference in the traffic statistical features when the capturing level is different.
Mohammad Reza Parsaei, Mohammad Javad Sobouti, Seyed Raouf khayami, Reza Javidan	Network Traffic Classification using Machine Learning Techniques over Software Defined Networks	Nowadays Internet does not provide an exchange of information between applications and networks, which may results in poor application performance. Concepts such as application-aware networking or network-aware application programming try to overcome these limitations. The introduction of Software-Defined Networking (SDN) opens a path towards the realization of an enhanced interaction between networks and applications.

ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

		SDN is an innovative and programmable networking architecture, representing the direction of the future network evolution. Accurate traffic classification over SDN is of fundamental importance to numerous other network activities, from security monitoring to accounting, and from Quality of Service (QoS) to providing operators with useful forecasts for long-term provisioning. In this paper, four variants of Neural Network estimator are used to categorize traffic by application. The proposed method is evaluated in the four scenarios: feedforward; Multilayer Perceptron (MLP); NARX (Levenberg Marquardt) and NARX (Naïve Bayes). These scenarios respectively provide accuracy of 95.6%, 97%, 97% and 97.6%.
Syifa Maliah Rachmawati; Dong-Seong Kim; Jae-Min Lee	Machine Learning Algorithm in Network Traffic Classification	Network traffic classification plays an important role in various network functions such as network security issues and network management. In addition to port-based and payload-based approaches, the classical machine learning approaches have been studied for past decades, but there are some limitations, namely time-consuming, frequent features updates, and the accuracy has decreased due to the rise of internet traffic, especially encrypted traffic. Deep learning comes with the ability of automatic feature learning, some studies try to apply it and reported better accuracy. This survey paper introduces the emerging research and general framework for deep learning-based methods for traffic classification. We present commonly used deep learning methods and their application in traffic classification tasks.

ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

4. Problem Understanding and Proposed Methodology

Before solving any data science, it is important to understand whether it is a data science problem or not. Hence we follow build hypothesis and try to test the statement with rightful evidence. Usually, it is denoted as:

H0: Null Hypothesis: Allow the internet traffic.

H1: Alternate Hypothesis: Not Allow the internet traffic.

Methodologies:

After formulating a hypothesis based on a problem statement in a Data Science (DS) project, the subsequent steps typically follow a structured project lifecycle. While different organizations or teams may have variations, here are the common steps constructed within a DS project lifecycle:

1. **Problem Definition:** Clearly define the problem statement and objectives of the project. This involves understanding stakeholder needs, defining success criteria, and framing the business problem in a way that can be addressed using data.
2. **Data Collection:** Gather relevant data sources necessary for the analysis. This could involve data from internal databases, external sources, APIs, or other data providers. Ensure data quality, completeness, and relevance to the problem at hand.
3. **Data Cleaning and Preprocessing:** Clean the data to handle missing values, outliers, duplicates, and inconsistencies. Preprocess the data to transform it into a format suitable for analysis, including feature engineering, normalization, and scaling.
4. **Exploratory Data Analysis (EDA):** Explore the data to gain insights, understand patterns, correlations, and relationships between variables. Visualize the data using charts, graphs, and statistical summaries to identify trends and anomalies.
5. **Feature Selection and Engineering:** Select relevant features that contribute most to the predictive power of the model. Engineer new features if necessary to enhance model performance.
6. **Model Development:** Select appropriate machine learning or statistical models based on the problem type (e.g., classification, regression, clustering). Train and evaluate the models using appropriate techniques such as cross-validation, hyperparameter tuning, and model selection.
7. **Model Evaluation:** Evaluate the performance of the models using appropriate evaluation metrics, considering factors such as accuracy, precision, recall, F1-score, or others depending on the problem domain.
8. **Model Deployment:** Deploy the trained model into production or implement it into business processes to make predictions or generate insights. This may involve integrating the model into existing systems or developing APIs for real-time predictions.

ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

9. **Monitoring and Maintenance:** Continuously monitor the performance of the deployed model in production. Update the model periodically with new data and retrain if necessary to ensure it remains accurate and relevant over time.
10. **Documentation and Reporting:** Document the entire process, including data sources, methodologies, assumptions, and decisions made throughout the project lifecycle. Prepare reports or presentations to communicate findings, insights, and recommendations to stakeholders.
11. **Feedback and Iteration:** Gather feedback from stakeholders and end-users, iterate on the model or analysis based on feedback, and refine the solution to address any emerging issues or changing requirements.

By following these steps within the DS project lifecycle, teams can effectively tackle data-driven problems, derive actionable insights, and deliver value to stakeholders.

5. Results and Discussion

The entire result is obtained using basic system/software requirement mentioned as follows:

a) Intel Core i5 CPU @ 2.30 GHz
b) 64-bit OS, 64-based processor on Windows 10
c) Python – 3.6.5 and ML / DS Libraries
d) Microsoft VS code / Jupyter Notebook / Deep Note
e) Github / Gitlab
f) Render – SAAS Platform

The dataset is divided as a 7:3 train-test split. The division of the dataset into 70% of the training set and 30% of the test set is randomized.

Evaluation Metrics

There are four evaluation metrics in the proposed methodology. The explanation of each metric is given in Tab.

$$precision = \frac{TP}{TP + FP} \quad (1)$$

$$recall = \frac{TP}{TP + FN} \quad (2)$$

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

$$f1 - score = 2 * \frac{precision * recall}{precision + recall} \quad (4)$$

International Scientific Journal of Engineering and Management
Volume: 03 Issue: 04 | April – 2024
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

Experimental Results

We performed the experiment with random forest and hyper parameters tuned to obtain the below satisfactory results.

	precision	recall	f1-score	support
0	1.00	1.00	1.00	11310
1	1.00	0.54	0.70	8350
2	0.00	0.00	0.00	0
3	0.00	0.00	0.00	0
accuracy			0.80	19660
macro avg	0.50	0.38	0.42	19660
weighted avg	1.00	0.80	0.87	19660

Overall discussions

Each algorithm to be used has its particularities, weaknesses, and strengths. The emphasis was on `Random Forest` classifier with a sample dataset which provide us reasonable accuracy. However, there are few cons to take care off. These are:

1. Increase in time complexity
2. Require more data to be provided to the model to check its latency
3. Need to be validated with boosting techniques to trade-off between accuracy and speed.

6. Plots Derived from Metrics

A confusion matrix is a matrix that summarizes the performance of a machine learning model on a set of test data. It is a means of displaying the number of accurate and inaccurate instances based on the model's predictions. It is often used to measure the performance of classification models, which aim to predict a categorical label for each input instance.

International Scientific Journal of Engineering and Management
Volume: 03 Issue: 04 | April – 2024
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

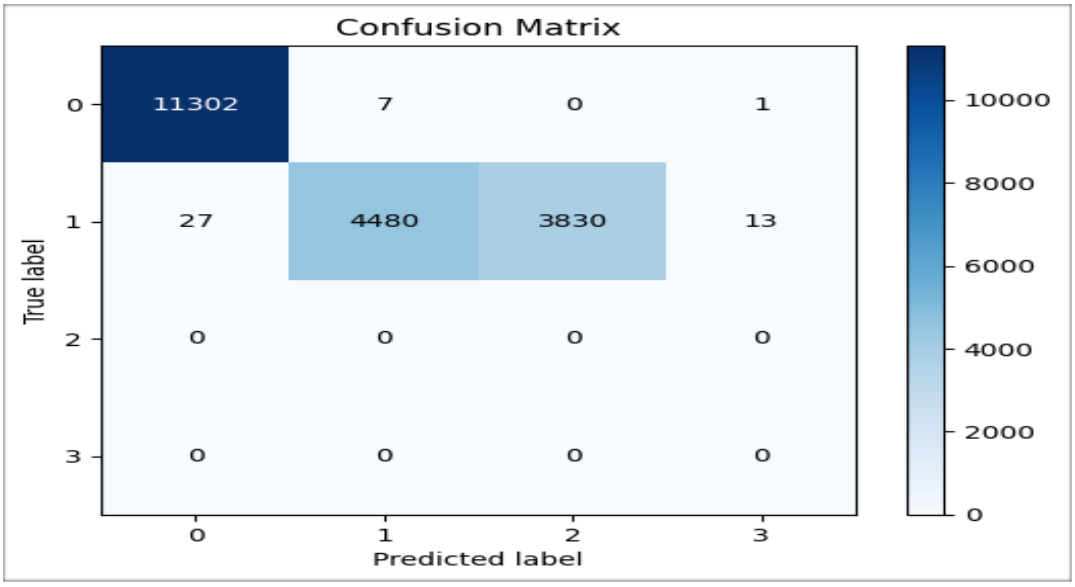

Figure: Confusion Matrix

An ROC curve (receiver operating characteristic curve) is a graph showing the performance of a classification model at all classification thresholds.

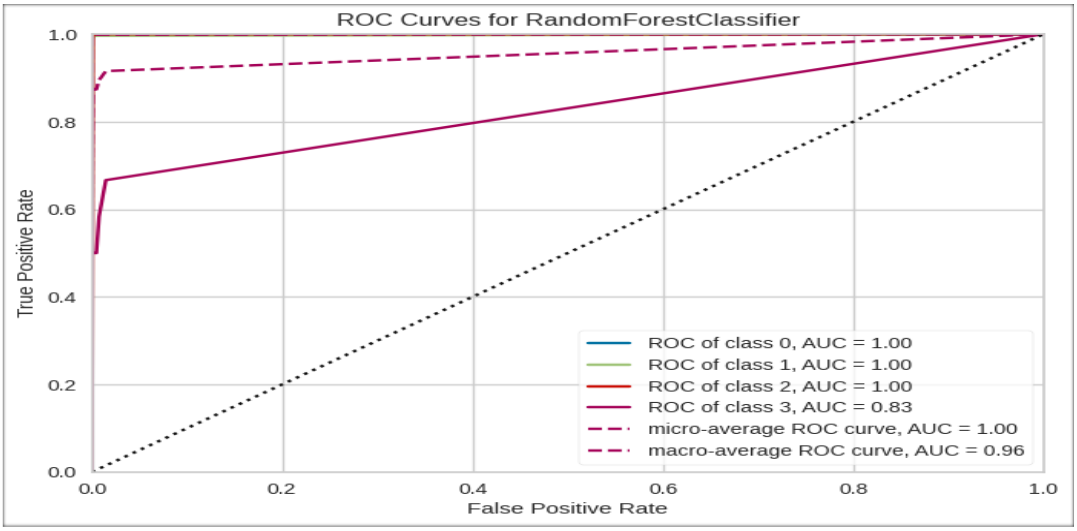

Figure: ROC Curves For Random Forest Classifier

7. Data availability statement

Available online at:

https://github.com/DebmalyaRay9989/traffic_classification_app/blob/main/log2.csv

ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

8. Code availability

The codes for analysing the data from mapping the feature variables (Usage Class, Title, Subjects, Publisher etc) to the target variable (Network Class Type) using both Train and Test Data are available on GitHub:

https://github.com/DebmalyaRay9989/traffic_classification_app/tree/main

9. Proof of Concept:

The application is created and deployed in the SAAS platform. The URL of the application is:

https://internet-traffic-classification.onrender.com

For General User: admin/admin

10. Authors contributions:

All authors take part in the discussion of the work described in this paper. All authors read and approved the final manuscript.

11. Informed consent statement:

Not applicable

12. Conflicts of interest:

The authors declare no conflicts of interest.

13. Funding:

None

14. Abbreviations:

The following abbreviations are used in this manuscript:

ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

ML	Machine Learning
LR	Logistic Regression
RF	Random Forest
SVM	Support Vector Machine
KNN	K Nearest Neighbor
XAI	Explainability AI
TP	True Positive
FP	False Positive
TN	True Negative
FN	False Negative

15. References:

1. H. Tahaei, F. Afifi, A. Asemi, F. Zaki and N. B. Anuar, “The rise of traffic classification in IoT networks: A su

rvey,” Journal of Network and Computer Applications, vol. 154, pp. 102538, 2020.

2. Y. Yue, S. Li, P. Legg and F. Li, “Deep learning-based security behaviour analysis in IoT environments: A sur

vey,” Security and Communication Networks, vol. 2021, no. 1, pp. 1–13, 2021.

3. M. Abbasi, A. Shahraki and A. Taherkordi, “Deep learning for network traffic monitoring and analysis (NTM

AA survey,” Computer Communications, vol. 170, pp. 19–41, 2021.

4. I. Cvitić, D. Peraković, M. Periša and M. Botica, “Novel approach for detection of IoT generated DDoS traffi

c,” Wireless Networks, vol. 27, no. 3, pp. 1573–1586, 2021.

5. T. T. T. Nguyen and G. J. Armitage, “A survey of techniques for internet traffic classification using machine l

earning,” IEEE Communications Surveys & Tutorials, vol. 10, no. 1–4, pp. 56–76, 2008.

6. IANA, Internet Assigned Numbers Authority (Accessed on 04 June 20212021. [Online]. Available: https://w

ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

ww.iana.org.

7. A. Madhukar and C. Williamson, “A longitudinal study of P2P traffic classification,” in 14th IEEE Int. Symp.

on Modeling, Analysis, and Simulation, Monterey, CA, USA, pp. 179–188, 2006.

8. A. W. Moore and K. Papagiannaki, “Toward the accurate identification of network applications,” in Int. Work

shop on Passive and Active Network Measurement, Springer, Berlin, Heidelberg, pp. 41–54, 2005.

9. H. Shi, H. Li, D. Zhang, C. Cheng and X. Cao, “An efficient feature generation approach based on deep learn

ing and feature selection techniques for traffic classification,” Computer Networks, vol. 132, pp. 81–98, 2018.

10. C. Yu, J. Lan, J. Xie and Y. Hu, “QoS-aware UTraffic classification architecture sing machine learning and d

eep packet inspection in SDNs,” Procedia Computer Science, vol. 131, no. 1, pp. 1209–1216, 2018.

11. A. Este, F. Gringoli and L. Salgarelli, “Support vector machines for TCP traffic classification,” Computer N etworks, vol. 53, no. 14, pp. 2476–2490, 2009.

12. G. Sun, T. Chen, Y. Su and C. Li, “Internet traffic classification based on incremental support vector machin es,” Mobile Networks and Applications, vol. 23, no. 4, pp. 1–8, 2018.

13. M. Shafiq, X. Yu, A. K. Bashir, H. N. Chaudhry and D. Wang, “A machine learning approach for feature sel ection traffic classification using security analysis,” The Journal of Supercomputing, vol. 74, no. 10, pp. 1–2 6, 2018.

14. L. Zhen and L. Qiong, “A new feature selection method for internet traffic classification using ml,” Physics Procedia, vol. 33, pp. 1338–1345, 2012.

15. P. Amaral, J. Dinis, P. Pinto, L. Bernardo, J. Tavares et al., “Machine learning in software defined networks: Data collection and traffic classification,” in 2016 IEEE 24th Int. Conf. on Network Protocols, Singapore, p p. 1–5, 2016.

16. J. Erman, M. Arlitt and A. Mahanti, “Traffic classification using clustering algorithms,” in Proc. of the 2006 SIGCOMM Workshop on Mining Network Data, Pisa, Italy, pp. 281–286, 2006.

17. D. Kreutz, F. M. V. Ramos, P. Verissimo, C. E. Rothenberg, S. Azodolmolky et al., “Software-defined netwo rking: A comprehensive survey,” Proc. IEEE, vol. 103, no. 1, pp. 14–76, 2015.

18. N. Namdev, S. Agrawal and S. Silkari, “Recent advancement in machine learning based internet traffic class ification,” Procedia Computer Science, vol. 60, pp. 784–791, 2015.

19. C. Zhang, X. Wang, F. Li, Q. He and M. Huang, “Deep learning-based network application classification for

ISSN: 2583-6129
DOI: 10.55041/ISJEM01429

SDN," Transactions on Emerging Telecommunications Technologies, vol. 29, no. 5, pp. e3302, 2018.

20. M. Lopez-Martin, B. Carro, A. Sanchez-Esguevillas and J. Lloret, "Network traffic classifier with convoluti

onal and recurrent neural networks for internet of things," IEEE Access, vol. 5, pp. 18042–18050, 2017.

21. G. Sun, L. Liang, T. Chen, F. Xiao and F. Lang, "Network traffic classification based on transfer learning,"

Computers & Electrical Engineering, vol. 69, no. 4, pp. 920–927, 2018.

22. S. Garg, K. Kaur, N. Kumar and J. J. P. C. Rodrigues, "Hybrid deep-learning-based anomaly detection sche

me for suspicious flow detection in SDN: A social multimedia perspective," IEEE Transactions on Multimed

ia, vol. 21, no. 3, pp. 566–578, 2019.

23. F. Ertam and E. Avcı, "A new approach for internet traffic classification: GA-WK-ELM," Measurement, vol.

95, no. 4, pp. 135–142, 2017.