**Introduction:**

The precise classification of Internet traffic is the basis of many network management tasks including Quality of Service (QoS) control, intrusion detection and diagnostic monitoring. Traditional traffic classification approaches are based on examining the port numbers in transport layer header or investigating the signature information in the packet payloads. These approaches proved to be inefficient as they encounter many problems such as dynamic port numbers, data encryption and user privacy protection.

However, the classification performance of ML techniques is severely degraded due to the high dimensionality and redundancy of flow statistical features, the imbalance in the number of traffic flows and concept drift of Internet traffic.

The difficulty of the traffic identification is to find the features in the flow data. The process is very time‐consuming. Also, these approaches are invalid to unknown protocol. To solve these problems, we found a method that is based on neural network and deep learning.

The detection is based on a number of features associated with the communication flow, for Example, source and destination ports and bytes transmitted per packet. NTC is important, because much information about a current network flow can be learned and anticipated just by knowing its network service.

We found out that a recurrent neural network (RNN) combined with a convolutional neural network (CNN) provides best detection results. The natural domain for a CNN, which is image processing, has been extended to NTC in an easy and natural way. We show that the proposed method provides better detection results than alternative algorithms without requiring any feature engineering, which is usual when applying other models.[1]

We also found the results of other several architectures that integrate a CNN and an RNN, and also found the impact of the features chosen and the length of the network flows used for training.[1]

Literature Survey:

A Network Traffic Classification (NTC) infers the service/application (e.g. HTTP, SIP...) being used by a network flow. This information is important for network management and Quality of Service (QoS), as the service used has a direct relationship with QoS requirements and user contracts/expectations.

There are several approaches to NTC: port-based, payload based, and flow statistics-based.

Port-based methods make use of port information for service identification. These methods are not reliable as many services do not use well-known ports or even use the ports used by other applications.

Payload-based approaches the problem by Deep Packet Inspection (DPI) of the payload carried out by the communication flow. These methods look for well-known patterns inside the packets. They currently provide the best possible detection rates but with some associated costs and difficulties.

Flow statistics-based methods rely on information that can be obtained from packets header (e.g. bytes transmitted, packets inter arrival times, TCP window size…). They rely on packet header high-level information which makes them a better option to deal with non-available payloads

Or dynamic ports.

These methods usually rely on machine learning techniques to perform service prediction. Two machine learning alternatives are available in this case: Supervised and unsupervised methods.

Supervised methods learn an association between a set of features and the desired labeled output by training an algorithm with samples containing

Ground-truth labeled outputs.

In unsupervised methods, we do not have data with their associated ground-truth labeled outputs; therefore, they can only try to separate the samples in groups (clusters) according to some intrinsic similarities.

In [3] they propose a multi-layer perceptron (MLP) with zero or one hidden layer, but it is actually adopted as the internal architecture to apply a fully Bayesian analysis. The best one vs. rest accuracy, using 246 features, for 10 grouped labels is 99.8%, and a macro averaged accuracy of 99.3%.

An ensemble of MLP classifiers with error-correcting output codes is applied in[6] ,achieving an average overall accuracy (for 5 labels) of 93.8%.

Zhou *et al.* [5] apply an MLP with 3 hidden layers and different numbers of hidden neurons to the Moore dataset. They give an overall accuracy greater than 96%, for a grouping of labels in 10 classes, resulting in a final class distribution very unbalanced (a frequency of almost 90% for highest frequency class), no F1 score is provided.

A Parallel Neural Network Classifier Architecture is used in [6]. It is made up of parallel blocks of radial basis function neural networks. To train the network is employed a negative reinforcement learning algorithm. They classify 6 labels reporting a realistic overall accuracy of 95%, no F1 score is provided.

A recurrent neural network (RNN) combined with a convolutional neural network (CNN) provides best detection results out of all types of neural network and even comparably better than other ML algorithms.

A simple RNN model provides already very good results, but it is interesting to appreciate that these results improve when the RNN model is combined with a previous CNN model. A model based on a particular combination of CNN plus RNN gives the best detection results, being these results better than other published works with alternative techniques.

We found that for architecture with both CNN and RNN for classification gives an accuracy that is always higher than 98%, and many cases higher than 99% and an F1 score higher than 0.96. The macro averaged accuracy for these 15 labels is 99.59% which is the best value in literature.[1]

Challenges:

The classification of internet even using machine learning is challenging task for the following reasons:-

1. Feature Selection(FS) problem-
   1. FS techniques conduct the search for an optimal subset using different evaluation criteria, which may make the optimal subset be local optima.
   2. Most FS techniques have been developed for improving classification accuracy by removing the redundant features, but neglect the stability of optimal subset for variations in the traffic data.
   3. FS techniques cannot capture the complex dependency across all flow statistical features, which have a great impact on traffic classification. Thus, one of the key challenges is to provide the optimal and robust features for traffic classification.
2. Multi class imbalance problem-

It refers to the situation where ML algorithms suffer from low recall for the minority classes. The proposed solutions to this solutions are resampling and cost sensitive approaches which can hardly be addressed when feature space is high dimensional.

1. Drift of internet traffic problem-

Due to the evolution of network techniques and changes in user activities and management strategies, the Internet traffic and its underlying class distribution dynamically changes with time.

Motivation:

1. An NTC infers the service/application (e.g. HTTP, SIP : : :)being used by a network flow. This information is important for network management and Quality of Service (QoS), as the service used has a direct relationship with QoS requirements and user contracts/expectations.[2]
2. Network traffic identification is crucial for implementing effective management of network policy and resources in IoT networks, as the network needs to react differently depending on traffic profile information.[2]

Problem Definition:

***///Fill here***

Objectives:

1) Our task is to classify different network services that are currently used by a communication flow. (HTTP, UDP, SMTP…).

2) Understand how different architectures of CNN and RNN can be used for network traffic classification.

3) Find out the performance of different types of deep neural network (CNN and RNN) for the problem of classifying network traffic.

4) Find out methods for reducing dimensionality of the dataset.

Methodology:

We studied 3 architectures, which are

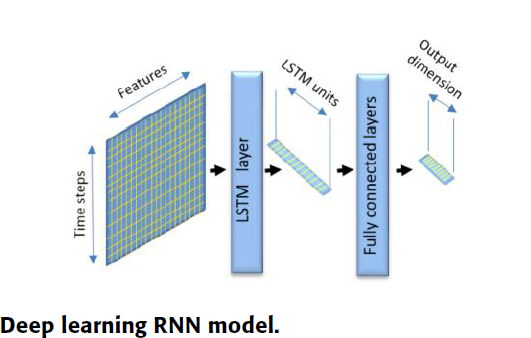
1) RNN

2) CNN

3) CNN + RNN

1) RNN:

The first model researched was a simple RNN. In particular, author used a variant of an RNN called LSTM, which is easier to train (it solves the vanishing gradient problem).An LSTM is trained with a matrix of values with two dimensions: the temporal dimension and a vector of features.[1]



2)CNN:

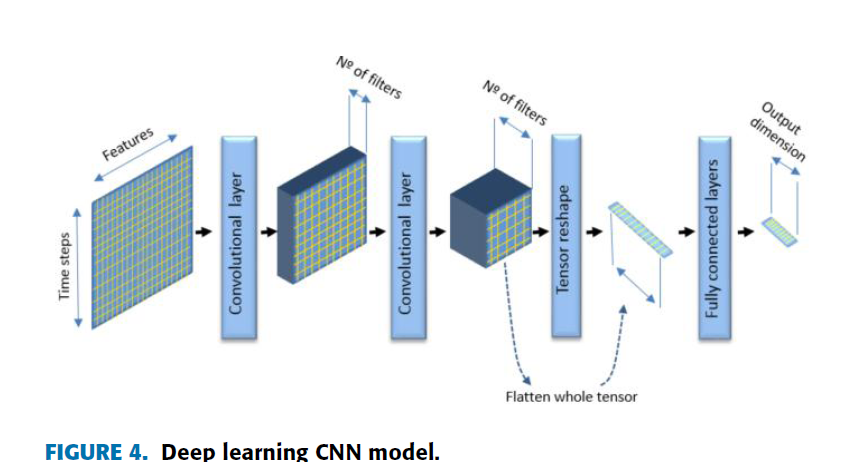
CNNs were initially applied to image processing, as a biologically inspired

model to perform image classification, where feature engineering was done automatically by the network thanks to the action of a kernel which extracts location invariant patterns from the image. Chaining several CNNs allows

extracting complex features automatically.

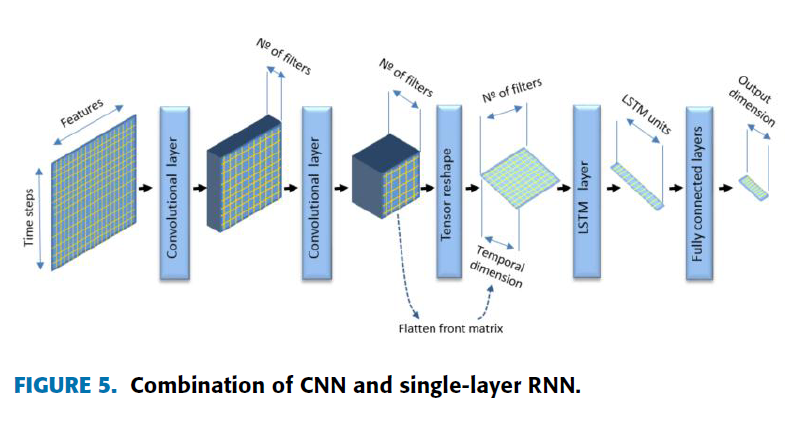
Each CNN layer generates a multidimensional array (tensor) where the dimensions of the image get reduced but, at the same time, a new dimension is generated, having this new dimension a size equal to the number of filters applied to the image. Consecutive CNN layers will further decrease the image dimensions and increase the new generated dimension size. To top off the model it is necessary to transform the

tensor to a vector that can be the input to the final fully connected layers. To accomplish this transformation a simple tensor flattening can be done.[1]



3)CNN + RNN:

The previous models can be combined in a single model as presented in Fig. In this combined model, the final tensor of several chained CNNs is reshaped into a matrix that can act as the input to an RNN (LSTM network). To reshape the tensor as a matrix we keep the dimension associated with the filter's action unchanged, performing a flattening on the other two dimensions, to finally reach a matrix shape. The values produced by the filters of the last CNN will be the equivalent of feature vectors, and the flattened vector produced by the reshaping operation will act as the time dimension needed by the LSTM layer.[1]



References:

[1] Network Traffic Classifier With Convolutional and Recurrent Neural Networks for Internet of Things

MANUEL LOPEZ-MARTIN1, (Senior Member, IEEE), BELEN CARRO1,

ANTONIO SANCHEZ-ESGUEVILLAS1, (Senior Member, IEEE),

AND JAIME LLORET2, (Senior Member, IEEE

[2] The Applications of Deep Learning on Traffic Identification

Zhanyi Wang

[3]T. Auld, A. W. Moore, and S. F. Gull, ``Bayesian neural networks for

Internet trafc classication,'' IEEE Trans. Neural Netw., vol. 18, no. 1,

pp. 223239, Jan. 2007.

[4] X. Xie, B. Yang, Y. Chen, L.Wang, and Z. Chen, ``Network trafc classi-

cation based on error-correcting output codes and NN ensemble,'' in Proc.

6th Int. Conf. Fuzzy Syst. Knowl. Discovery, Tianjin, China, Aug. 2009,

pp. 475479.

[5] W. Zhou, L. Dong, L. Bic, M. Zhou, and L. Chen, ``Internet trafc classi-

cation using feed-forward neural network,'' in Proc. Int. Conf. Comput.

Problem-Solving (ICCP), Chengdu, China, Oct. 2011, pp. 641646.

[6] B. Mathewos, M. Carvalho, and F. Ham, ``Network trafc classication

using a parallel neural network classier architecture,'' in Proc. 7th Annu.

Workshop Cyber Secur. Inf. Intell. Res. (CSIIRW), New York, NY, USA,

2011, p. 33.