ANN and DNN-based Models for DDoS Detection via Network Traffic Forecasting

كشف هجمات رفض الخدمة الموزعة عبر التنبؤ بتدفق بيانات الشبكة باستخدام نماذج الشبكات العصبونية الاصطناعية والتعلم العميق

By:

Amjad Ibrahim Gendary

Supervisor:

Dr. Abdelrahman Abuarqoub

A Thesis Submitted in Partial Fulfillment of the Requirements for the Master Degree in Computer Science

Department of Computer Science

Faculty of Information Technology

Middle East University

May, 2019

Authorization

I, **Amjad Ibrahim Gendary**, hereby authorize Middle East University to supply copies of my thesis to Libraries, organizations or individuals when required.

Name: Amjad Ibrahim Gendary.

Date: 03 / 06 / 2019

Signature: 

Thesis Committee Decision

This thesis titled "**ANN and DNN-based Models for DDoS Detection via Network Traffic Forecasting**" **was successfully defended and approved on 03 / 06 / 2019**

Thesis Committee Members	**signature**
(Supervisor) **Dr. Abdelrahman Abuarqoub** **Middle East University**	
(Head of the Committee and Internal Examiner) **Dr. Bassam Al-shargabi** **Middle East University**	
(External Examiner) **Prof. Dr. Sadeq Al-Hamouz** **Islamic Science University**	17-6-2019

Acknowledgment

(الحمد لله على نعمه السابغات المتواليات وعلى رسوله الامين افضل الصلوات)

Special thanks to my mother and father who supported me and encouraged me in all stages of my life to complete it to the fullest.

Thanks to the supervisor, **Dr. Abdelrahman Abuarqoub** was keen to help me to present this scientific work as it should be.

Thanks to my College Professors who gave me the scientific guidance.

Thanks to my colleagues who supported me with their friendliness and their pure love.

The Researcher

Amjad Gendary

بسم الله الرحمن الرحيم

"وقل رب زدني علما"

Dedication

This thesis is dedicated:

To my dear homeland Iraq and my second homeland Jordan.

To those who are unable to describe them **my Father** and **my Mother**.

To my sister **Dr. Shirin**.

To my brother **Dr. Ashraf**.

And to those who started with me the footsteps of life my dear brother **Majdy.**

Table of Contents

List of Figures

List of Tables

Table of Abbreviations

Abbreviations	Meaning
ANN	Artificial Neural Network
AR	AutoRegressive
ARIMA	AutoRegressive Integrated Moving Average
ARMA	AutoRegressive Moving Average
DDoS	Distributed Denial of Service
DNN	Deep learning Neural Network
FARIMA	Fractal AutoRegressive Moving Average
SARIMA	Seasonal AutoRegressive Integrated Moving Average
UDP	User Datagram Protocol

ANN and DNN-based Models for DDoS Detection via Network Traffic Forecasting

By: Amjad Ibrahim Gendary

Supervisor: Dr. Abdelrahman Abuarqoub

Abstract

Cyber-attacks such as DDoS critically affect the available network bandwidth which means that by analyzing the coming network traffic, DDoS attacks can be detected. Detecting DDoS attacks is never a simple task. It typically relies on classifying the coming network requests and distinguishes between the traffic coming from attacking sources and the normal legitimate network traffic. One of the most efficient approaches for DDoS detection is via bandwidth forecasting as it provides a clear understanding of the legitimate traffic and helps determine the infected DDoS attack from the legitimate user traffic coming to the servers. Thus, this work proposes a DDoS detection method via forecasting network bandwidth using an Artificial Neural Networks (ANN) and Deep learning Neural Networks (DNN).

The ANN models are Single-Layer Feedforward architecture ANN model, and Single-Layer Elman architecture ANN model. The DNN model is a Multi-Layer Feedforward Neural Network These models are built in MATLAB and are trained using a set of time-series network traffic data set to first predict future traffic demands. Secondly, further analysis is applied to the forecasted bandwidth outcomes to detect DDoS attacks.

All three models forecasting performance is critically analyzed and compared to each other in a number of network bandwidth training and predicting experiments of which all have achieved extremely high forecasting results of accuracy rates above 97.8%.

Such results of forecasting performance of the proposed ANN models allow for further analysis of the forecasted bandwidth for DDoS detection by comparing the outcomes of the ANN forecasting model to the actual coming traffic. The results of this work have shown that the DNN model and the single-layer feedforward NN model have the highest accuracy rates in comparison to the previously proposed recurrent NN and the competitive NN models.

Keyword: Artificial neural network (ANN), Deep learning neural network (DNN), Forecasting, DDoS Attack.

كشف هجمات رفض الخدمة الموزعة عبر التنبؤ بتدفق بيانات الشبكة باستخدام نماذج الشبكات العصبونية الاصطناعية والتعلم العميق

إعداد: أمجاد إبراهيم جنداري

إشراف: الدكتور عبد الرحمن ابو عرقوب

الملخص

تؤثر الهجمات الالكترونية مثل رفض الخدمة الموزعة بشكل كبير على عرض النطاق الترددي للشبكة المتوفرة، مما يعني أنه من خلال تحليل حركة مرور الشبكة القادمة يمكن اكتشاف هجمات رفض الخدمة الموزعة، لا يعد اكتشاف هجمات رفض الخدمة الموزعة مهمة بسيطة. عادةً ما يعتمد على تصنيف طلبات الشبكة القادمة ويميز بين حركة المرور القادمة من المصادر المهاجمة وحركة مرور الشبكة الشرعية العادية. أحدى الطرق الاكثر فاعلية لاكتشاف هجمات رفض الخدمة هي من خلال التنبؤ بالنطاق الترددي حيث انه يوفر فهما واضحاً لحركة المرور المشروعة ويساعد في تحديد المصاب بهجوم رفض الخدمة الموزعة من حركة مرور المستخدمين الشرعية القادمة الى الخوادم. وبالتالي، يقترح هذا العمل طريقة للكشف عن هجمات رفض الخدمة الموزعة من خلال التنبؤ بالنطاق الترددي للشبكة باستخدام الشبكات العصبية الاصطناعية (ANN) والشبكات العصبية للتعلم العميق (DNN).

تتكون الـ ANN من نموذجين اولاً بنية التغذية الامامية احادي الطبقة، ثانياً بنية التغذية العكسية احادي الطبقة، بينما الـ DNN فيتكون من بنية التغذية الامامية متعددة الطبقات. تم تصميم هذه النماذج في برنامج ماتلاب ويتم تدريبها باستخدام مجموعة من بيانات حركة مرور الشبكة في سلسلة زمنية للتنبؤ اولاً بمتطلبات حركة المرور المستقبلية، ثانياً يتم تطبيق مزيداً من التحليل على نتائج عرض النطاق الترددي المتوقعة للكشف عن هجمات رفض الخدمة.

يتم تحليل اداء النماذج الثلاثة للتنبؤ بشكل نقدي ومقارنتها مع بعضها البعض في عدد من التدريبات على االنطاق الترددي للشبكة والتجارب التي توقعت جميعها نتائج تنبؤ عالية للغاية بمعدلات دقة اعلى من 97.8 في المائة. ويتم تحليل نتائج الاداء التنبؤي للنماذج المقترحة تحليلا اضافيا لاكتشاف هجمات رفض الخدمة الموزعة من خلال مقارنة نتائج التنبؤ للنماذج المقترحة بالحركة الغعلية القادمة.ولقد اظهرت نتائج هذا العمل ان نموذج التغذية الامامية متعددة الطبقات (DNN) ونموذج التغذية الامامية احادية الطبقة (ANN) يتمتعان باعلى معدلات للدقة مقارنة مع نموذج التغذية العكسية احادية الطبقة.

الكلمات المفتاحية: الشبكة العصبية الاصطناعية (ANN)، الشبكة العصبية للتعلم العميق (DNN)، التنبؤ، هجوم DDoS.

Chapter One

Introduction

The presented work of this thesis highlights the great importance of which accurate bandwidth forecasting plays in detecting DDoS attacks. It also demonstrates the ability of neural network models to effectively predict the future bandwidth demands of a network provided detailed historical bandwidth data.

In this chapter an overview of the main motivations, objectives, the scope and the main limitations of the proposed approach are presented. Additionally, a brief introduction into DDoS detection mechanism via network traffic forecasting is also covered in this chapter.

1.1 Research Context

Prosperity in the connected world is heavily reliant on online connectivity being available for normal tasks. The situation results from an abundance of internet applications supplying the basic services for our daily routines. The huge growth of network-enabled applications and services has drawn much attention over the last few decades. This situation arises from the huge demands made on internet resources, wherein the attainment of stable networking performance is critical to handling ever-increasing requirements. Such capabilities encourage people of interest to investigate and develop smarter approaches for meeting the rising network usage (Vinayaka Jyothi 2016).

1.2 Background

Due to the rapidly increasing number of online-based applications and services, there has been a greater need for better more robust online networks infrastructures. This calls for an improved network performance which includes an increased immune to outside cyber-attacks and better detection capabilities of abnormal network behaviors. One of the most common cyber-attacks is the Distributed Denial of Service (DDoS) cyber-attacks. DDoS harms some of the most important metrics of an online network (Adeilson Marques da Silva Cardoso 2018).

Network bandwidth is typically defined as the rate at which the data bits passes/travel through the connections and the nodes of a network and is usually measured in kbits/second (Vinayaka Jyothi 2016). Bandwidth prediction plays a key role in suppling users with steady and consistent bandwidth. Its practical advantages stem from the fact that advance knowledge of anticipated demand will enable adaptive bandwidth allocation to specific network nodes. Additionally, further analysis of the bandwidth forecasting allows for bandwidth-based attacks prediction including volume DDoS type of attacks. This is because bandwidth forecasting enables the network administrator to distinguish between a legitimate user traffic and infected abnormal traffics (Adeilson Marques da Silva Cardoso 2018).

1.3 Problem Statement

Network availability is heavily threatened when cyber-attacks such as DDoS take place on the network. In order to enable the network to have effective immune to such threats, the network must be able to first detect an attack when it occurs in order to block it out of its coming traffic.

Studies such as (Adeilson Marques da Silva Cardoso 2018) has shown that one of the most effective methods to detect DDoS attack is via bandwidth forecasting as it provides the network with a clear vision of future legit traffic demands and allows it to distinguish critical attacks that impacts the network performance.

Due to the importance of bandwidth forecasting to enable the network to detect such attacks, this work adopts several ANN and DNN models for precise traffic forecasting and then utilize the outcomes of the forecasting process in further analysis to detect DDoS attacks as will be detailed in Chapter 3.

1.4 Aim and Objectives

This work aims to provide a precise bandwidth prediction technique based on ANN and DNN approach to be utilized in DDoS attacks detection. The main objectives of the proposed work are:

1. Implement a high accuracy bandwidth forecasting model using an Artificial Neural Network and Deep Learning models.
2. Determine the most suitable feedforward and recurrent NN models to enhance bandwidth forecasting and DDoS detection performance.
3. Determine the most suitable training data size for the mentioned models and the highest prediction period while maintaining high bandwidth forecasting and DDoS performance
4. Compare single-layer feedforward ANN architecture & feedback (Elman) architecture and multi-layer Feedforward in terms of their bandwidth forecasting performance and DDoS detection.

1.5 Motivation

The ability to predict future traffic demands is extremely helpful to a network as it allows for a more controlled services environment with less down times and more effective immune against outside cyber-attacks enabling better security measures.

Although a number of previous works have been proposed to achieve decent bandwidth forecasting, most suffered from issues such as having very minimal generalizing capabilities (providing decent forecasting results only for specific data domains). Others suffered from the need for huge amounts of training data and relatively massive computational power in order to achieve precise results. These shortcomings in the previous works along with the need for highly accurate traffic forecasting have been the motivation for this work to propose more suitable solutions that provide better network forecasting capabilities with the least amount of training data possible that can still achieve high forecasting accuracies to be further utilized in precise DDoS cyber-attacks detection.

1.6 Significance of Research

The significance of this work is summarized in the following contributions:

1) Building a single-layer Feedforward, a multi-layer feedforward and a single-layer Recurrent NN architectures to compare the performance of each one of them when other works mainly focus on optimizing one architecture.
2) Implementing a 3rd model of Multi-layer (deep learning) NN architecture for further analysis and comparisons to the single-layer models for both bandwidth forecasting and DDoS detection performance.
3) Enabling flexible customizations of the proposed ANN and DNN models in terms of the number of layers, number of hidden neurons, activation functions, training data and testing data sizes which enables detailed comparisons among the suggested models and enables better understanding of the pros and cons of each model.
4) DDoS detection by further analysis of the forecasting process outcomes which provides more accurate detection of the legitimate and harmful traffic specifically for the analyzed network.

1.7 Research Questions

1- How accurate is a supervised ANN and DNN feedforward and feedback models are in predicting a certain network future bandwidth demands?

2- Will the usage a feedback Artificial Neural Network model instead of a feedforward model improve the same network bandwidth forecasting and DDoS performance?

3- Does a Deep learning (multi-layer) NN model provide better forecasting and DDoS detection results compared to single layer ANN models?

4- Does changing the training data set size have a significant impact on the bandwidth forecasting and/or DDoS detection performance?

5- What are the best parameters (activation functions, number of neurons, training rule…etc) for a feedforward & feedback ANN and DNN models for a highly accurate bandwidth forecasting and DDoS detection?

6- How much of an impact does precise bandwidth forecasting of a network can have on detecting DDoS attacks on the network?

1.8 Scope and limitations

The work of this thesis proposes several Neural Network models to achieve high performance of traffic forecasting of a local network. The proposed models are Single-layer Feedforward NN, Single-layer feedback (Elman) NN and Multi-layer NN (deep learning). In addition, the forecasting outcomes are utilized in the same mentioned models to detect DDoS attacks which be extremely harmful on any network services performance.

The lack of annual traffic data allows this work to only provide short to mid-term forecasting which doesn't allow for further investigation for how precise the proposed models can be for long-term forecasting. In addition, some of the random aspects of the ANN and DNN models such as random initial weights and random neuron bias values can negatively impact the forecasting and DDoS detection of the proposed models although minimally. The models were built on a PC with average specifications of Intel core i5 2.4 Ghz CPU, and 4 Gbs of RAM using MATLAB R2018b. The hardware specifications of the PC didn't play a major rule in the model performance. However, it can impact the training time of each implemented model such that the faster the CPU threads and the bigger than RAM are the less training time required.

1.9 Thesis Organization

The rest of this thesis is organized as follows:

- Chapter 2: provides a detailed description of the proposed models along with key neural network parameters and learning methods. Selected previous works for both network forecasting and DDoS detection are also discussed in this chapter
- Chapter 3: provides a detailed description of the proposed approach along with a description of the used data set.
- Chapter 4: provides a detailed description of the excused experiments using the proposed approach and a comparison of the proposed approach results vs the closest previous work
- Chapter 5: provides a conclusion of the study along with suggested future works to be investigated as a continuation of the proposed approach.

Chapter Two

Literature Review and Related Work

This chapter presents a summary of ANN as well as descriptions of key ANN parameters and their effects on network performance. Sections 2.1 and 2.2 covers the main concepts of neural networks. Section 2.3 describes the main architectures of single-layer NNs. Section 2.4 details the learning methods of NNs. Finally, Section 2.5 summarizes previous works.

2.1 Definition of Artificial Neural Networks

As demonstrated in Figure 2.1, Artificial Neural Networks (ANN) are mathematical models that attempt to simulate the structures and functionalities of biological neural networks. The key building block of an ANN is the artificial neuron, which is structurally expressed in terms of a basic mathematical function which is typically referred to as ANN Model (Aykut Tahtirvanci 2018).

The scheme comprises 3 basic rulesets: activation, multiplication, and summation rules. At the entry of each neuron, each input value is weighted, with all input values multiplied by individual weights. At its mid part, the summation function sums up all input weights and biases. At its exit, the sum of all prior bias and weighted input values is passed through the activation or transfer function (K. C. Sriharipriya 2017).

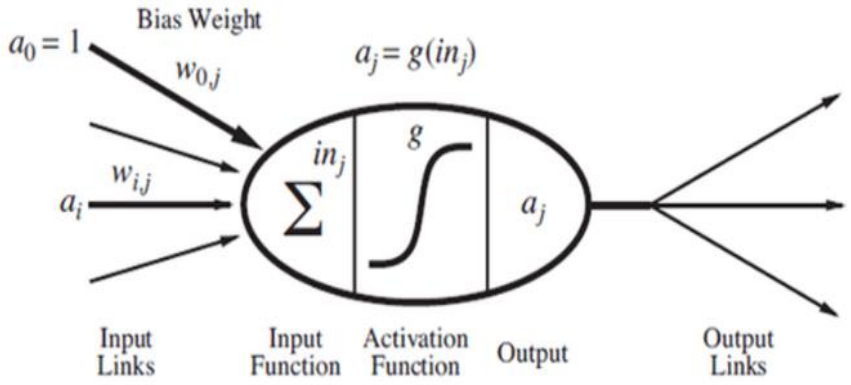

Figure 2.1: Basic Structure of an artificial neuron (K. C. Sriharipriya 2017).

Even though the operating principles and basic rulesets regarding artificial neurons do not seem so special, the full potentials and computational powers of these systems become manifest once interconnected with ANNs. The advantages arise from the fact that complexities can emerge from a mere few simple rules. Figure 2.1 below shows how neurons in ANN are interconnected to form a complex structure (Aykut Tahtirvanci 2018).

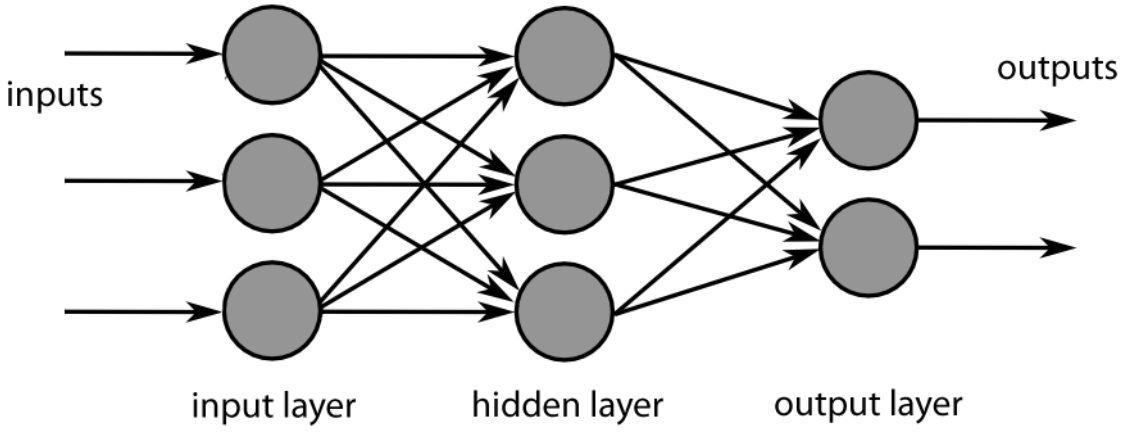

Figure 2.2: Typical ANN structure (Aykut Tahtirvanci 2018).

To fully exploit the mathematical complexity attained via the interconnection of discrete artificial neurons, and not just by rendering such systems with complicated and unmanageable means, the neurons are normally not interconnected in some random manner. Previously, researchers advanced several standardized ANN topographies. Such predefined structures can assist with faster, easier, and more effective problem-solving. Various ANN topographies are differently suited for resolving various types of problems. After defining the nature of the problem to be resolved, one must choose the ANN topology that will be used, then fine-tune its features as well as its parameter set (Shubhankar Kapoor 2016)

A fine-tuned ANN topology does not lead to the immediate operation of an artificial neural network, for it is just a prerequisite condition. Before the ANN can be used, it must be trained to solve the type of a given problem. Where a biological neural network may acquire behavioral responses according to the inputs received from the natural environment, an artificial neural network must perform similarly (Swagat Ranjit 2018).

There are two primary learning methods: unsupervised and supervised. Such methods are selected in the way ANN topography is chosen, in accordance with the problems we are attempting to solve. Even though training paradigms vary in their operating principles, all feature one element in common. For with the use of learning rules and training data, all such systems attempt to attain proper output responses based on input signaling (Swagat Ranjit 2018).

Following the selection of an ANN topology, with the fine-tuning of its features as well as the learning of its proper behaviors, the neural network can then be used to solve

given problems. Artificial neural networks are in much use, for these can be found working in various fields, including genetics, chemistry, astronomy, gaming, spaceflight, banking, automotive industry, fraud detection, radar systems, process control, and so on. ANNs are regularly applied to problem-solving in terms of functional approximation, regression analysis, time-series prediction, classification, pattern recognition, decision-making, clustering, filtering, data processing, among others (Gerd Bramerdorfer 2014).

2.2 Neurons of Artificial Neural Networks

Artificial neurons comprise the essential building blocks of all artificial neural networks. Their key functionalities and design derive from understandings of the organic neurons that comprise the essential building blocks of biological neural network systems, which include brains, spinal cords, and peripheral ganglia. The similarities in design and functions are shown in Figure 2.3, wherein the left part of the figure displays a biological neuron along with the soma, dendrites, and axons, whereas the right part of the figure displays an artificial neuron along with its various inputs and weights, transfer functions, biases, and outputs (Leonid Kupershtein 2016).

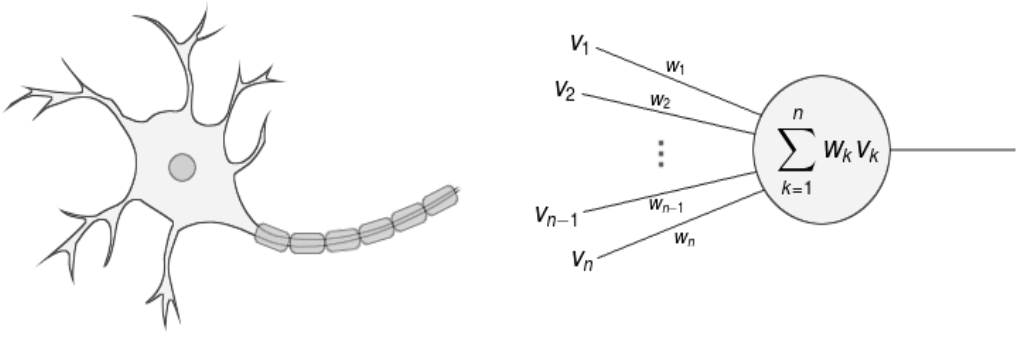

Figure 2.3: Artificial neuron compared to real neuron (Leonid Kupershtein 2016).

According to (Gerd Bramerdorfer 2014), in biological neurons, information passes through dendrites into neurons, while their soma process information for onward transmission via axons. With an artificial neuron, all information passes into the artificial neuron's body through weighted inputs. The artificial neuron sums all bias and weighted input values, then processes this sum through the transfer function. Ultimately, the neuron transmits all processed information through outputs. The beneficial simplicity of the artificial neuron scheme is clearly shown in its mathematical expression, as follows:

$$y(k) = F\left(\sum_{i=0}^{m} w_i(k) \cdot x_i(k) + b\right) \quad (1)$$

Where xi the inputted values to the neuron, wi is the corresponding weight to each inputted value to the neuro, b is the neuron bias value and F is the activation function of the neuron.

As with the artificial neuron model, the key unknown variable in the scheme is the transfer function. These define the properties of the neurons and may present as any mathematical function. The particular function is chosen according to the problem which the ANN is meant to resolve. In the majority of cases, it is selected from among the function set comprising the Step, Linear, and Non-linear (Sigmoid) functions (Arif Selçuk Öğrencı 2018).

The Step function denotes a binary function, with outputs of only two probable values, 1 and 0. This implies that if an input value meets the specified threshold, then the output

results in a particular value, whereas if the value does not meet the threshold then another output value results (Swagat Ranjit 2018).

Where this variant of transfer functions is applied to an artificial neuron, the latter is referred to as a perceptron. Perceptrons are used to resolve classification problems and thus may be found most frequently in the last ANN layers. With the linear transfer function, the artificial neurons are performing basic linear transformation across the sum of all bias and weighted input values. Such neuronic configurations contrast with the perceptrons most frequently used in ANN input layers. With non-linear functions, the sigmoid function is the most frequently used. This function features easily-calculated derivatives, an advantage that can be critical in the calculation of ANN weight updates (Swagat Ranjit 2018).

2.3 Artificial Neural Networks Architectures

Artificial neural networks have several architectures based on the way the neurons are interconnected and distributed in the network (K. C. Sriharipriya 2017). The main architectures of a neural network are:

1) **Feedforward-ANN**

An artificial neural network featuring feedforward topology is termed a feedforward artificial neural network. It intrinsically involves only a single condition, that information has to flow from inputs to outputs in a single direction only and without any back-loop. No constraints exist on the numbers of layers, types of transfer functions applied to discrete neurons, or the numbers of connections made among discrete

neurons. The simplest feedforward ANN comprises only one perceptron, which can only learn to resolve linear discrete problems. A simple multilayer feedforward ANN can be constructed for purpose of analytical definition. Figure 2.4 demonstrates the data flow in a Feedforward-ANN architecture along with its neuron's interconnections (LeZhangP.N.Suganthan, 2016).

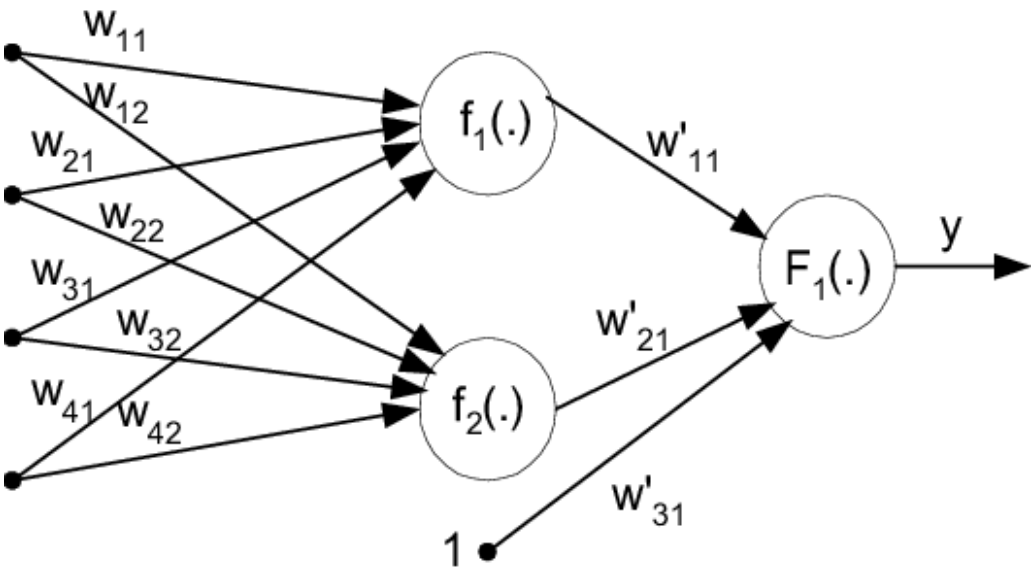

Figure 2.4: Feedforward-ANN data flow (LeZhangP.N.Suganthan, 2016).

As shown in Figure 2.4, the simplest feedforward ANNs can result in comparatively extended mathematical expressions, wherein manual parameter optimization for ANN problem-solving would be impractical. Even though such analytical expressions apply to all complex ANNs, in practice only computing hardware and specialized software are used to mathematically construct, describe, and optimize all types of ANNs (LeZhangP.N.Suganthan, 2016).

2) Feedback-ANN

Artificial neural networks that feature recurrent topologies are termed Feedback or Recurrent artificial neural networks. These resemble feedforward neural networks with back-loop restrictions. In such schemes, the information is no longer sent in only a single direction, but backwards as well. The scheme generates internal states within networks that allow them to manifest dynamic temporal behaviors. A recurrent ANN can use its internal memory units for processing any series of inputs. Figure 2.5 displays a small Recurrent ANN and the complex artificial nature of its neuronic interconnections (Grégoire Mesnil, 2015).

The simplest topology for a recurrent ANN is the fully recurrent artificial network model, wherein all basic building blocks (artificial neurons) are directly connected to all other such units in every direction. Recurrent ANNs including Elman and Jordan, bi-directional (Grégoire Mesnil, 2015), are the most commonly-known recurrent ANNs (Weibo Liu, 2017).

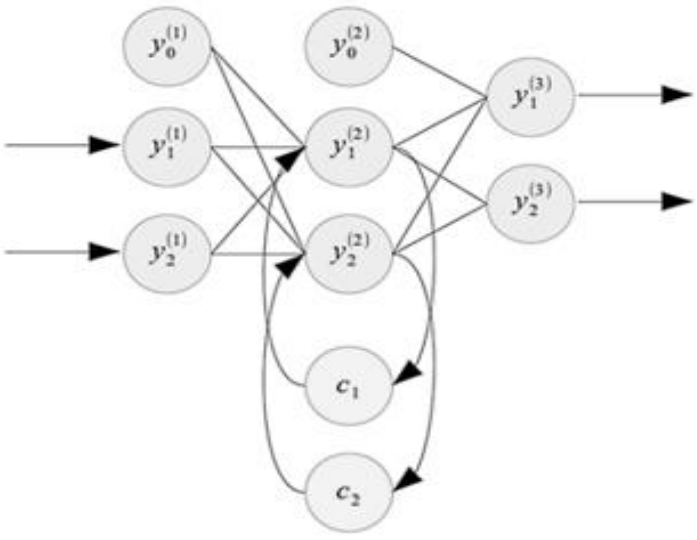

Figure 2.5: Feedback-ANN data flow (Grégoire Mesnil, 2015).

- **Elman Recurrent ANN**

As shown in figure 2.6 (left), Elman networks, also termed Simple Recurrent Networks, comprise special cases of recurrent ANNs. These differ from classic 2-layer networks as the scheme's first layer features a recurrent connection. The type features a basic 3-layer artificial neural network with a back-loop from its hidden to input layers, via a context unit. It features memory that enables it to detect as well as produce time-variant patterns (Grégoire Mesnil, 2015).

Elman artificial neural networks typically have sigmoid artificial neurons in their hidden layers and linear artificial neurons in their output layers. Such a combination of transfer functions for artificial neurons can approximate all target functionalities with arbitrary precision, given sufficient neurons in the hidden layer. With its ability to retain information, an Elman ANN can generate temporal and also spatial patterns in response (Grégoire Mesnil, 2015).

- **Jordan Recurrent ANN**

As demonstrated in figure 2.6 (right). Jordan ANN network resembles an Elman network, with the only difference being that its context units receive their feed from output layers in place of hidden layers (Grégoire Mesnil, 2015).

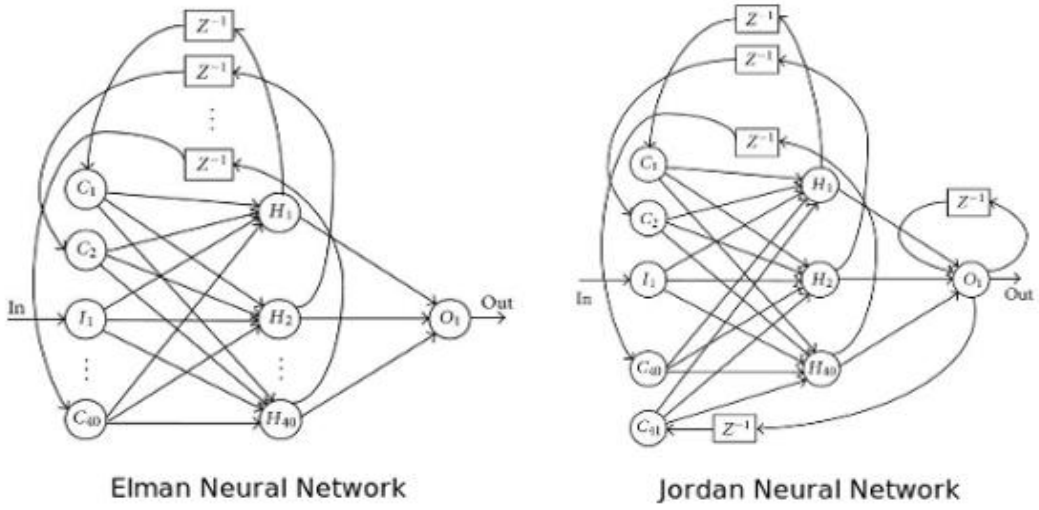

Figure 2.6: Elman (left) vs. Jordan (right) ANNs (Grégoire Mesnil, 2015).

2.4 Artificial Neural Networks Learning

There are two primary learning models: unsupervised, supervised learning paradigms. These are typically used by all types of ANN architectures, with every learning model able to run numerous training algorithms (R. Sathya, 2013).

2.4.1 ANN Supervised Learning

Supervised learning comprises machine learning methods that set parameters for artificial neural networks based on training data. A learning ANN is tasked to establish its parameter set for all valid input values following its examination of output values. ANN training data comprises a series of paired input and required output values, that are conventionally characterized as data vectors. ANN Supervised learning is similarly referred to as classification. For any given problem, the choice of an appropriate classifier from a range that includes Support Vector Machine, Multilayer perceptron, Gaussian mixture scheme, Gaussian, k-nearest neighbor algorithm, naive Bayes, radial

basis function classifier, decision tree, and so on, can involve more art than science (K. C. Sriharipriya 2017).

Solving any problem in supervised learning entails a variety of stages. For the initial stage, we must establish which types of training examples apply. At the second stage, a training dataset must be assembled that can satisfactorily describe the given problem. At the third stage, the assembled training dataset must be described in a form that is understandable to the selected ANN type. At the fourth stage, learning is performed, after which the performances of trained ANNs are evaluated using the testing dataset. The testing method uses data that has not yet been introduced into the ANN during training (Aykut Tahtirvanci 2018).

2.4.2 ANN un-supervised Learning

Unsupervised learning paradigms comprise machine learning methods that set ANN parameters according to given data as well as a minimizing cost function. The latter could be any function that is determined via expression of the task requirements. The unsupervised learning paradigm is mainly employed in applications that involve the domain of various estimation problems, including filtering, compression, blind- source separation and clustering, and statistical modelling (Swagat Ranjit 2018).

In un-supervised learning, ANN is only supplied with unlabeled examples as in you only provide the input data without a corresponding output (labels) to it. Clustering is a widespread type of unsupervised learning, which attempts to classify data into different clusters based on similarities (R. Sathya, 2013).

In clustering, great care should be paid in choosing suitable neural network topologies which typically relies on the data domain. As previously detailed, feedforward models

differ from recurrent models by not having feedback loops which changes the network behavior when trying to learn the data similarities for clustering. Additionally, various feedforward and various recurrent models such as: Elman recurrent and Jordan recurrent models also have their specific differences. Thus, selecting the proper topology and the proper model within the selected topology is crucial in the clustering performance which usually are selected using trial and error approaches (Arif Selçuk Öğrencı 2018).

This work in describes the design and implementation of several NN models for network data flowing traffic forecasting. The outcomes of such forecasting model can further be used to detect and classify several network attack types mainly DDoS attacks. A number of previous works have been suggested and implemented for such purposes with different degrees of success.

2.4.3 DDoS Attack

Distributed Denial of Service (DDoS) is a common type of cyber-attack in which a server (single IP address) gets flooded with too many connections/requests. This attack typically take place via a huge number of botnets to flood the server which prevents the normal user legit traffic from reaching to the attacked server (Marquette Poremba, Sue 2017). DDoS can be one of the following three main categories:

- Volume DDoS attack: utilizes high traffic bursts to flood the target network available bandwidth.
- Protocol DDoS attack: targets the target server resources
- Application DDoS attack: targets web-based type of applications and is one of the most critical DDoS attacks

This work is concerned with the volume DDoS attacks, in particular User Datagram Protocol (UDP) attack. This DDoS attack aims to flood the server's connection ports which results in the server continuously checking for the targeted ports looking for applications on these ports to serve and returns a destination unreachable messages which eats up the server resources preventing legit user traffic from going through to the server (Marquette Poremba, Sue 2017).

In the next section, different previous works that suggested different network bandwidth forecasting approaches including Autoregressive and Neural network models are summarized and compared to the proposed approach of this work.

2.5 Previous Works

The studies in (Marti, S 2010), (Shivashankar, T 2012) and (Cabrera, J. B. D 2008) have proposed different network bandwidth forecasting solutions based on Autoregression methods. Including autoregressive moving average (ARMA) model, fractal autoregressive moving average model (FARMIA) and an autoregressive integrated moving average (ARIMA) model.

However, according to (Yuanming Ding 2015), these (AR) models suffer from the need for large training data which is time consuming and increases the computational loads. In addition, they can only describe short term bandwidth forecasting with decent performances.

This work in terms of bandwidth forecasting has adopted neural network approach as mentioned similar to the approach in (Yuanming Ding 2015) which has only utilized a recurrent Elman approach for bandwidth forecasting while this work utilized two additional models: single-layer feedforward and multi-layer feedforward (deep learning) NN models which have managed to achieve higher forecasting performance when compared to Elman's.

Concerning DDOS attack-detection, the proposed strategy follows a similar approach to what's suggested in (Anjali, 2014) and improve up on it significantly. In (Anjali, 2014), DDOS attacks are detected using Competitive neural network for DDoS detection which is proven in Chapter 4 that it resulted in a much higher detection error rates compared to the proposed neural network models of this work.

This work is different from the previous works in that it forecast the traffic using three distinguishable NN models: Single-Layer Feedforward NN, Single-Layer Feedback Elman NN and Multi-Layer NN (deep learning). All the mentioned studies have either used AR models which is proven to provide worse forecasting and DDoS detection results in (Modi 2013) and that NN models are proven to perform better. Additionally, the proposed models are proven to provide much better performance as shown later in Chapter 4 when compared to (Anjali 2014) Competitive Learning NN for DDoS detection

Table 2.1 summarizes the previously proposed solutions compared to the proposed approach of this work.

Table 2.1: Summary of the most similar previous solutions

Proposed Approach	Comparison to Proposed Approach
ARMA (Marti, S 2010)	Achieved good outcomes, but are not efficient or precise in predictions as artificial neural network strategies for mid-term or long-term forecasting as proven in (Modi 2013)
FARMIA (Shivashankar, T 2012)	
ARIMA (Cabrera, J. B. D 2008)	
Elman NN (Yuanming Ding 2015)	The proposed single-layer feedforward and multi-layer feedforward (deep learning) NN models have managed to achieve higher forecasting performance
Competitive NN (Anjali 2014)	The proposed feedforward single-layer and multi-layer managed to achieve higher performance in DDoS detection

In the next chapter, a detailed description of the proposed approach and the selected data features are presented.

Chapter Three

Methodology and the Proposed Model

Introduction

This chapter presents a detailed description of the proposed methodology along with a description of the data set utilized in this work. Sections 3.1 covers the main proposed algorithm and the implemented ANN and DNN models along with a description of the DDoS detection procedure. Section 3.2 describes the DNN model and the main difference between DNN and ANN models.

3.1 Methodology

This section describes the proposed methodology used for both network traffic forecasting and DDoS detection. In subsection 3.1.1 describes the proposed algorithm and subsection 3.1.2 described the proposed approach models. The proposed algorithm follows the following steps:

1) The collected data set is first inputted into MATLAB
2) The data is filtered by removing any noise and redundancy of the inputted data
3) The desired data features are selected and the data is separated into training and testing data sets
4) The ANN models are created (feedforward and recurrent models)
5) The models are trained and tested several times using different model parameters to find the most suitable parameter values
6) The DDoS simulated data is built and inputted into the models the same way as described in steps 4 and 5

7) The DDoS detection performance is evaluated to determine the best NN model

3.1.1 Proposed algorithm for traffic forecasting and DDoS detection

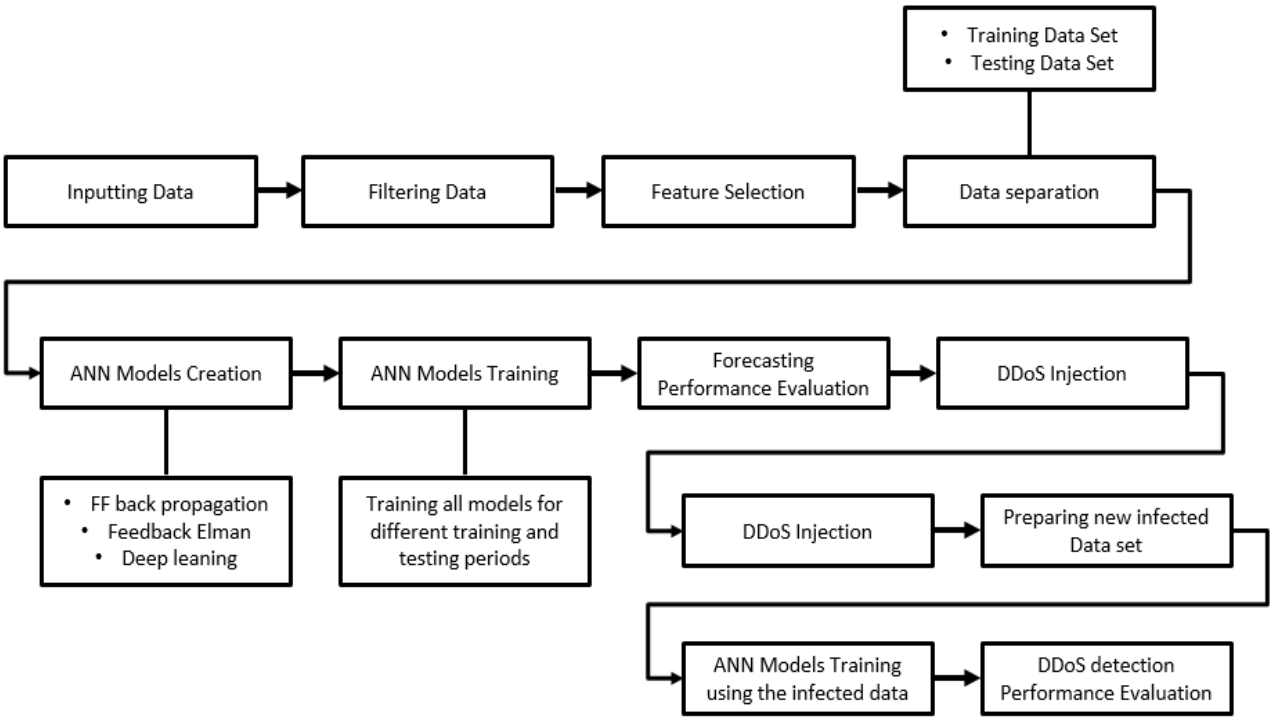

Figure 3.1: Proposed Algorithm

3.1.2 The Table of selected features

The proposed selected features (Table 3.1) are then re-formatted to match the ANN model input format. These features are selected to suit the provided data set domain. There are no certain guidelines for this data set in particular. Trial and error approach is what determines which features actually affect the outcomes and which doesn't

Table 3.1: Proposed approach selected features

Selected feature	Description
Year	Counts for annual seasonality
Month	Counts for monthly seasonality
Day of the week	Counts for weekly seasonality (1=Sat, 7=Fri)
Day of the month	Counts for daily seasonality
Semester	1st ,2nd and Summer (1=1st,2=2nd,3=Summer)
Holidays	Counts for off days (1=Work day,2=Off Day)
Registration	Counts for online registration period (1=Reg,2=No Reg)
Online exams period	Counts for online exams days (1=Exm,2=No Exm)
Hour of the day	Counts for hourly seasonality (1-24)

The formatting of the selected features shown in the table requires each feature to be inputted as one column of data and each training sample as a row. Before the data being fed into the proposed models, the data is first separated into two parts: Training data and testing data. The training part is used to train the proposed models and the testing data is used to evaluate the performance of each model.

After the data is prepared, the ANN and DNN models are created. As mentioned previously, the proposed approach consists of 3 NN models: Single-layer feedforward back propagation model, a single-layer recurrent Elman model and a multi-layer (deep learning) model.

Once the NN models are created, each model is fed with the prepared data to begin its training process. The topology of the proposed approach is supervised learning which as previously described requires feeding the NN model with training samples that contains the input data and the corresponding output data as demonstrated in table 3.1.

Before the training process begins, each NN model is configured with the most suitable NN parameters that best fit the provided data trends and seasonality. Table 3.2 shows these parameters and their selected values in the proposed approach.

Table 3.2: Proposed approach ANN models parameters

Parameter	Selected in proposed approach
The training sample size	Custom-selected each experiment
The testing sample size	Custom-selected each experiment
Number of hidden layers	1 (single layer) |2 (deep learning)
Number of input neurons	9
Number of hidden neurons	Custom-selected each experiment
Number of output neurons	1
Learning algorithm	Gradient Descent
Number of training epochs	5000
Activation function	Sigmoid
Training error type	MAPE
Testing error type	MAPE

As shown in table 3.2, some of the parameters are chosen to be the same for all experiments including: Number of input neurons, Number of output neurons, learning algorithm, Number of training epochs, Activation function and training error type MAPE (equation 2) (Paul Goodwin , 1999) while number of hidden layers and the training and the testing window sizes are specific to each experiment. These parameters are chosen via trial and error in which the network was run many times (20-30) time and each time the parameters were changed to find the most fit parameter values.

Learning algorithm is the algorithm the network uses to modify the weights of the connections between its nodes every time a new data sample is fed to the network. Number of training epochs determines how many times the provided data set is re-fed to the network with different samples order. The activation function is a mathematical function that determines the output of each neuron in the network. Sigmoid (equation 3) (Xintou Yin 2003) activation function is picked for this work as it's more suitable for semi-linear data trends which is the case for the utilized data set.

$$\mathrm{M} = \frac{100\%}{n} \sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right|, \tag{2}$$

Where n is the number of samples, At is the actual output value and Ft is the predicted value

$$f(x) = \tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{3}$$

Once the NN model is built and configured with the mentioned parameters, the training process begins. Once the training process is done, the model forecasting performance is

measured by comparing the forecasted outcomes of the NN model to the testing data part. As mentioned, the training data size and the testing data size are different for different experiments as will be detailed in Chapter 4.

Once the NN forecasting performance is evaluated. It is stored in a separate array and the training session is repeated which different NN parameters (shown in table 3.2). The parameters are re-selected in a greedy manner of trial and error where they are modified after each and every training session to find the parameters that manage to achieve the highest possible performance (lowest error rates).

After the network traffic forecasting has taken place and the optimum forecasted traffic outcomes are extracted. The DDoS detection procedure begins.

First, the network is injected with simulated DDoS attack in the form of UDP flooding (Marquette Poremba, Sue 2017) using Opnet modeler software and the network traffic is captured again. This creates a new set of infected data that contains two features: The Actual Infected Traffic and the type of traffic (infected and non-infected).

This means that two data sets are now available for analysis for the same network configuration: Legit traffic data set and an infected traffic data set. For DDoS detection, a new data set is formed as detailed in Table 3.3.

Table 3.3: Infected DDoS data sample

Predicted legit	Actual infected	Actual infected - Predicted	Status (1=not infected & 2 = infected)
110	114	4	1
140	139	1	1
144	144	0	1
155	157	2	1
152	155	3	1
136	139	3	1
136	160	11	2
138	162	24	2
132	166	34	2
126	126	0	1
122	122	0	1

As shown in the table, the difference between the forecasted legit traffic and the DDoS infected traffic is then stored in a newly formed data set that represents the difference between legit traffic and DDoS infected traffic (Actual – forecasted).

After the new data set is formed, the NN models are trained again utilizing the new data set which only uses the last two columns of Table 3.3 (Actual-forecasted and Status). The data set is treated the same way as before by being separated into training part and testing part and the NN models are configured, trained and had their performance evaluated using the same procedure as the previous procedure.

3.1.3 The proposed approach models

As mentioned in the previous section, this work is implemented over two main procedures: *Legit network traffic forecasting* and *Infected network traffic for DDoS detection*. Both the network traffic forecasting and the DDoS detection procedures are implemented using the same following models:

1) Single-layer Feedforward Backpropagation ANN
2) Single-layer Recurrent Elman ANN Model
3) Multi-layer feedforward Model (Deep Learning)

These three models are compared to the Competitive NN (Anjali 2014) in terms of DDoS detection via bandwidth forecasting in this work.

Both the Single-layer Feedforward ANN model and the Single-layer Recurrent Elman ANN Model were detailed in Chapter 2 of this work. In the next section, a comparison between the Multi-layer feedforward Model (deep learning) and Single-layer Feedforward Backpropagation is presented.

3.2 Multi-layer (Deep Learning) vs. Single Layer Feedforward models

Figure 3.2 below shows the main difference between single layer feedforward model and multi-layer deep learning feedforward model (Weibo Liu, 2017).

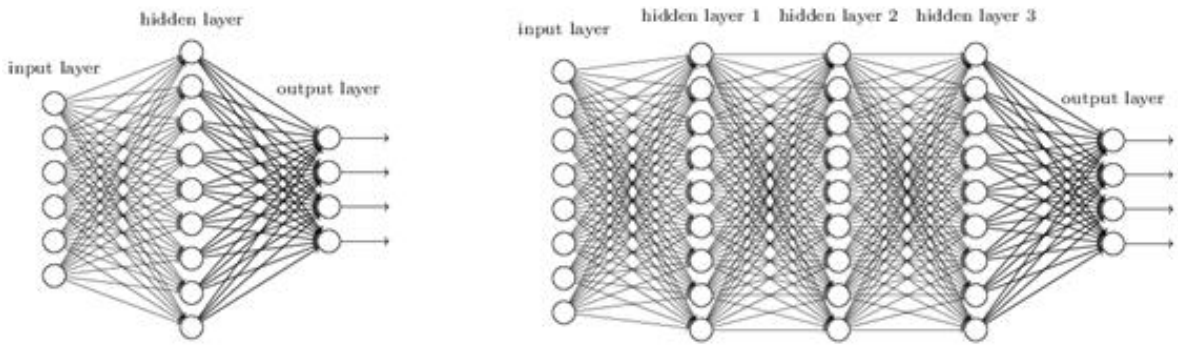

Figure 3.2: Single-layer feedforward (left) vs. Multi-layer feedforward (right) (Weibo Liu, 2017).

As demonstrated in figure 3.2, the main difference between the single layer and the multi-layer is the number of hidden layers in each model. When the hidden layers are more than one layer, the model becomes a multi-layer feedforward model also known as "deep learning" model which according to (Weibo Liu, 2017) can sometimes provide better function estimations than single layer models.

Each layer in the deep learning model holds its own parameters just like any hidden layer in terms of number of hidden neurons, activation function of each neuron, and neuron bias values. There is no final rule that clearly specifies whether single-layer models is better or worse than multi-layer models and are different from one application domain to another. Thus, it is worth being investigated in this work (Weibo Liu, 2017).

In the next chapter, a description of the bandwidth forecasting and DDoS detection experiments are presented in detail using all three mentioned models along with a comparison between each model performance.

Chapter Four

Experimental Results and Discussion

This chapter presents detailed description of the experimental results of the proposed approach for all three mentioned models in both network traffic forecasting and DDoS detection. The experiments are separated into a number of sections based on the training data set and the testing data sets sizes. The most common approach for separating training and testing data of a single data set is 70% (training) and 30% (testing). This ratio was maintained in the held experiments of this work whenever possible.

Table 4.1 shows a sample of the used data which demonstrates the selected features (detailed in Chapter3) that were fed into the proposed models.

Table 4.1: Proposed Approach input & output data for legit traffic

Input									Output
Year	Month	Day of the week	Day of the month	Semester	Holidays	Registration period	Online Exam Period	Hour of the day	Traffic Mbps
2019	1	1	6	2	1	2	2	1	109
2019	1	1	6	2	1	2	2	2	85
2019	1	1	6	2	1	2	2	3	79
2019	1	1	6	2	1	2	2	4	82
2019	1	1	6	2	1	2	2	5	94
2019	1	1	6	2	1	2	2	6	93
2019	1	1	6	2	1	2	2	7	98
2019	1	1	6	2	1	2	2	8	100
2019	1	1	6	2	1	2	2	9	113
2019	1	1	6	2	1	2	2	10	136
2019	1	1	6	2	1	2	2	11	135
2019	1	1	6	2	1	2	2	12	159
2019	1	1	6	2	1	2	2	13	153

4.1 Experiment 1: 27 Days Training and 3 Days Testing

In this experiment, a dataset totaling 30 days, wherein every day consists of 24 traffic values, was segregated according to 27 Days of Training and 3 Days of Testing and then fed to the 3 proposed ANN models. Figures [4.1-4.3] showcase the forecasting results in comparison to the actual traffic values.

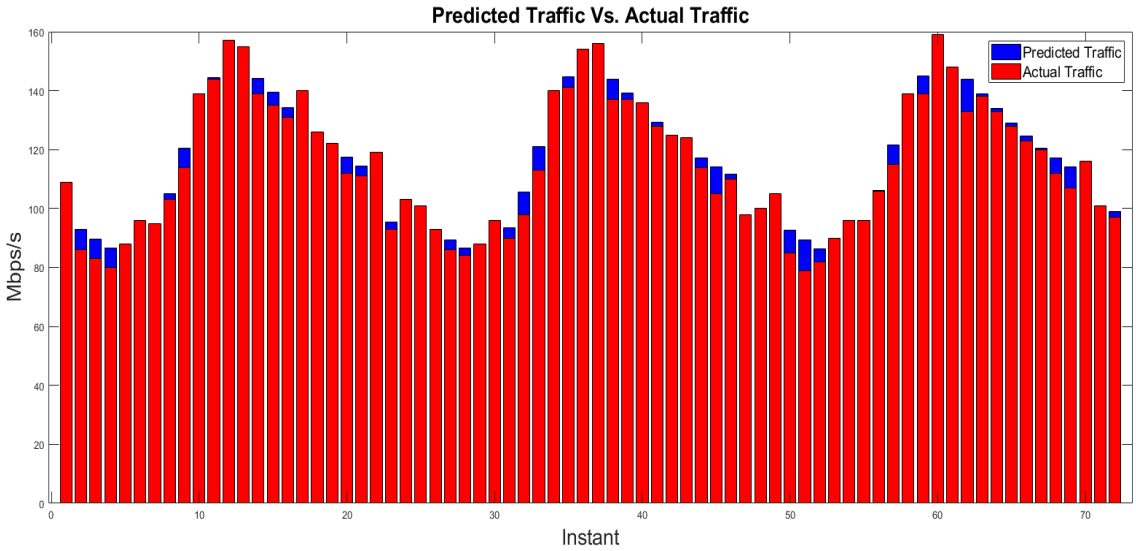

Figure 4.1: Forecasting performance of single-layer FF ANN

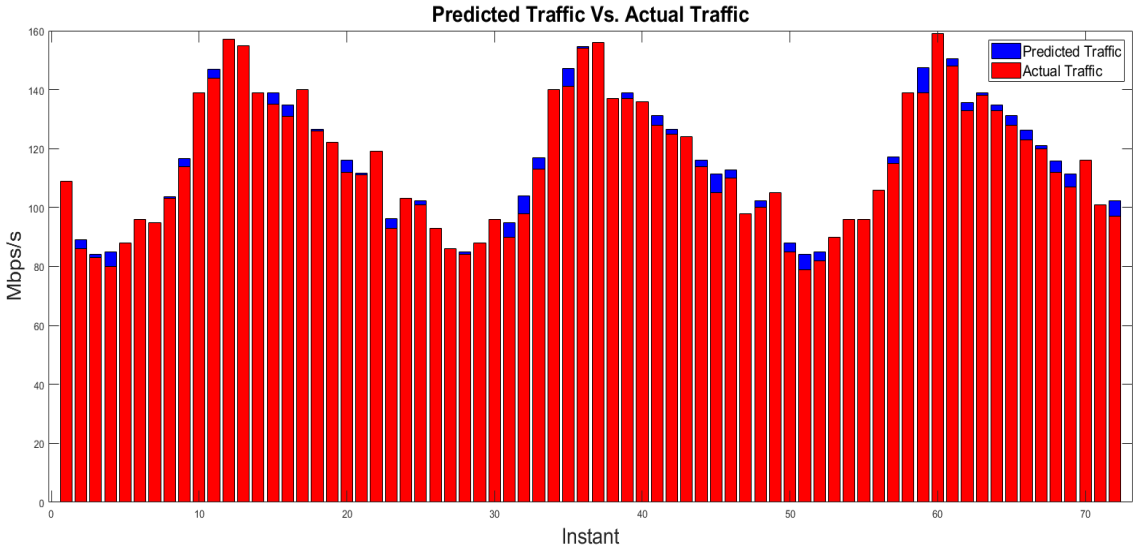

Figure 4.2: Forecasting performance of Multi-Layer FF NN

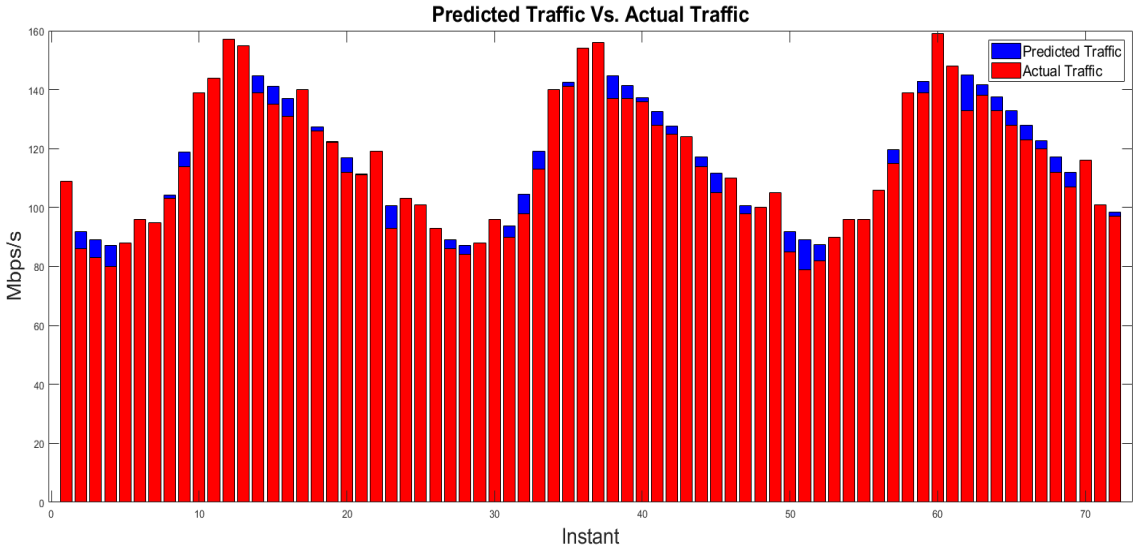

Figure 4.3: Forecasting performance of single-layer Recurrent ANN

The error values of all utilized models are summarized in table 4.2.

Table 4.2: 27 Days Training and 3 Days Testing results

Model	1st layer Hidden Neurons	2nd layer Hidden Neurons	Error %
Single-Layer Feedforward ANN	3	N/A	4.2
Single-Layer Recurrent ANN (Yuanming Ding 2015)	5	N/A	4.48
Multi-Layer (Deep Learning)	3	3	2.9

As demonstrated in the table, every model has accomplished similar experimental results, wherein the recurrent featured the highest errors rates. For as with recurrent models, the prior sample prediction impacts the forecast result of the following sample. This causes slightly higher error rates than feedforward models when the current forecasted value isn't dependent on the value of the previous sample which is the case here.

4.2 Experiment 2: 10 Days Training and 4 Days Testing

In this experiment, a dataset totaling 14 days, wherein every day consists of 24 traffic values, was segregated according to 10 Days of Training and 4 Days of Testing and then fed to the 3 proposed NN models. The error values of this experiment are summarized in table 4.3. Figures [4.4-4.6] showcase the forecasting results in comparison to the actual traffic values.

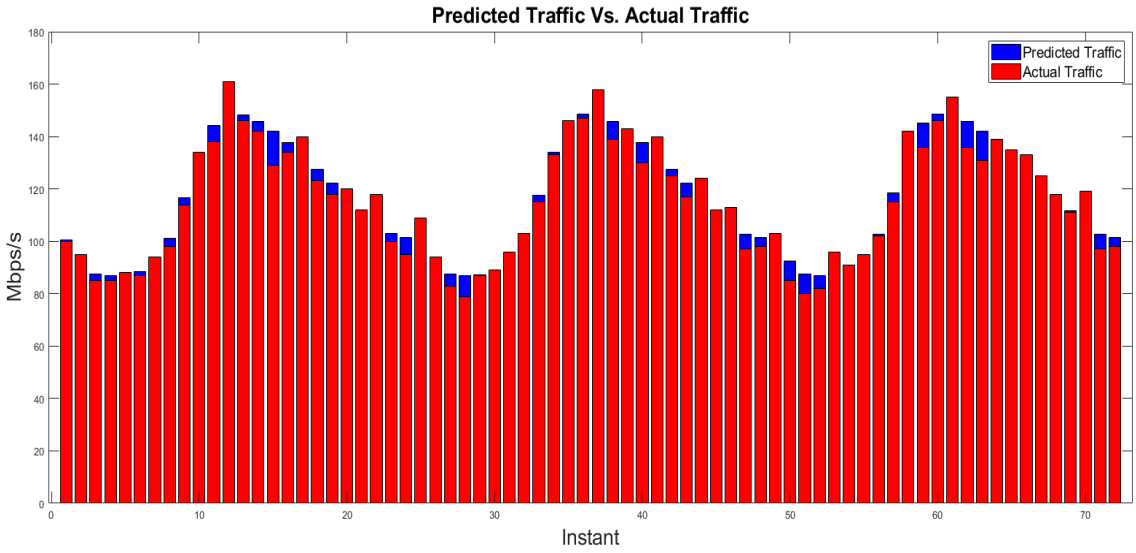

Figure 4.4: Forecasting performance of single-layer FF NN

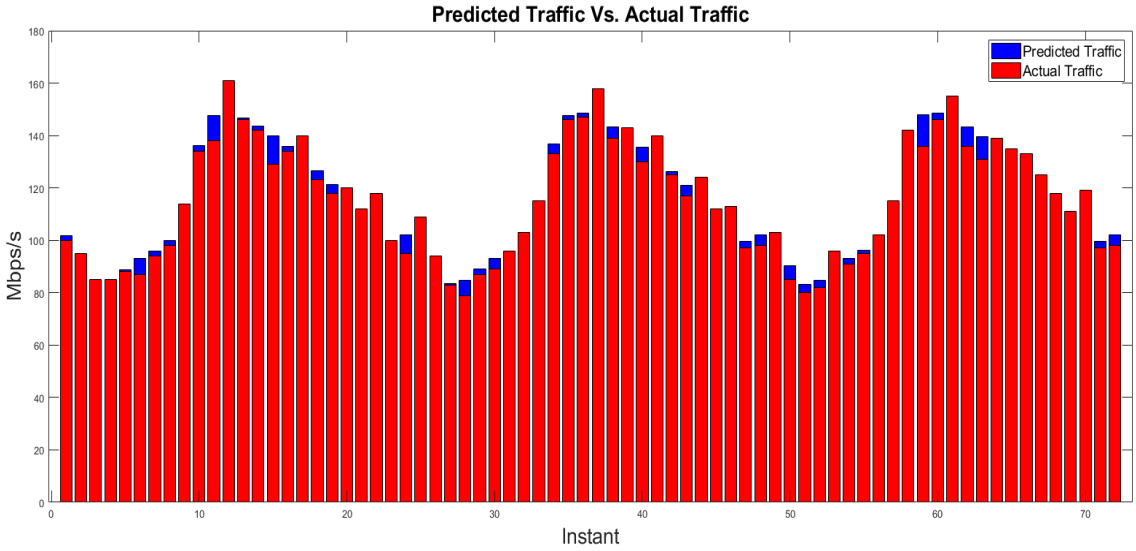

Figure 4.5: Forecasting performance of Multi-Layer FF NN

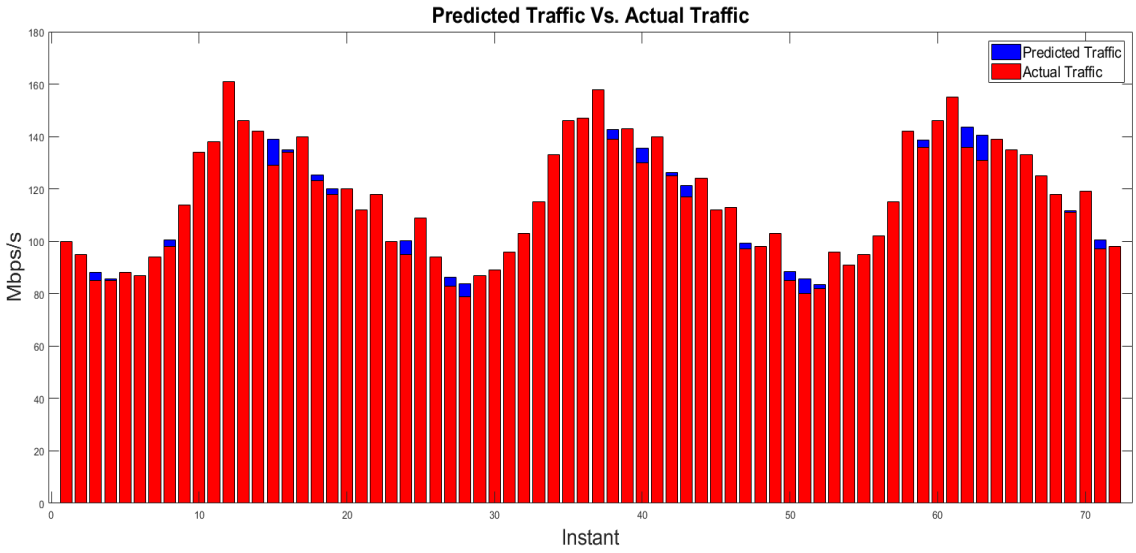

Figure 4.6: Forecasting performance of single-layer Recurrent NN

Table 4.3: 10 Days Training and 4 Days Testing results

Model	1st layer Hidden Neurons	2nd layer Hidden Neurons	Error %
Single-Layer Feedforward ANN	3	N/A	3.82
Single-Layer Recurrent ANN (Yuanming Ding 2015)	5	N/A	4.25
Multi-Layer (Deep Learning)	3	3	3.60

As displayed the table, every model again accomplished similar experimental results, wherein Elman recurrent featured the highest errors rates, for the same rationales noted in the prior experiment.

4.3 Experiment 3: 5 Days of Training and 2 Days of Testing

In this experiment, a dataset totaling 7 days, wherein every day consists of 24 traffic values, was segregated according to 5 Days of Training and 2 Days of Testing and then fed to the 3 proposed NN models. The error values of this experiment are summarized in table 4.4.

Table 4.4: 5 Days of Training and 2 Days of Testing results

Model	1st layer Hidden Neurons	2nd layer Hidden Neurons	Error %
Single-Layer Feedforward ANN	3	N/A	4.48
Single-Layer Recurrent ANN (Yuanming Ding 2015)	5	N/A	5.61
Multi-Layer (Deep Learning)	3	3	4.47

As displayed in the table, feedforward models again offered superior forecasting performance. Nevertheless, the experimental differences between Elman's model and the 2 other schemes is not as significant, as the data samples in this test followed smoother growth than that observed in the prior two experiments.

These results show that the both feedforward models (single-layer and multi-layer) have managed to achieve higher network performances compared to the recurrent model suggested in (Yuanming Ding 2015).

This is due to the feedback loops existing in the recurrent model which implies that the previous forecasting sample heavily impacts the current forecasting sample which when not the case, it increases the forecasting error margin as shown in the experiments.

However, for DDoS detection procedure, all the three proposed models are compared to the previously proposed model (Competitive ANN) as suggested in (Anjali 2014). The authors in (Anjali 2014) have utilized a model for DDoS detection with training and testing data built is a similar fashion to how the new data set is built in this work as shown in Table 3.3.

The forecasted data size of this experiment are only of 3 days (72 sample), the newly created data set that is infected with DDoS attack as described in section 3.1.1 is separated into two sets: training set of two days and testing set of one day in size which pushes the proposed approaches to the hardest possible case.

Table 4.5 shows the custom-selected model parameters for Elman model and both the single and multi-layer FF backpropagation models along with the model proposed in (Anjali 2014).

Table 4.5: DDoS detection results

	Model	layer Hidden Neurons	Error %	Success %
Proposed Approaches	Single-Layer Feedforward ANN	3	0	100
	Single-Layer Recurrent Elman ANN	5	4.16	95.84
	Multi-Layer (Deep Learning)	3	0	100
Previous Work	Competitive Algorithm NN (Anjali 2014)	2	4.16	95.84

Table 4.5 shows that the proposed single and multi-layer feedforward (deep learning) preformed the best in this experiment while Elman's and the proposed model in (Anjali 2014) being the models with the highest error rates. This is due to the fact that the proposed model in (Anjali 2014) is un-supervised ANN model which shows its weaknesses compared to the supervised models proposed in this work.

The next Chapter concludes this work and covers some future work suggestions.

Chapter Five

Conclusion and Future Work

5.1 Conclusion

The proposed NN models are built in MATLAB software using a two month in size historical local network bandwidth data to train three NN models: Single-layer feedforward back propagation model, a single-layer recurrent Elman model and a multi-layer (deep learning) model. The proposed models achieved extremely low error rates of errors below 3% in some experiments.

Due to the precise forecasting performance of the proposed models, the forecasting outcomes were further utilized in DDoS detection. Different training windows and testing periods were utilized in the proposed model over several different scenarios. Even though some previous works have managed to detect such attack, these were either too costly given the server computational resources involved or else insufficiently practical in the majority of actual cases, since such methods entail the challenging capture of huge datasets.

The proposed models were constructed with MATLAB for both traffic forecasting and DDoS detection and have managed to achieve error rates of below 3% for both traffic forecasting and DDoS detection while the closest previous work has managed to achieve error rates as high as 16% using their Competitive NN approach when using the same data for this work.

5.2 Future Work

In the future, Other NN architectures could be tested such as Jordan's architecture in both single-layer and multi-layer structures. Additionally, Other AI approaches such as modified autoregressive methods can also be investigated along with support vector machine models.

References

Adeilson Marques da Silva Cardoso, 2018 "***Poster Abstract: Real-Time DDoS Detection Based on Complex Event Processing for IoT***" in IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI)

Anjali. M, 2014 "***Detection of DDoS Attacks based on Network Traffic Prediction and Chaos Theory***", in International Journal of Computer Science and Information Technologies,

Arif Selçuk Öğrencı, 2018" ***Anomaly detection in walking trajectory*** " , in 2018 26th Signal Processing and Communications Applications Conference (SIU)

Aykut Tahtirvanci,2018 " ***Classification of EEG signals using spiking neural networks*** " , in2018 26th Signal Processing and Communications Applications Conference (SIU)

Dingding Zhou, 2013 "***Network traffic prediction based on ARFIMA model***" in International Journal of Computer Science Issues (IJCSI)

Gerd Bramerdorfer, 2014" ***Using FE Calculations and Data-Based System Identification Techniques to Model the Nonlinear Behavior of PMSMs*** " , in IEEE Transactions on Industrial Electronics (Volume: 61 , Issue: 11 , Nov. 2014)

Grégoire Mesnil, 2015 ” ***Using Recurrent Neural Networks for Slot Filling in Spoken Language Understanding*** “ , in IEEE/ACM Transactions on Audio, Speech, and Language Processing Volume: 23

Han, (2010). “***Data Mining: Concepts and Techniques. 3rd edition***” Published in August 2000

K. C. Sriharipriya,2017,” ***Artifical neural network based multi-dimensional spectrum sensing in full duplex cognitive radio networks***” in 2017 International Conference on Computing Methodologies and Communication (ICCMC)

Leonid Kupershtein, 2016” ***Neural network approach in the stroke diagnosis*** “ , in 2016 IEEE First International Conference on Data Stream Mining & Processing (DSMP)

LeZhangP.N.Suganthan, 2016 ” ***A survey of randomized algorithms for training neural networks 2016*** “ , in Information Sciences Volumes 364–365, 10, Pages 146-155

Modi, 2013 "***A survey of intrusion detection techniques in cloud***." In Journal of network and computer applications.

R. Sathya, 2013” ***Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification*** “, in International Journal of Advanced Research in Artificial Intelligence (IJARAI)

Shubhankar Kapoor, 2016" ***Design and implementation of a robust system for recognizing alphabets using artifical neural network*** ", in 2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)

Swagat Ranjit,2018" ***Comparison of algorithms in Foreign Exchange Rate Prediction*** ", in 2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS)

Vinayaka Jyothi,2016 "***BRAIN: BehavioR Based Adaptive Intrusion Detection in Networks: Using Hardware Performance Counters to Detect DDoS Attacks***" in 2016 29th International Conference on VLSI Design and 2016 15th International Conference on Embedded Systems (VLSID)

Weibo Liu, 2017 "***A Survey of Deep Neural Network Architectures and Their Applications***" in Neurocomputing Volume 234, 19, Pages 11-26

Marquette Poremba, Sue."***Types of DDoS Attacks."*** ESecurity Planet: Internet Security for IT Professionals, 2017, www.esecurityplanet.com/network-security/types-of-ddos-attacks.html.

Marti, S., Giuli, T.J. and Lai, K., 2010 "***Mitigating routing misbehavior in mobile Ad Hoc networks***," in Proc. the 6th Annual Intl. Conf. on Mobile Computing and Networking

Cabrera, J. B. D., Gutierrez, C. and Mehra, R. K., 2008 "***Ensemble methods for anomaly detect- ion and distributed intrusion detection in mobile Ad Hoc networks***," in Information Fusion.

Shivashankar, T., Sivakumar, B., Varaprasad, M., 2012"***Identification of critical node for the efficient performance in MANET***," in International Journal of Advanced Computer Science and Applications.

Yuanming Ding 1,2, Jiayao Gao 2,3 and Xue Wang, 2015 "***Ad Hoc network traffic prediction based on the Elman neural network***" in 3rd International Conference on Machinery, Materials and Information Technology Applications.

Paul Goodwin , 1999 "***On the asymmetry of the symmetric MAPE***" in International Journal of Forecasting 15 (1999) 405–408

Xintou Yin, 2003 " ***A Flexible Sigmoid Function of Determinate Growth***" in Annals of Botany, Volume 91, Issue 3, February 2003, Pages 361–371