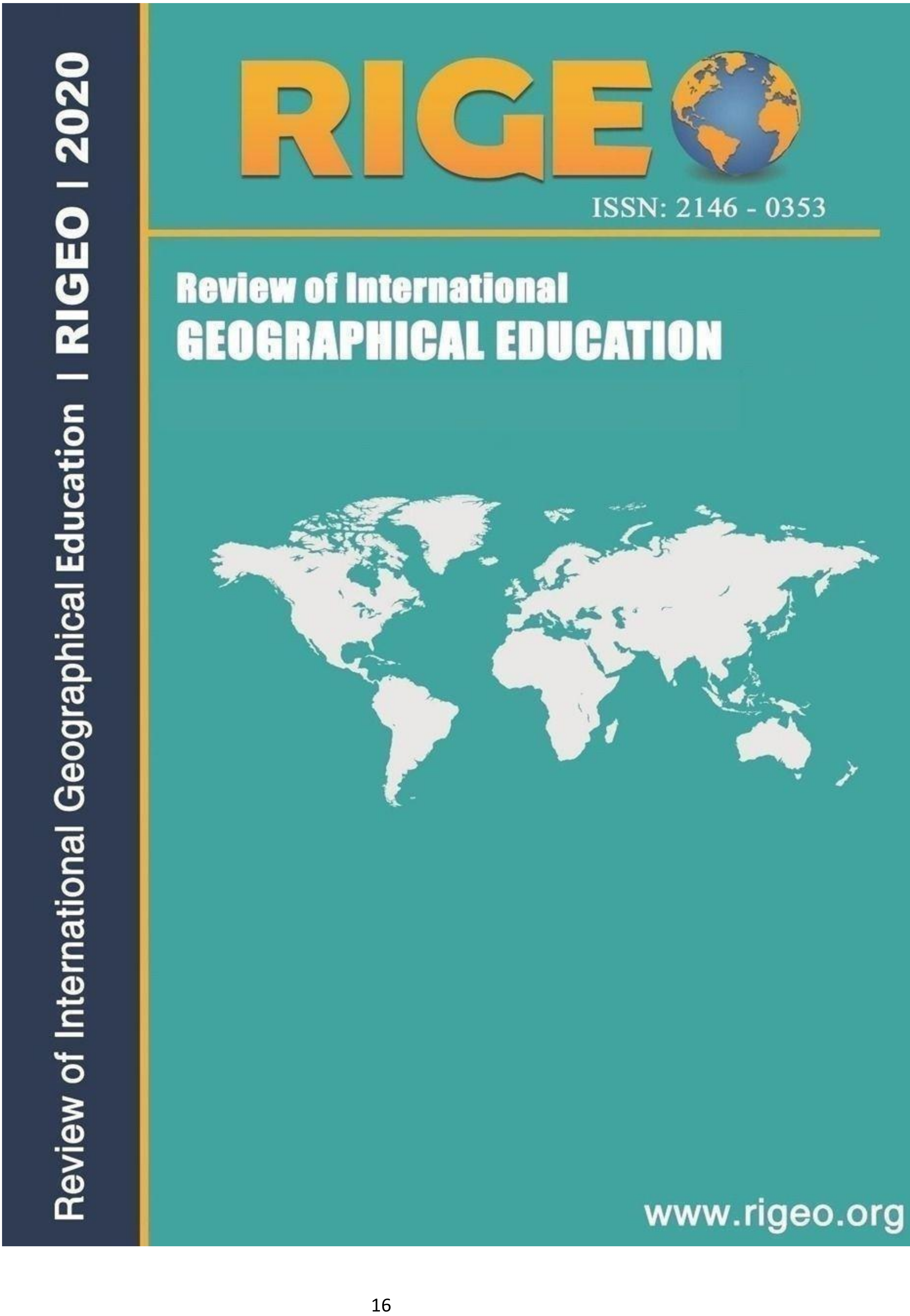


Review of International Geographical Education I RIGEO I 2020

RIGEO

ISSN: 2146 - 0353

Review of International
GEOGRAPHICAL EDUCATION

www.rigeo.org

CLASSIFICATION OF SOFTWARE DEFINED NETWORK TRAFFIC TO PROVIDE QUALITY OF SERVICE

[1] Y.Sravani Reddy, [2] B. Dharma,[3] K.Madhavi, [4] KYASAN SHIRISHA

[1,2,3] Assistant Professors,Department of Computer Science and Engineering,
Kasireddy Narayanreddy College Of Engineering And Research, Abdullapur (V),
Abdullapurmet(M), Rangareddy (D), Hyderabad - 501 505

[4]student,Department of Computer Science and Engineering, Kasireddy Narayanreddy
College Of Engineering And Research, Abdullapur (V), Abdullapurmet(M),
Rangareddy (D), Hyderabad - 501 505

ABSTRACT

Accurate traffic classification is of fundamental importance to numerous network activities, from security monitoring to accounting, and from Quality of Service to providing operators with useful forecasts for long-term provisioning. The initial stage in analysing and classifying the various types of applications running via a network is network traffic classification. This method allows network operators or internet service providers to control the overall performance of a network. We apply machine learning models to categorize traffic by application. This can be done by extracting the features of the traffic. This classified data can be used to stop unnecessary traffic and allow only user required traffic. Basically we prioritize the network traffic based on the features extracted during the classification. Features related to OTT are identified and we try to restrict them in the network for reducing the traffic in the network for providing better quality of service. We intend to stop traffic from Over-the-top(OTT) platforms like Netflix, Prime Videos, etc. Hence, by this the quality of service can be improved for user required applications.

1. INTRODUCTION

Network traffic is categorized in a variety of ways of extreme interest for both internet service providers and also network operators. It helps to classify the types of data flowing and link each one to the apps that produce it. This information is crucial for many purposes, including network monitoring, applications behavior and network security, and to improve Quality of Service(QoS). The term "Software-defined networking" (SDN) refers to a method of networking where traffic on a network is controlled by application programming interfaces (APIs) or software-based controllers that communicate with the underlying hardware infrastructure This architecture is distinct from traditional networks, which employ specialized hardware to regulate network traffic (such as switches and routers). SDN can manage traditional hardware or create and manage virtual networks through software. Although software-defined networking provides a fresh way to control how data packets are routed through a single server, network virtualization enables organizations to segment different virtual networks within a single physical network or to connect devices on different physical networks to create a single virtual network. Compared to traditional networking, SDN is far more versatile since the control plane is software-based. Without adding extra hardware, it enables administrators to manage the network, alter configuration options, supply resources, and boost network capacity from a single user interface. By providing a knowledge basis for identifying the performance levels required by applications, classification is a critical mechanism for traffic treatment. Deep Packet Inspection (DPI) and port-based classification are the two most used approaches for traffic classification. As more communication is encrypted and more apps use dynamic ports and ports for other well-known applications, these techniques are becoming outdated [1]. Machine Learning (ML), an alternate approach for traffic classification, employs the statistical characteristics of network traffic flows to address the basic issues with DPI and port-based categorization for encrypted flows. Network traffic classification is a crucial component for managing infrastructure and ensuring the QoS for various applications. In reality, a thorough traffic classification process enables the effective management of alreadyavailable network resources, enabling more precise and reliable resource allocation systems [2]. The classification of network traffic can be carried out using the features related to the OTT platforms. The feature extraction is a process of recognising features and attributes that often represent video streaming platforms or OTT platforms.

II.LITERATURE SURVEY

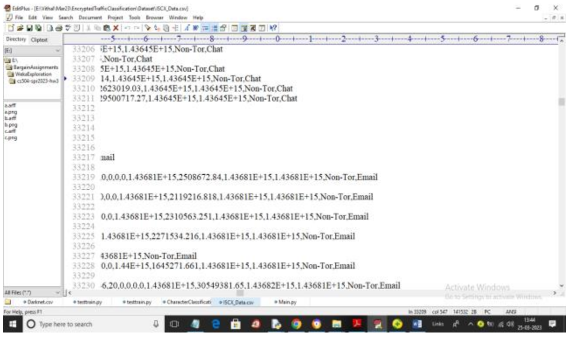
Gianni D’Angelo et al. suggested a model that begins with statistical characteristics (basic features), taken from traffic flows over a predetermined time period, and creates additional features that explain the correlations between the features (spatial features), as well as changes in those features over time (temporal features). They suggested a deep architecture made up of neural networks based on autoencoders (AEs). The autoencoders encode-decode function contain various combinations of recurrent and convolutional network layers in order to extract such information. The following combinations were looked into: CNN, LSTM, ConvLSTM, CNN-LSTM, and Stacked-CNN-LSTM. The LSTM recurrent network was used to extract temporal features, while the convolutional network was utilised to extract spatial features.

III.EXISTING SYSTEM

Traditional network management relies on static rules and predefined policies to classify traffic and manage quality of service (QoS). These systems typically use packet headers and fixed attributes such as IP addresses, ports, and protocols to categorize network traffic. Network devices like routers and switches apply these static rules to enforce QoS policies, including traffic shaping, prioritization, and bandwidth allocation. However, this approach has several limitations. Static classification lacks flexibility, often failing to adapt to changing traffic patterns or dynamic network conditions, which can result in inefficient QoS management. Additionally, it provides limited contextual awareness, as it does not consider application-layer data or deeper packet inspection, leading to suboptimal handling of different types of traffic. Scalability is also an issue, as managing and updating static rules becomes increasingly cumbersome with growing and diverse network traffic. Finally, static systems are typically not capable of real-time adaptation to network anomalies or performance issues, delaying responses and potentially leading to degraded QoS during peak times or attacks.

IV.PROPOSED SYSTEM

The proposed system aims to overcome these limitations by leveraging the dynamic and adaptive capabilities of Software-Defined Networking (SDN). SDN introduces a centralized controller that enables more flexible and intelligent traffic management. This system utilizes machine learning algorithms to classify traffic dynamically based on various attributes, including application-level data and contextual information, rather than relying solely on static rules. This dynamic classification approach improves the accuracy and adaptability of traffic management, allowing for more precise QoS allocation. The system integrates deep packet inspection and application-layer insights to gain a comprehensive understanding of traffic requirements, ensuring that high-priority applications receive the necessary resources. Real-time adaptability is another key feature, as the SDN controller allows for continuous monitoring and immediate adjustment to changing network conditions and traffic patterns. This capability helps in promptly addressing performance variations and anomalies. Additionally, the SDN architecture supports scalable and flexible traffic management by centralizing control and automating policy updates, which simplifies rule management and enhances efficiency. Overall, this approach provides improved security and performance optimization, ensuring a higher quality of service by effectively managing and responding to evolving network demands.


EditPlus - [E:\Vithal\Mar23\EncryptedTrafficClassification\Dataset\ISCX_Data.csv]
File Edit View Search Document Project Tools Browser Window Help
Directory Cliptext
Darknet.csv
testtrain.py
CharacterClassificati
ISCX_Data.csv
Main.py
For Help, press F1
Type here to search

```
E+15,1.43645E+15,Non-Tor,Chat
,Non-Tor,Chat
5E+15,1.43645E+15,Non-Tor,Chat
14,1.43645E+15,1.43645E+15,Non-Tor,Chat
623019.03,1.43645E+15,1.43645E+15,Non-Tor,Chat
9500717.27,1.43645E+15,1.43645E+15,Non-Tor,Chat





nail

0,0,0,0,1.43681E+15,2508672.84,1.43681E+15,1.43681E+15,Non-Tor,Email

),0,0,1.43681E+15,2119216.818,1.43681E+15,1.43681E+15,Non-Tor,Email

0,0,1.43681E+15,2310563.251,1.43681E+15,1.43681E+15,Non-Tor,Email

1.43681E+15,2271534.216,1.43681E+15,1.43681E+15,Non-Tor,Email

43681E+15,Non-Tor,Email
0,0,1.44E+15,1645271.661,1.43681E+15,1.43681E+15,Non-Tor,Email

6,20,0,0,0,0,1.43681E+15,30549381.65,1.43682E+15,1.43681E+15,Non-Tor,Email
```

In above dataset each row contains one traffic data and in last column we have traffic classification labels as Email, Chat, Browsing and etc. so by using above dataset we will train Ensemble algorithm and calculate traffic classification accuracy.

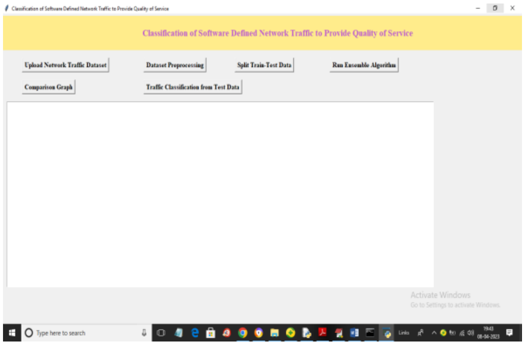
To implement this project we have designed following modules

1) Upload Network Traffic Dataset: using this module we will upload dataset to application and then find and plot graph of different traffic found in dataset
2) Dataset Preprocessing: using this module we will process dataset to remove missing values, normalization and shuffling all dataset values. Dataset contains some non-numeric values but machine learning accept only numeric values so by employing Label encoding class we will convert all non-numeric data to numeric values
3) Split Train-Test Data: using this module we will split dataset into train and test where application using 80% dataset for training and 20% for testing
4) Run Ensemble Algorithm: 80% processed data will be input to a group of ensemble algorithms such as SVM, Random Forest and J48 trained a model and

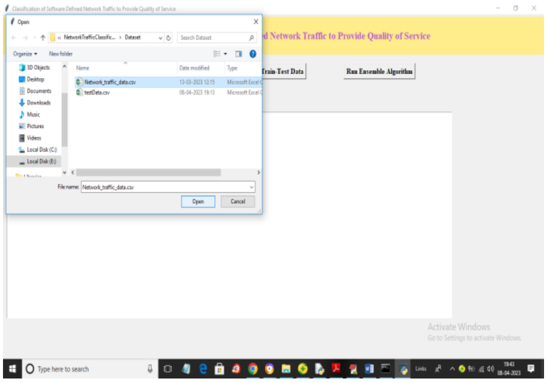
this model will be applied on 20% test data to calculate classification accuracy. In an ensemble group one classifier with the best accuracy will be selected.

5) Comparison Graph: using this module we will plot accuracy, precision, recall and FSCORE comparison graph
6) Traffic Classification from Test Data: using this module we will input TEST data and then Ensemble model will classify test data into possible traffic types.

To run project double click on run.bat file to get below screen

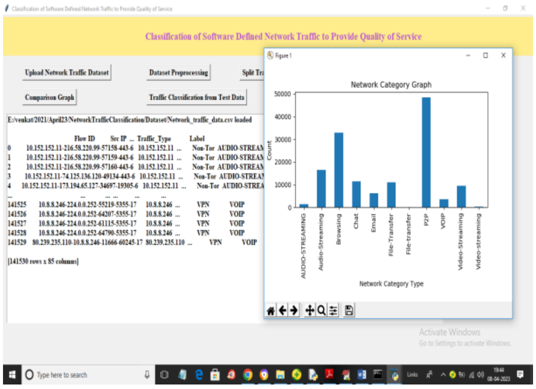

Classification of Software Defined Network Traffic to Provide Quality of Service
Classification of Software Defined Network Traffic to Provide Quality of Service
Upload Network Traffic Dataset
Dataset Preprocessing
Split Train-Test Data
Run Ensemble Algorithm
Comparison Graph
Traffic Classification from Test Data
Activate Windows
Go to Settings to activate Windows.
Type here to search

In above screen click on 'Upload Network Traffic Dataset' button to upload dataset and get below output



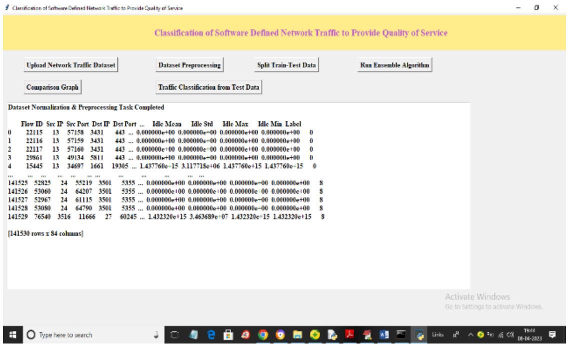
File name: Network_traffic_data.csv

In above screen selecting and uploading dataset and then click on 'Open' button to load dataset and get below output



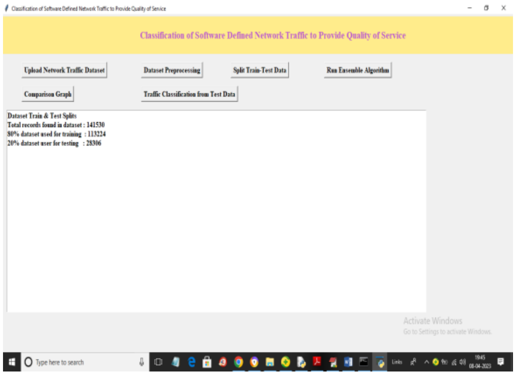
```
E:/venkat/2021/April23/NetworkTrafficClassification/Dataset/Network_traffic_data.csv loaded

                    Flow ID        Src IP ... Traffic_Type        Label
0      10.152.152.11-216.58.220.99-57158-443-6  10.152.152.11 ...   Non-Tor  AUDIO-STREAM
1      10.152.152.11-216.58.220.99-57159-443-6  10.152.152.11 ...   Non-Tor  AUDIO-STREAM
2      10.152.152.11-216.58.220.99-57160-443-6  10.152.152.11 ...   Non-Tor  AUDIO-STREAM
3      10.152.152.11-74.125.136.120-49134-443-6  10.152.152.11 ...   Non-Tor  AUDIO-STREAM
4     10.152.152.11-173.194.65.127-34697-19305-6  10.152.152.11 ...   Non-Tor  AUDIO-STREA
...                        ...           ... ...          ...           ...
141525    10.8.8.246-224.0.0.252-55219-5355-17    10.8.8.246 ...      VPN        VOIP
141526    10.8.8.246-224.0.0.252-64207-5355-17    10.8.8.246 ...      VPN        VOIP
141527    10.8.8.246-224.0.0.252-61115-5355-17    10.8.8.246 ...      VPN        VOIP
141528    10.8.8.246-224.0.0.252-64790-5355-17    10.8.8.246 ...      VPN        VOIP
141529   80.239.235.110-10.8.8.246-11666-60245-17  80.239.235.110 ...      VPN        VOIP

[141530 rows x 85 columns]
```

In above screen dataset loaded and we can see dataset contains both numeric and non-numeric data with '.' Symbols as encrypted data but ensemble algorithm only accept numeric data so we need to convert above data into numeric format by applying Label Encoding class. In above graph x-axis represents different traffic types exists in dataset and y-axis represents counts and now close above graph and then click on 'Dataset Preprocessing' button to process dataset and get below output


Classification of Software Defined Network Traffic to Provide Quality of Service
Classification of Software Defined Network Traffic to Provide Quality of Service
Upload Network Traffic Dataset
Dataset Preprocessing
Split Train-Test Data
Run Ensemble Algorithm
Comparison Graph
Traffic Classification from Test Data
Activate Windows
Go to Settings to activate Windows.
Type here to search

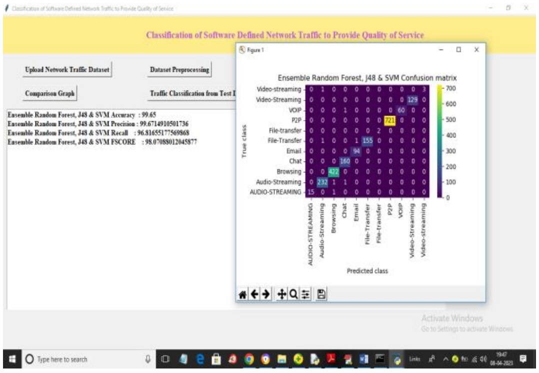
```
Dataset Normalization & Preprocessing Task Completed

        Flow ID  Src IP  Src Port  Dst IP  Dst Port  ...     Idle Mean      Idle Std      Idle Max      Idle Min  Label
0         22115      13     57158    3431       443  ...  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00      0
1         22116      13     57159    3431       443  ...  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00      0
2         22117      13     57160    3431       443  ...  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00      0
3         29861      13     49134    5811       443  ...  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00      0
4         15445      13     34697    1661     19305  ...  1.437760e+15  3.117718e+06  1.437760e+15  1.437760e+15      0
...         ...     ...       ...     ...       ...  ...           ...           ...           ...           ...    ...
141525    52825      24     55219    3501      5355  ...  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00      8
141526    53060      24     64207    3501      5355  ...  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00      8
141527    52967      24     61115    3501      5355  ...  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00      8
141528    53080      24     64790    3501      5355  ...  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00      8
141529    76540    3516     11666      27     60245  ...  1.432320e+15  3.463689e+07  1.432320e+15  1.432320e+15      8

[141530 rows x 84 columns]
```

In above screen entire dataset converted to numeric format and now click on 'Split Train-Test Data ' button to split dataset into train and test and then will get below output


Classification of Software Defined Network Traffic to Provide Quality of Service
Classification of Software Defined Network Traffic to Provide Quality of Service
Upload Network Traffic Dataset
Dataset Preprocessing
Split Train-Test Data
Run Ensemble Algorithm
Comparison Graph
Traffic Classification from Test Data
Activate Windows
Go to Settings to activate Windows.
Type here to search

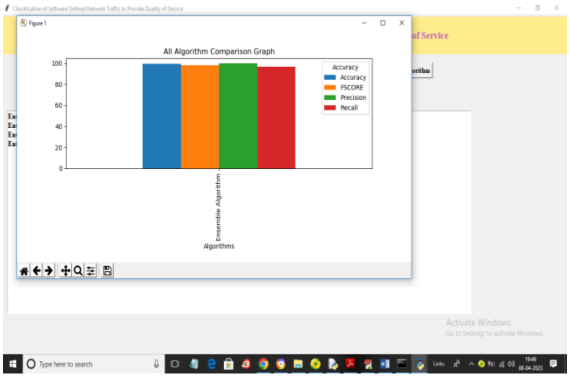
```
Dataset Train & Test Splits
Total records found in dataset : 141530
80% dataset used for training  : 113224
20% dataset user for testing   : 28306
```

In above screen we can see total records exists in dataset and we can see train and test split details and now click on 'Run Ensemble Algorithm' to ensemble algorithm and get below output

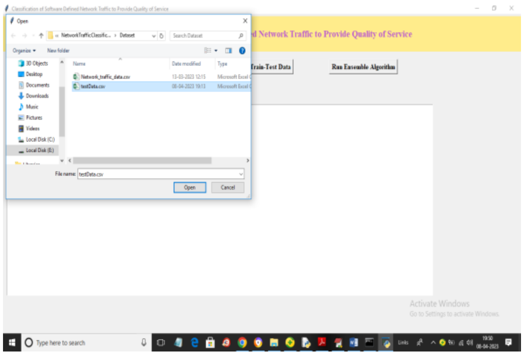

Classification of Software Defined Network Traffic to Provide Quality of Service
Upload Network Traffic Dataset
Dataset Preprocessing
Comparison Graph
Traffic Classification from Test I
Figure 1
Ensemble Random Forest, J48 & SVM Confusion matrix
Video-streaming 0 1 0 0 0 0 0 0 0 0 3
Video-Streaming 0 0 0 0 0 0 0 0 0 129 0
VOIP 0 0 0 1 0 0 0 0 60 0 0
P2P 0 0 0 0 0 0 0 721 0 0 0
File-transfer 0 0 0 0 0 0 2 0 0 0 0
File-Transfer 0 1 0 0 1 155 0 0 0 0 0
Email 0 0 0 0 94 0 0 0 0 0 0
Chat 0 0 0 160 0 0 0 0 0 0 0
Browsing 0 0 422 0 0 0 0 0 0 0 0
Audio-Streaming 0 232 1 1 0 0 0 0 0 0 0
AUDIO-STREAMING 15 0 1 0 0 0 0 0 0 0 0
True class
AUDIO-STREAMING
Audio-Streaming
Browsing
Chat
Email
File-Transfer
File-transfer
P2P
VOIP
Video-Streaming
Video-streaming
Predicted class
700
600
500
400
300
200
100
0
Activate Windows
Go to Settings to activate Windows
Type here to search

```
Ensemble Random Forest, J48 & SVM Accuracy  : 99.65
Ensemble Random Forest, J48 & SVM Precision : 99.6714910501736
Ensemble Random Forest, J48 & SVM Recall    : 96.81655177569868
Ensemble Random Forest, J48 & SVM FSCORE    : 98.07088012045877
```

In above screen with Ensemble algorithm we got 99% accuracy and we can see other metrics also and in confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels and all blue boxes contains INCORRECT prediction count which are few and all different colour boxes contains CORRECT prediction count. Now close above graph and then click on 'Comparison Graph' button to get below graph



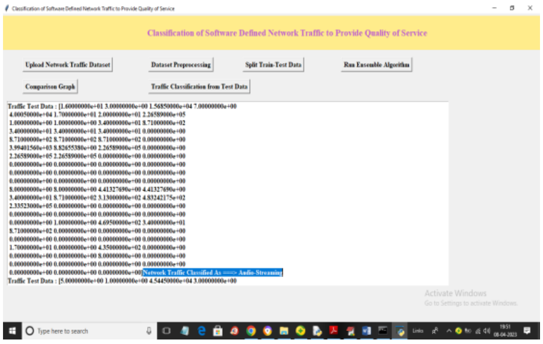
In above graph x-axis represents algorithm name and y-axis represents accuracy, precision and other metrics in different colour bars and all metrics got performance values greater than 98%. Now click on 'Traffic Classification from Test Data' button to upload test and then Ensemble model will classify traffic



Name	Date modified	Type
Network_traffic_data.csv	13-03-2023 12:15	Microsoft Excel C
testData.csv	08-04-2023 19:13	Microsoft Excel C

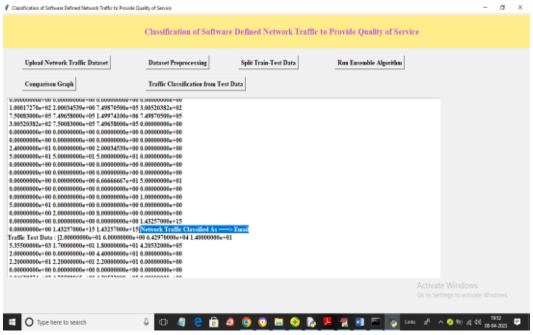
File name: testData.csv

In above screen selecting and uploading Test Data file and this file will not have traffic classification label and Ensemble model will analyse above file and classify network traffic type and get below output


Classification of Software Defined Network Traffic to Provide Quality of Service
Classification of Software Defined Network Traffic to Provide Quality of Service
Upload Network Traffic Dataset
Dataset Preprocessing
Split Train-Test Data
Run Ensemble Algorithm
Comparison Graph
Traffic Classification from Test Data
Activate Windows
Go to Settings to activate Windows.
Type here to search

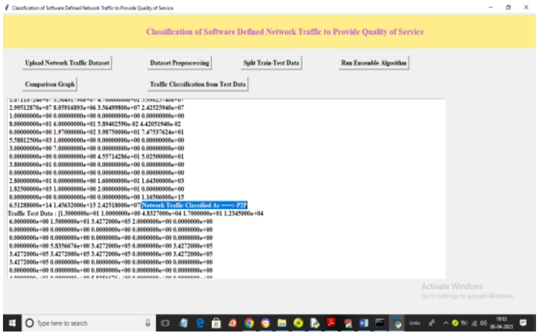
```
Traffic Test Data : [1.60000000e+01 3.00000000e+00 1.56850000e+04 7.00000000e+00
 4.00050000e+04 1.70000000e+01 2.00000000e+01 2.26589000e+05
 1.00000000e+00 1.00000000e+00 3.40000000e+01 8.71000000e+02
 3.40000000e+01 3.40000000e+01 3.40000000e+01 0.00000000e+00
 8.71000000e+02 8.71000000e+02 8.71000000e+02 0.00000000e+00
 3.99401560e+03 8.82655380e+00 2.26589000e+05 0.00000000e+00
 2.26589000e+05 2.26589000e+05 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 8.00000000e+00 8.00000000e+00 4.41327690e+00 4.41327690e+00
 3.40000000e+01 8.71000000e+02 3.13000000e+02 4.83242175e+02
 2.33523000e+05 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 1.00000000e+00 4.69500000e+02 3.40000000e+01
 8.71000000e+02 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 1.70000000e+01 0.00000000e+00 4.35000000e+02 0.00000000e+00
 0.00000000e+00 0.00000000e+00 8.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00]Network Traffic Classified As ===> Audio-Streaming
Traffic Test Data : [5.00000000e+00 1.00000000e+00 4.54450000e+04 3.00000000e+00
```

In above screen in square bracket we can see test data and then in blue colour text we can see classified traffic as ‘Audio Streaming’ and scroll down above screen to view all predicted output


Classification of Software Defined Network Traffic to Provide Quality of Service
Classification of Software Defined Network Traffic to Provide Quality of Service
Upload Network Traffic Dataset
Dataset Preprocessing
Split Train-Test Data
Run Ensemble Algorithm
Comparison Graph
Traffic Classification from Test Data
Activate Windows
Go to Settings to activate Windows.
Type here to search

```
 1.00017270e+02 2.00034539e+00 7.49870500e+05 3.00520382e+02
 7.50083000e+05 7.49658000e+05 1.49974100e+06 7.49870500e+05
 3.00520382e+02 7.50083000e+05 7.49658000e+05 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 2.40000000e+01 0.00000000e+00 2.00034539e+00 0.00000000e+00
 5.00000000e+01 5.00000000e+01 5.00000000e+01 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 6.66666667e+01 5.00000000e+01
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 1.00000000e+00
 5.00000000e+01 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 2.00000000e+00 8.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 1.43257000e+15
 0.00000000e+00 1.43257000e+15 1.43257000e+15]Network Traffic Classified As ===> Email
Traffic Test Data : [2.00000000e+01 6.00000000e+00 6.42970000e+04 1.40000000e+01
 5.35500000e+03 1.70000000e+01 1.80000000e+01 4.20532000e+05
 2.00000000e+00 0.00000000e+00 4.40000000e+01 0.00000000e+00
 2.20000000e+01 2.20000000e+01 2.20000000e+01 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
```

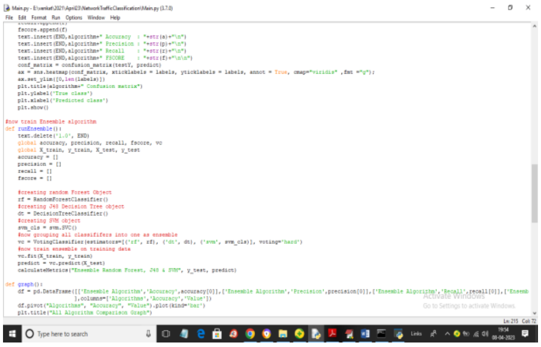
In above screen traffic classified as Email


Classification of Software Defined Network Traffic to Provide Quality of Service
Classification of Software Defined Network Traffic to Provide Quality of Service
Upload Network Traffic Dataset
Dataset Preprocessing
Split Train-Test Data
Run Ensemble Algorithm
Comparison Graph
Traffic Classification from Test Data
Activate Windows
Go to Settings to activate Windows.
Type here to search

```
 2.99512870e+07 8.05916893e+06 3.56499800e+07 2.42525940e+07
 1.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 8.00000000e+01 6.00000000e+01 5.89402590e-02 4.42051940e-02
 0.00000000e+00 1.97000000e+02 3.98750000e+01 7.47537624e+01
 5.58812500e+03 1.00000000e+00 0.00000000e+00 0.00000000e+00
 3.00000000e+00 7.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 4.55714286e+01 5.02500000e+01
 3.80000000e+01 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 2.80000000e+01 0.00000000e+00 1.60000000e+01 1.64300000e+03
 1.82500000e+03 1.00000000e+00 2.00000000e+01 0.00000000e+00
 0.00000000e+00 0.00000000e+00 0.00000000e+00 1.16506000e+15
 6.51288000e+14 1.45632000e+15 2.42518000e+07]Network Traffic Classified As ===> P2P
Traffic Test Data : [1.3000000e+01 1.0000000e+00 4.8327000e+04 1.7000000e+01 1.2345000e+04
 6.0000000e+00 1.5000000e+01 3.4272000e+05 2.0000000e+00 0.0000000e+00
 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
 0.0000000e+00 5.8356676e+00 3.4272000e+05 0.0000000e+00 3.4272000e+05
 3.4272000e+05 3.4272000e+05 3.4272000e+05 0.0000000e+00 3.4272000e+05
 3.4272000e+05 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
```

In above screen traffic classified as ‘P2P’. Similarly by following above screens you can run code

In below screen showing code for Ensemble Model by combining 3 algorithms



Main.py - E:\venkat\2021\April23\NetworkTrafficClassification\Main.py (3.7.0)
File Edit Format Run Options Window Help
Ln: 215 Col: 72
Type here to search

```
    fscore.append(f)
    text.insert(END,algorithm+" Accuracy  : "+str(a)+"\n")
    text.insert(END,algorithm+" Precision : "+str(p)+"\n")
    text.insert(END,algorithm+" Recall    : "+str(r)+"\n")
    text.insert(END,algorithm+" FSCORE    : "+str(f)+"\n\n")
    conf_matrix = confusion_matrix(testY, predict)
    ax = sns.heatmap(conf_matrix, xticklabels = labels, yticklabels = labels, annot = True, cmap="viridis" ,fmt ="g");
    ax.set_ylim([0,len(labels)])
    plt.title(algorithm+" Confusion matrix")
    plt.ylabel('True class')
    plt.xlabel('Predicted class')
    plt.show()

#now train Ensemble algorithm
def runEnsemble():
    text.delete('1.0', END)
    global accuracy, precision, recall, fscore, vc
    global X_train, y_train, X_test, y_test
    accuracy = []
    precision = []
    recall = []
    fscore = []

    #creating random Forest Object
    rf = RandomForestClassifier()
    #creating J48 Decision Tree object
    dt = DecisionTreeClassifier()
    #creating SVM object
    svm_cls = svm.SVC()
    #now grouping all classififers into one as ensemble
    vc = VotingClassifier(estimators=[('rf', rf), ('dt', dt), ('svm', svm_cls)], voting='hard')
    #now train ensemble on training data
    vc.fit(X_train, y_train)
    predict = vc.predict(X_test)
    calculateMetrics("Ensemble Random Forest, J48 & SVM", y_test, predict)

def graph():
    df = pd.DataFrame([['Ensemble Algorithm','Accuracy',accuracy[0]],['Ensemble Algorithm','Precision',precision[0]],['Ensemble Algorithm','Recall',recall[0]],['Ensemb
                      ],columns=['Algorithms','Accuracy','Value'])
    df.pivot("Algorithms", "Accuracy", "Value").plot(kind='bar')
    plt.title("All Algorithm Comparison Graph")
```

In above screen read red colour comments to know about ensemble algorithms grouping and classifying from one best algorithm

V. CONCLUSION

In this study, we conducted extensive research on network classification and examined numerous research papers on various classification models. Many papers presented feature extraction algorithms for classifying network traffic to its application. A significant amount of work is required to extract specific features of OTT platforms. We intend to restrict network traffic from OTT platforms. This lessens traffic and raises quality of service as a result.

VI.REFERENCES

[1] H. Shi, H. Li, D. Zhang, C. Cheng, W. Wu, Efficient and robust feature extraction and selection for traffic classification, Comput. Netw. 119 (2017) 1–16.

[2] M. Finsterbusch, C. Richter, E. Rocha, J.-A. Muller, K. Hanssgen, A survey of payload-based traffic classification approaches, IEEE Commun. Surv. Tut. 16 (2) (2014) 1135–1156 doi: https://doi.org/10.1016/j.media.2020.101813

[3] Meenaxi M Raikara, Meena S Mb, Mohammed Moin Mullac, Nagashree S Shetty, Meghana Karanandie,Data Traffic Classification in Software Defined Networks (SDN) using supervised-learning,ScienceDirect 171 (2020) 2750–2759

[4] Yoshinobu Yamada, Ryoichi Shinkuma, Takehiro Sato, and Eiji Oki Graduate School of Informatics, Kyoto University Yoshida-honmachi, Sakyo-ku, Kyoto, 606-8501, Japan, Feature-selection based data prioritization in mobile traffic prediction using machine learning , 978-1- 5386-4727-1/18/$31.00 ©2018 IEEE

[5] Amirhossein Moravejosharieh ,Kourosh Ahmadi ,Saghir Ahmad , A Fuzzy Logic Approach To Increase Quality of Service in Software Defined Networking, Communication Control and Networking (ICCC 2018)

[6] Gagangeet Singh Aujla,Rajat Chaudhary,Neeraj Kumar, An Ensembled Scheme for QoS-aware Traffic Flow Management in Software Defined Networks, 978-1-5386-3180-5/18/$31.00 ©2018 IEEE [7] Thomas Favale, Francesca Soroa , Martino Trevisana, Idilio Drago b, Marco Mellia, Campus traffic and e-Learning during COVID-19 pandemic.Computer Networks 176 (2020) 107290.

[8] Xiaoling Tao, Yang Peng, Feng Zhao, Changsong Yang, Baohua Qiang, Yufeng Wang, Zuobin Xiong, Gated recurrent unit-based parallel network traffic anomaly detection using subagging ensembles.Ad Hoc Networks 116 (2021) 102465.

[9] Xiaoshi Fana, Yanan Wang , Mengyu Zhang. Network traffic forecasting model based on long-term intuitionistic fuzzy time series Information Sciences 506 (2020) 131–147.

[10] REN-HUNG HWANG, (Senior Member, IEEE), MINCHUN PENG, CHIEN-WEI HUANG, PO-CHING LIN, AND VAN-LINH NGUYEN, (Member, IEEE). An Unsupervised Deep Learning Model for Early Network Traffic Anomaly Detection.Digital Object Identifier 10.1109/ACCESS.2020.2973023

[11] D. Jankowski and M. Amanowicz, "A study on flow features selection for malicious activities detection in software defined networks," 2018 International Conference on Military Communications and Information Systems (ICMCIS), 2018, pp. 1-9, doi: 10.1109/ICMCIS.2018.8398697.

[12] Ridwana, M. A., N. A. M. Radzib, and F. Abdullah. "Quality-of-Service Performance Comparison: Machine Learning Regression and Classification-Based Predictive Routing Algorithm." Turkish Journal of Computer and Mathematics Education (TURCOMAT) 12.14 (2021): 2808- 2817.