COGNITIVE ARTIFICIAL INTELLIGENCE – A COMPLEXITY BASED MACHINE LEARNING APPROACH FOR ADVANCED CYBER THREATS

by

Sana Siddiqui

A Thesis submitted to the Faculty of Graduate Studies of

The University of Manitoba

in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Electrical and Computer Engineering,

University of Manitoba, Winnipeg, MB, Canada

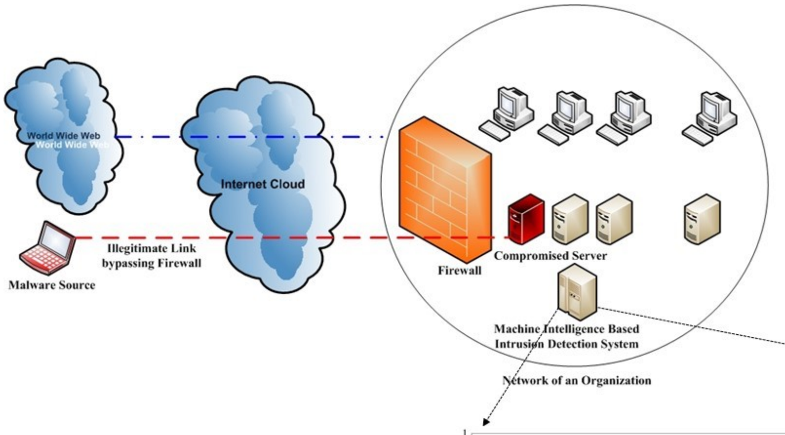
Abstract

Application of machine intelligence is severely challenged in the domain of cyber security due to the surreptitious nature of advanced cyber threats which are persistent and defy existing cyber defense mechanisms. Further, zero day attacks are also on the rise although many of these new attacks are merely a variant of an old and known threat. Machine enabled intelligence is limited in solving advanced and complex problems of detecting these mutated threats. This problem can be attributed to the single scale analysis nature of all the machine learning algorithms including but not limited to artificial neural networks, evolutionary algorithms, bio-inspired machine intelligence et al.

This M.Sc. thesis addresses the challenge of detecting advanced cyber threats which conceal themselves under normal or benign activity. Three novel cognitive complexity analysis based algorithms have been proposed which modify the existing single scale machine learning algorithms by incorporating the notion of multiscale complexity in them. Particularly, network based threats are considered using two different publicly available data sets. Moreover, fractal and wavelet based multiscale analysis approach is incorporated in decision making backbone of k-Nearest Neighbours (k-NN) algorithm, Gradient Descent based Artificial Neural Network (ANN), and Hebbian learning algorithm. The classification performance of these algorithms is compared with their traditional single scale counterparts and an improvement in performance is observed consistently.

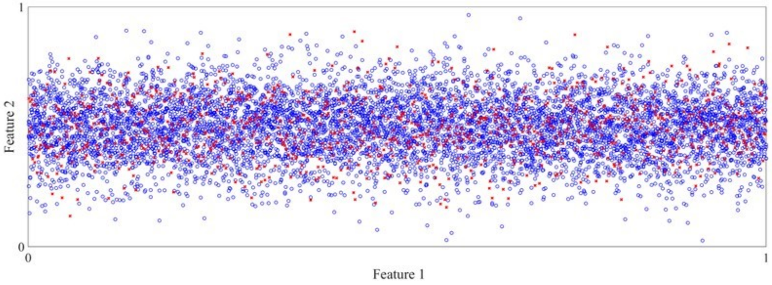
This improvement is attributed to the usage of multiscale based complexity measures in the analysis of algorithm, features and error curve. The notion of multiscale evaluation reveals the hidden relationship which otherwise are averaged out when observed on a single scale. Also, the problem of class overlap which arises due to the stealth nature of cyber-attacks is addressed using the same concept. Conceptually, it is analogous of human cognitive capability employed in pattern discovery from complex objects based on their knowledge about how to connect and correlate various aspects together. It is imperative to note that this multiscale relationship should be a representative of the complexity measure of whole object so that it can characterize patterns based on various scales.

Problem illustration



An abstract network diagram connected to internet and the malicious source which bypassed the firewall

A 2-dimensional feature space showing overlapping samples of normal(o) and malicious(x) packets flowing through the firewall



Acknowledgment

If I have seen further, it is by standing on the shoulders of giants.

a. Sir Isaac Newton (1643 AD – 1727 AD)

I am extremely grateful to everyone who contributed directly or indirectly during the research, development and drafting of this M.Sc. thesis.

Foremost, I am thankful to my thesis advisor Prof. Dr. Ken Ferens for accepting me into his research group and motivating me to explore the complex domain of cognitive intelligence in machine learning to devise solutions for current cyber-security challenges. His unrelenting insights and guidance is and will always be appreciated. He is a true Canadian educational leader who always motivates his students in pursuing their goals of seeking excellence in academics, research and career. I cannot express my heartfelt gratitude for his support, trust, motivation, advice, and continued guidance towards my pursuits.

I am very grateful for my committee members for their interest in my research. Particularly, I would like to thank Prof. Dr. Yang Wang for his time and interest in evaluating my thesis. Also, I am extremely thankful to Prof. Dr. Witold Kinsner for his graduate course on fractals and chaos engineering where I learned how to model cognition using fractals and chaos. Indeed, his relentless efforts and personal attention instilled in me the confidence to venture into the difficult domain of cognitive machine intelligence. Moreover, I would like to extend my deepest gratitude to Prof. Dr. Ken Ferens (ECE), Prof. Dr. Ian Jeffrey (ECE), Prof. Dr. Sherif Sherif (ECE), and Prof. Dr. Ben Li (Computer Science) whose graduate classes on computational intelligence, advance matrix algorithms, signal processing, estimation theory, and combinatorial optimization helped steer my research goals.

Every experiment and algorithm I mentioned in this thesis is made possible with the continuous collaborative support of my lab colleague and a Ph.D. candidate Muhammad Salman Khan. Indeed his propensity in simplifying the complex problems and generating innovative ideas is commendable. I truly appreciate his sincere support and guidance during the course of my research work.

I am also thankful to Amy Dario, ECE graduate student advisor, for her commendable support in facilitating our administrative and paper work.

Further, I would like to extend my deepest gratitude to IMPACT Cyber Trust USA (formerly PREDICT), Mila Parkour of Contagio Malware Repository and UNSW-NB15 data set provider from University of New South Wales for sharing their data sets which proved to be radical in validating the proposed algorithms in the presented thesis.

Also, I am grateful to University of Manitoba and Canada for providing me an opportunity to join the league of world class researchers by allowing me to study, experiment, and pursue my ambitious research goals. A special thanks goes to the Manitoba government for motivating me by awarding Manitoba Graduate Scholarship in recognition of my scholastic achievements.

I could not have progressed without the support of my beloved family and friends who always remained supportive, always trusted me and stood by me through the test of times.

Finally, I am thankful to Almighty Allah (SWT) for everything He bestowed me with and for blessing me with an opportunity to be thankful to everyone.

Thesis synopsis

With the advent of e-commerce, cloud computing, social networks and Internet of Things (IoT), human life has gradually transformed into a cyber space which is virtual but critical not only for individuals but for the organizations and governments. Cyber security is an ever evolving domain which requires expertise from interdisciplinary subjects including but not limited to machine learning, data mining, signal processing, information theory, cognitive science, neuropsychological studies of mind and brain. Further cyber security can be classified in different aspects. For example, cryptography, signature and behavioral analysis of attacks, host and network based threat detection and mitigation, cyber security policy and management. With the evolution in technology, threat actors are also getting smarter and have evidentially proven their ingenuity in breaking into sophisticated cyber defenses which include not only dedicated security devices such as firewalls, antimalware software, intrusion detection and prevention mechanisms but also involve human experts and teams. Using artificial intelligence and machine learning techniques for cyber security is relatively a new domain although practical machine learning itself has found its roots since 1950's, when Alan Turing proposed a Turing test to evaluate if a machine has real intelligence.

In the domain of cyber security, artificial intelligence and machine learning have found enormous application in the context of anomaly detection. Further, various heuristic approaches are developed to process high dimensional data sets so that meaningful features can be extracted for threat detection with optimal resource consumption. Major impediments in the successful application of contemporary machine intelligence techniques in cyber security are driven not only by the hyper exponential growth in the volume, velocity, veracity and variety of the data in the past few years, but also due to advanced morphing and mutating threats where threat actors devise strategies, tactics, techniques and procedures to bypass automated threat detection systems, which are heavily reliant on pre-known signatures of the threats.

Another challenge for machine intelligence techniques lies in the inherent methodology to generate decision boundary between the normal and malicious samples. These boundaries require a feature space and therefore, threat actors exploit this requirement of machine learning by changing the feature space [1]. It can be stated that feature space is no more unique and changes dynamically. These signatures are not only the static behavior or characteristic of a threat e.g. byte code of a file, malicious IP addresses and packet content, but also the knowledge of a particular behavior that is known to be malicious. For example, port scanning of a network is always treated malicious unless otherwise performed for maintenance or legitimate purpose. This behavior can be treated as a flow or behavior signature of a threat. However, a threat actor still can perform port scanning that may span for long duration of time with intentional breaks in scanning to avoid anti scanning software which tries to search for port scanning using specific time interval windows. Threat actors can spread the scanning operation on various time windows and thus can be avoided by detection engines. This strategy can be termed as morphing that disguise the threat as normal and increases the probability of not being detected.

Primary challenge that this thesis addresses is identification of stealth threats on network data e.g. packet based communications, where the unique features that can be used to identify specific category of threats are no more unique and apparently threats cannot be distinguished from normal data, because they occupy the same feature space coordinates as that of legit instances over which traditional machine intelligence approaches fail in finding a unique decision boundary. It has been shown in this research that contemporary internet data sets having threats and normal samples render overlapping and indistinguishable feature space. Also, this thesis proposes modification in existing machine intelligence methods using multiscale analysis that is akin to cognitive human analytical methods of finding hidden patterns using complexity analysis. Empirical results show promising results in detection performance and are tractable due to the elegance of multiscale tools of fractal and wavelets.

Thesis organization

This thesis presents novel application of multiscale as a tool to improve cognitive complexity based analytical methods of machine intelligence algorithms to detect advanced cyber threats. Further, it discusses advanced stealth techniques used by cyber threat actors which are posing unprecedented challenges for existing machine intelligence algorithms. Therefore, two major discussions are covered in this thesis; (1) cyber threats, and (2) machine intelligence and cognition to detect them.

Chapter I discusses the motivation behind research in cyber security domain. Further, it defines the scope of this thesis along with a formal problem statement being addressed in this thesis.

Chapter II provides an overview on contemporary intrusion detection systems, techniques and methodologies. In particular, it describes the strength and challenges of existing machine learning techniques in the context of cyber security.

Chapter III gives an overview of the current state-of-the-art cyber threats landscape and the existing cyber defence mechanisms. Also, this chapter signifies the importance of cyber kill chain which is a tool used by cyber security experts to analyze data and detect threats.

Chapter IV signifies the concept of cognitive cyber intelligence and the role of complexity in cognitive analysis. The use of multiscale tools such as fractals and wavelets is also discussed to highlight their significance in establishing complexity of objects.

Chapter V brings details of the data sets and feature extraction which is used in this work to model and implement cognitive complexity measures for advanced threat detection. This chapter also discusses the evolving challenges of feature space obfuscation and class imbalance which are persistently used as tools to mutate the behavior of cyber-attacks by the threat actors.

Chapter VI discusses details of three machine intelligence algorithms i.e. k-NN, gradient descent based neural network and Hebbian learning algorithm in the context of their performance to detect threats which is followed by improvements obtained by modifying these algorithms using cognitive multiscale methods.

Chapter VII provides a thorough discussion on the detection performance along with comparison of both traditional and modified algorithms.

Chapter VIII concludes this thesis and also mentions about the possible future work as an extension of this thesis.

Table of Content

List of Tables

List of Figures

Chapter I: Introduction

1. Motivation

Internet based attacks, terrorism and crime has become extremely prevalent with the increased dependency of organizations and individuals on internet services and applications. On one hand, this has resulted in enhanced accessibility, easy communication and improved quality of life. However, this situation has also exacerbated the problem of securing the personal information and preventing its unauthorized access by threat actors. The fundamental challenge is to detect advanced morphing attacks which imitate the behavior of normal internet activities in order to prevent their detection. Although, cutting-edge cyber-defence mechanisms like firewall, antivirus, packet inspectors and intrusion detectors have evolved from using either signature based techniques or machine learning strategies for threat detection to a combination of both. Nevertheless, it is still not able to address the challenge of dynamic evolution observed in advanced cyber-attacks recently. This is due to the heavy reliance of detection mechanisms on historical events which is in addition to the static learning approaches that hinders the self-evolutionary learning capability of these systems. Generally, cyber threat detections systems are updated offline after the attack source is identified and severe damage has already been caused.

The productivity and competitiveness of governments and citizens is being challenged each year with the loss caused by the cyber-crime, state-sponsored attacks and lone hackers. Moreover, the rate at which vulnerabilities and specifically zero-day attacks are being discovered, it will not take it long to become a commercial product that can be exploited for carrying out malevolent activities. Therefore, there is a need to introduce dynamic learning capabilities in existing state-of-the-art machine learning algorithms. This thesis provides a proof of concept of the proposed cognitive solution that can be used by the organizations, banks, and businesses alike to autonomously detect advanced cyber fraud, intrusion, or attack.

As known, no silver-bullet solution exists for cyber security problem which implies that it is nearly impossible to stop internet based malicious activities completely. However, the primary objective of this research is to refine cognition in machine learning algorithms to detect mutating cyber threats in network flows. The driving factor behind this effort is to analyze and assess the classification performance of standard machine learning algorithms by altering their core logic to utilize strategies which mimic human brain's inferential and perception learning. In other words, this work was carried out to experimentally investigate the effect of applying human brain's model and intelligence processing methodology towards engineering application of cyber defence.

Finally, it is expected that the benefit of this work will not be confined to the improvement of cyber security research but, will be applied to other domains as well having emphasis on introducing intelligence in data processing including but not limited to financial, medical, and economic analysis.

2. Problem definition and scope

According to IBM [2], 90% of data today in the world has been created in the past 2 years. This is because in past few years, internet has become an express way for establishing instant communication across the world and a round-the-clock channel for ecommerce and banking services. This has transformed the world into a highly-connected network where information sharing should be characterized by confidentiality, integrity, and security. However, the reality is far cry from the current situation of the cyber-realm which encounters frequent multiple breaches whereby each compromising millions of records frequently. The resultant loss is not limited to the financial stress only but extends to the reputation damage, critical information compromise, and often physical harm.

Stealth and persistence is the key to ensure successful cyber-assaults. In order to achieve this, threat actors employ a combination of simple and highly complex techniques to ascertain that the malicious activity do not leave a distinct signature behind. It implies that from infiltration into target system to final disruption of the host and network is only attainable if the malevolent activity is not detected by the cyber defence tools at any stage of the kill chain. The only way to realize this objective is to follow the normal activities pattern such that the negative (normal) and positive (attack) feature samples when observed on a global scale overlap each other. Consequently, determination of a unique behavior to detect anomaly becomes increasingly difficult. This situation is further aggravated by the imbalance in the cyber-security datasets which are often characterized by a very low positive (attack) to negative (normal) samples ratio. Hence, the task of finding the attack (positive) samples becomes analogous to finding a needle in a haystack problem [3].

Nowadays data is generated from multiple data sources, for example: computers, cell phones, medical devices, home automation systems. Each of the device connected to the internet has variable data generation and data exchange speed thus, adding to the complexity of the network. Moreover, with the extensive use of the mobile applications, newer and diverse protocols which have open architectures are finding their way into the network. Further, the data can be accessed simultaneously from multiple location causing the data surveillance and control mechanisms to become increasingly intricate. Therefore, for the purpose of providing a basic working model of proposed cognitive approach for threat detection, the network based attacks have been selected as the focus of this study.

In other words, the data logs generated on a host machine are not being considered. Also, the data contributed on the internet by the mobile devices is not included in the scope of this study. Only, the TCP or more specifically, HTTP based communication have been the topic of research whether it is generated by a desktop system or a handheld device. The primary reason for this choice is the statistical fact according to which HTTP protocol is being exploited by 92% of the cyber threats in various ways [4]. In addition, different types of internet threats i.e. worms, exploits, bots, insider threats have been grouped into one positive class thus, translating the scenario into binary classification problem with extremely varying complexity levels.

The contemporary machine learning algorithms perform analysis on a single scale, masking out the local behavior of the system which indicates the presence of any underlying complex phenomenon. This work aims to take into consideration the multiscale complexity measure by using wavelet and fractal as mathematical tools to differentiate anomaly from an overlapping legitimate samples. Moreover, the proposed solution should not only classify known threats correctly but, new and unknown threats as well which include both the morphed variants of the known attacks and a completely new threat. Formally, this thesis proposes solution for the problem stated as follows:

How to classify known and new attack samples from legitimate samples with improvement in relative precision and accuracy, using complexity measure as a characteristic feature in the traditional machine learning algorithms, considering the problems of class overlap on feature space, imbalance class distribution, and heavy-tailed statistical data distribution are also addressed while reducing both false positives and false negatives simultaneously?

Chapter II: Contemporary cyber threat detection approaches

With the incorporation of internet based intelligent devices in various sectors of business and personal life, the security of resources, network and connectivity has become a critical challenge. While the unification in different communication systems, protocols and standards has resulted in enhanced ease and efficiency, it has however, exposed mission critical resources of an organization, government or individual to malicious activities. Therefore, a combination of security tools and policies are enforced to detect and often, prevent cyber-attacks. This chapter explains and evaluates in detail the existing Intrusion Detection Systems (IDS) which are either based on host or network data respectively. Further, different approaches for detecting the cyber threats are also discussed at length. Finally, a brief discussion on the application of anomaly detection in network based intrusion detection approaches is included.

1. Intrusion Detection System (IDS) - a brief overview

As the threat landscape is evolving, the cyber threat detection is also advancing concurrently. These threat detection systems are often termed as Intrusion Detection Systems (IDS) and their primary purpose is to detect any vulnerability, exploit or anomalous activity in a system. As these systems are only geared towards threat detection, they were initially positioned out-of-band of the network infrastructure. It implies that instead of being real time in processing internet data, these were kept offline and processed a copy of the original data to find an anomaly. Therefore, these systems were also considered a listen-only device and could not prevent any damage from happening online and rely on human expertise to further control and mitigate the damage [5].

IDS in general collect information about the host and network resources of a system, and attempts to analyze it in order to detect any probable intrusion in the system. It provides a mechanism for security specialists in an organizations to monitor activities across the computer systems and network infrastructure. In addition, it helps in assessing and managing the current situation of the network across all the connected systems by raising flags in case an anomaly is detected. Sometimes, these systems are connected to the more sophisticated control mechanisms which automatically get activated on receiving an alert.

Initially, most of the cyber-defence strategies targeted the evaluation and analysis of network or host based resources separately to detect malicious activities. However, with the evolution in the design and sophistication of cyber threats lately, the concept of IDS has also expanded to include the monitoring and analysis of host and network features together. The simple concept behind this convergence is to assess the interconnection of all features that may act as the indicator of compromise for the whole system. This combined analysis strategy may reveal information which can lead to the detection of cyber-attack.

2. Types of Intrusion Detection Systems (IDS)

In general, the Intrusion Detection Systems (IDS) can be classified in two different categories namely:

- Host based Intrusion Detection System (HIDS)
- Network based Intrusion Detection System (NIDS)

As illustrated in Figure 1, this categorization is based on the function and topology of the Intrusion Detection System and is discussed in detail in the next sub-sections. However, for the scope of this thesis work, only the network based intrusion detection is being considered.

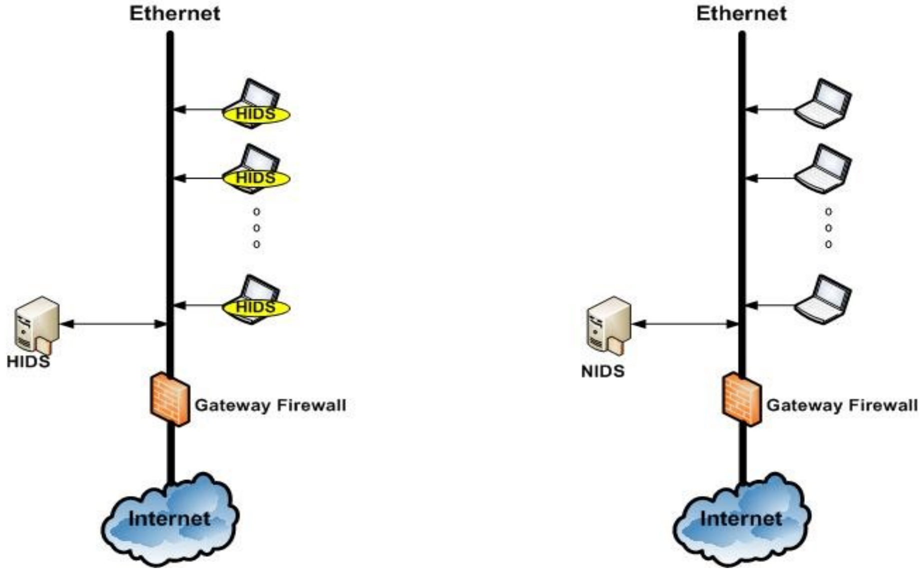


Figure 1: Host versus Network Intrusion Detection System (IDS).

2.1 Host Intrusion Detection System (HIDS)

Host based Intrusion detection systems work in a client-server setting. Basically, each computer node has an IDS client installed on it which sends regular updates to the IDS server. The IDS client is often termed as an agent and collects information about different activities and events taking place in a computer node [6]. This information can be extracted through the audit trails which is an operating system's mechanism to record events. Other sources may include system logs and third party software which can extract process tree, threads and module information. Sometimes, special purpose software are used to read memory foot prints to gather deep kernel level information to find traces of advanced malware attacks which gains unauthorized access to an operating system kernel.

An important consideration in HIDS design is that often requisite data is not available in the standard logging mechanism provided by the operating system. Therefore, kernel level modification in the operating system is performed or specialized software libraries are written to extract the events details. However, the major drawback in such approaches is the associated resource utilization cost which can be impractical in real-time detection of threats.

HIDS continuously monitors the changes in the system like file system, system calls, registry, network events, threads information, memory resources, processes and modules details. For example, if a process is initiating multiple system calls frequently then the associated file system and the related modules information may reveal an underlying attack pattern. Also, this malicious process may spawn multiple child processes to carry out the malicious activities and pretends characteristics of a benign process. This operating system related information can be correlated further to detect an intrusion attempt. Nevertheless, the performance of the HIDS depend on the system and network level data and information that can be collected from a particular node.

One of the biggest challenges with the HIDS is the storage and processing of information. In other words, in order for the HIDS to perform efficiently and effectively, the designers must address the challenges of ever-increasing volume, velocity and data complexity which increase with the increase in computing nodes in a network. Otherwise, a threat remain hidden inside a large pile of data samples and could become a safe haven for threats for a long time. In addition, the performance of host based systems is susceptible to any existing vulnerability in the operating system. It implies that exploitation of such an inherent weakness by a threat-actor may lead to successful attack without being detected by the HIDS.

Moreover, portability of the detection software is another problem which implies that the platform compatible agents are required to be installed on each node. This may become an issue with a large organization where different teams may use various operating systems. For example, research and development team may wish to use Linux based operating systems and human resource team would opt for MS Windows operating system which necessitates the requirement of a global HIDS that should be compatible with existing operating systems.

Notwithstanding the above mentioned limitations, host based systems are desirable because they keep track of each user’s activity and hence, can immediately point any illicit event occurring on a particular node. It is also helpful from the perspective of threat control or prevention as attacks can be stopped from spreading further towards other nodes by removing the affected node from the network.

Another important characteristic of HIDS is their ability to detect encrypted attacks. This is because an encrypted malware has to be decrypted at the host in order to be executed and hence, it can be detected at the computer host. Nevertheless, this is not possible with the NIDS which often deals with the encrypted traffic (packets/flows). In addition, host based intrusion detection systems are inherently distributive and therefore, are scalable (Figure 1). This is beneficial in the circumstances where the volume of the network traffic increases to a capacity where it becomes difficult to be monitored and hence, the network traffic of individual node or a group of nodes can be observed and analyzed.

2.2 Network Intrusion Detection System (NIDS)

Instead of monitoring individual node activities, network based IDS gathers information directly from the network traffic stream which is performed often through tapping. It implies that these devices get a copy of the data that flows through an organization's network gateway. Generally, the inspection mechanism utilized by NIDS involves checking the contents of the packet header of the inbound or outbound network traffic to detect any illicit activity. There are different variations of NIDS which are currently available, some of which involve inspecting only the IP packets while others may monitor application level protocols or both.

As illustrated in Figure 1, NIDS are deployed at the network gateway/backbone so that every packet is required to pass through it. This particular positioning of the device gives it an edge over HIDS in terms of cost efficiency as it does not use the resources of individual nodes or hosts. In addition, currently available state of the art network monitoring is performed at the line-rate and therefore, it is difficult to be tampered by the threat actors. Further, contrary to the host based IDS, network based IDS is not only portable but also operating system independent. It implies that it can detect different types of attacks targeted towards distinct operating systems using the same network attributes or signatures. This feature becomes handy in large organizations where network topology can vary with respect to the departments or teams.

One of the major challenges with the network intrusion detection systems is the scalability. Since, these devices inspect every packet, they may affect the line-rate or otherwise perform their analysis offline which make them unsuitable for real-time cyber threat detection. Moreover, with the ubiquitous role of internet in human's daily lives, the network traffic is growing at a tremendous rate rendering many gigabit of data per second. Thus, evaluation of each packet become cumbersome while keeping the data rate intact. Encryption is another critical limitation of the NIDS. With the advancement in the cyber-attack design, the content of the packets are encrypted and therefore, are not readable by the network security devices. Hence, the need for host based security arises which has the access to the exact packet content.

NIDS are broadly based on two strategies [7] [8] [9]:

1) Signature based detection

2) Anomaly based detection

2.2.1 Signature based network intrusion detection

One form of the NIDS utilize signatures to detect malicious activities. These devices keep a database of network packet based signatures which include but not limited to header patterns, packet body content and packet exchange sequence of known intrusions. Sometimes, these records may have particular profiles in place to detect intrusion. These profiles are called rules and depend on human analyst or an advanced user to update them with respect to the latest cyber-attack signatures. Every packet arriving in the network is then compared with the available record of the malicious activity. If a match is found, an alert is generated which may further triggers a follow-up action e.g. blocking the traffic completely or rate limiting it.

Although, signature based NIDS are fairly accurate in detecting already known cyber threats, they are futile in the face of advanced cyber-attacks which morph and can change their behavior in merely 15 seconds [10], and thus apparently become new attacks. Fundamentally, signature based detection systems are reactive in their approach as they depend on the knowledge of the previous attacks only and have no evolutionary learning mechanism involved in their detection approach. In short, these systems will always be a step behind the threat actors in spite of the regular update of the signature database.

2.2.2 Anomaly based network intrusion detection

The other type of NIDS which are proactive in their decision mechanisms are based on determining anomalies in the network. It entails that these devices look for changes in the regular network traffic patterns and generate an alert in case a deviation is observed. However, anomalies can be both malicious and normal and the legitimacy of the anomaly needs to be evaluated. In other words, the change in the behavior of the network traffic can be attributed towards an intrusion attempt or it could be due to a legitimate change in network activity. This gives rise to the false detection rates which are often observed in anomaly based NIDS. Nevertheless, the ability of such devices to detect unknown cyber-attacks is making them useful in today's threat landscape.

Various categories of anomaly detection algorithms are available in the literature, however, they can be broadly classified into the following three categories [11] [12]:

1. Machine learning based NIDS – involves classifying the anomalous instances using supervised or unsupervised machine learning algorithms using class boundaries to distinguish the normal and anomalous samples.
2. Graph based NIDS – this utilizes the concept of high dimensional graphs to represent data samples and estimate the class of the instance using a distance metric.
3. Heuristics based NIDS - this employs statistical analysis to estimate network traffic behavior and utilize statistical thresholds for the definition of anomalous and normal data flows.

All the above mentioned anomaly detection techniques have their fair share of limitations. The statistical assumptions used in heuristics based NIDS often fail to comply for the continuous data stream that has varying statistical properties and hence, has low detection accuracy. Further, they require the data streams to be statistically stationary which is close to impossible to achieve for continuous dataset [13] [14] [15]. Likewise, the performance of the graph based algorithms is limited by the dimensionality of the dataset which grows with the increasing number of input features and simultaneously causes error rate to increase. Relatively, machine learning algorithms are providing better results compared to the previously mentioned approaches. However, the challenge of high false alarm rates must be addressed when employing them in real-world settings because determination of a true attack becomes equivalent to the problem of finding a needle in a haystack [3]. Moreover, these technique detect anomalies based on their deviation from the known normal traffic pattern. However, the advanced threats are able to morph their behaviors [16] [17] [18] to that of the legitimate network flow and remain stealthy and thus, go undetected using contemporary machine learning techniques.

This research work addresses the problem of class overlap that emerges on a feature space formed by the internet flow dataset having advanced cyber threats. This is due to the mutation of the cyber-attacks which mimic the behavior of benign traffic as decoy and are resilient towards state-of-the-art machine learning based cyber defences. Therefore, scale based cognitive complexity measures need to be introduced in the current machine learning algorithms to detect anomalies in the network traffic.

3. Current literature on anomaly based network intrusion detection in cyber-security

Machine learning algorithms have found their way in the detection of cyber threats using datasets consisting of network flows. The first step in this process is the feature selection which is crucial for better classification performance [17]. The University of Twente researchers proposed a list of features including but not limited to internet traffic flow count, number of bytes and number of packets for the detection of malicious activity in a time series [19]. The study revealed that attacks have varying features and require different metrics to be used for their detection. This implies that instead of few selected attributes, a collection of metrics are needed to classify various types of attacks [19]. Another interesting approach for feature extraction in network IDS has been proposed in [20] where optimal mixture coefficients of features determine the selection criteria. It is based on the support vector data description and can be expressed in terms of semi-infinite linear set of equations. The idea was tested on HTTP dataset and satisfactory results were obtained. Further, feature selection and the actual classification problem has also been explored using a combination of data mining and machine learning techniques. As discussed in [21] the data mining techniques have been employed to find a suitable feature set. However, classification task is performed using a decision tree which demonstrated low false positive rate. Also, a combination of two machine learning algorithms like genetic algorithm and support vector machines have been used to extract feature vector and packet classification respectively [22].

Similarly, clustering mechanisms have been used in the attack detection and prevention mechanisms. For example, in [23] authors described a Gaussian clustering technique to identify network traffic patterns. Moreover, attack characterization strategy has been proposed as well in the same paper using Hidden Markov Model (HMM). Berezinski et al. [24] have explored the entropy based measurement for the anomaly detection in internet flows. The study suggests that Tsallis and Renyi entropies provide valuable results using internet addresses, ports and flow duration as features. Authors in [25], depicted similar results using two different feature distributions consisting of flow headers like IP addresses, ports, packet size and the other being the behavior of the internet flows. Further, an elaborative discussion on different machine learning based NIDS is available in [26].

In addition, single layer neural network with backpropagation algorithms has been used for the anomaly detection in cyber-world as discussed in [27]. In this paper, the authors briefly analyzed the limitations of signature based detection methods and then, presented the concept of machine learning for malicious activity classification based on its conceptual similarity with the fingerprint detection system. The DARPA intrusion detection data set has been extensively utilized to test the performance of the neural network as reported in [28] [29] [30]. The aforementioned papers discuss different techniques for the reduction of false positive rate while improving the classification performance keeping reliability of the results intact. Moreover, when various backpropagation algorithms were compared in terms of their classification performance in [31], the conjugate gradient based neural network outperforms others. In another study [32], web spam samples were classified using three different learning algorithms i.e. conjugate gradient, resilient backpropagation, and Levenberg-Marquardt. It was found that the resilient backpropagation algorithms were the best from the perspective of classification accuracy and speed. Nevertheless, generalization of the results for the network data remains a challenge to date which is being addressed using multilayer or deep neural network as discussed in [33] [34]. However, it is not addressed within the scope of this research thesis.

In this thesis, the importance of multiscale analysis in machine learning for cyber threat detection is highlighted particularly the modification of classical single scale Hebbian learning rule to multiscale analysis. A supervised neural network based on Hebbian rule to detect malicious pattern is discussed in [35]. This network grows dynamically while classifying the malicious and normal patterns. Authors in [36], elaborated Hebbian rule based internet anomaly detection system which resides in an independent processing core while inspecting every internet packet captured through Libpcap library. A distributed version of the Hebbian learning rule based neural network has been explained in [37]. In this work, the dataset is basically divided into non-overlapping sets, each of which is then trained individually through a neural network in a parallel fashion. KDD99 dataset was used to validate the proposed methodology and promising results were reported.

Today, a combination of multiscale tools and machine intelligence algorithms have been of interest among cyber-security researchers. The complexity measures based on the multiscale analysis of the dataset are being used to improve threat classification performance. Authors in [38] exploits the self-similarity dimension of the clusters for unsupervised learning. Basically, box-counting dimension was used to cluster datasets having irregular shape and the points are assigned such that there is minimum change in the fractal dimension of the cluster. Further, the multifractals theory was also used to obtain a feature vector for the classification purpose in [39]. The underlying technique is the multiplicative binomial cascades which resulted in classification performance of over 90%. In addition, the Hurst parameter which is another measure of self-similarity was employed to detect anomalies in LAN traffic [40]. The core idea was to estimate the Hurst parameter for the network traffic attributes and compare it with the standard pre-defined values. Contrary to the variance based estimation, this approach showed better results. Literature [41] also discusses combined fractal and wavelet approaches for the analysis of network time series. It is based on decomposing the signal, a time-series representing the network statistics in this case, using discrete stationary wavelet transform. Then the fractal dimension of the obtained signal is determined using sliding window concept which is further used for anomaly detection. Besides, fractal based neural networks are relatively unexplored and the available research [42] [43] [44] [45] elaborates on the theory and results of the re-organizing neural networks which adopts a fractal pattern in their growth. This implies that the increase or decrease in the neuronal connections form a fractal structure. The primary advantage of this neural network is its tremendous ability to learn new patterns compared to the traditional structure resulting in better performance. Contrary to these neural network structure based fractal methods, this thesis, in particular, deals with the inclusion of multiscale tools like fractals and wavelets in the existing machine learning functions to detect network based anomaly which is a relatively unexplored area of research.

Chapter III: Cyber threat landscape and cyber kill chain

In today's world, cyber-security is being considered as a global challenge since the expansive and rapid growth of cyber threats has profoundly affected everyone from ordinary people to local governments and world-organizations. The diversity in the attack methods and targets along with the dynamism in the behavior of the state-of-the-art cyber threats has caused the threat-landscape to be multi-faceted. One of the important methods used for the modelling and analysis of the threat domain is the utilization of the concept of kill chain. It aids in approximating the sequential behavior of the threat process, from reconnaissance stage to the action state, which can be utilized to detect and further prevent the cyber-attack. However, the ever changing threat landscape requires modification in the standard cyber kill chain as well. This is required in order to accommodate for the complex nature of new cyber threats. This chapter initiates with an in-depth description of current cyber threat landscape which further leads to the discussion about the standard cyber kill chain model and how the two important concepts can be linked together to strengthen defences against cyber-attacks.

1. Cyber threat landscape

Internet based devices such as computer, mobiles, tablets, medical devices, smart home devices, embedded devices, vehicles, and smart TVs have become prevalent. From healthcare to economy and education to politics, every field is not only connected to the cyber space but as well, dependent on the internet for its successful operation. On one hand, this shift towards cyber based world has enabled widespread dissemination of information, easier access to facilities, single-click connection with friends and family, closer monitoring of economic changes, and even running successful political campaigns. However, this has posed an unprecedented challenge of securing critical infrastructure, processes, and information while additionally taking steps to prevent any malicious actor from attacking the cyber-space leaving the entire world crippled.

The domain of cyber threats is a fusion of simple and complex techniques and approaches which has helped adversaries to launch successful attacks. In other words, there are two fundamental principles shaping the threat landscape. The first is the usage of immature and relatively simple technology to ensure covertness of the attack amidst the normal internet traffic. The other one is the utilization of cutting-edge techniques to ensure reliability and robustness in the attack. Together, the two techniques create a final cyber-espionage which often goes undetectable for years causing unimaginable havoc.

1.1 Cyber threats - a statistical glance

Due to diminishing division between the real life and virtual life also known as online life, cyber threats have become integral part of our daily lives. From the statistical point of view, a zero-day vulnerability was found almost every week in the year 2015 alone. Moreover, nine different massive breaches with each compromising more than 10 million records were reported within a year. Further, an average of more than a million internet users became the target of web-attacks on a daily basis. From organizations' perspective, a large company or industry faced more than 3 successful attacks on the average. The ransomware which has become a common tool for threat actors to monetize their malign intent through locking victim's computer system has increased by 35% between years 2015 and 2016 [46] .

It is important to note that since last few years, malware is the most damaging category of cyber threats from the perspective of sensitive data theft, compromise of the credentials, reputation harm and financial loss. It has outclassed all other threats in terms of causing loss thus, grabbing the top position in the cyber threats list with an addition of one million malware samples per day [47]. Also, it is reported in [48] that a known malware is downloaded every 81 seconds while an unknown malware is downloaded every 4 seconds. In addition, web-based attacks which are often based on the web-server vulnerability have almost doubled in 2015. Further, same year approximately 58 thousand new malicious URLs were found on a daily basis. In addition, on average a host machine communicates with a malicious website every 5 seconds [48]. The major reason is that the website owners do not patch their websites and webservers and thus, leaving a window open for the adversaries to exploit the vulnerability [46]. Moreover, cyber-security researchers have noted a manifold increase in Denial of Service (DoS) attacks such that they reached bandwidth of 100 Gbps which was assumed to be unattainable before. These sophisticated attacks have taken down major DNS providers in the past causing tech-giants like Twitter to become unavailable [49]. In addition, there is one new DoS attack observed every 20 minutes [50]. Also, exploit kits which aid adversaries in successfully carrying out their malevolent plan have increased by almost 67% [47]. These kits are often the basic cause of zero-day attacks. One such kit is named as Angler and in 2015 researchers from Symantec alone observed over 19.5 million attacks which were based on this kit [46]. Further, social engineering based exploitation of people to gain access to their private information like credentials, photos and medical history as well surfaced in the past few years. For example, one of the scams based on social engineering tactics was able to access Gmail accounts of the users thus, bypassing the two-factor authentication. The greatest challenge for any organization in terms of security and privacy is to detect insider-threat which corresponds to 60% of all cyber-attacks. Sometimes these attacks are launched by a direct adversary while at other times an inadvertent actor is exploited to achieve malign targets. In general, a significant increase of 64% was reported in security related incidents and thus, staying ahead of the adversaries is the most difficult challenge for cyber defenders.

These days the privacy and security of individuals and organization is anything but private. Threat actors have varying motives for attacking targets which include but is not limited to financial loss, physical harm, reputation damage, sensitive information compromise, surveillance and control. Determining the motivation of attack is one of the ways to find the impact of any malicious activity. However, the cost to recover from the loss which includes responding to the threats, investing in new tool and methods to increase the security and creating backup plans to combat another such attack is used as a measurement to determine the threat influence [51]. In terms of numbers, from year 2013 to 2015, there was a 400% increase in the business related data loss alone. Also, the approximated annual cost of cyber threats which are observed globally has gone over 100 billion US dollars. A single variant of deadly Zeus malware known as Gameover Zeus caused a loss of more than 100 million US dollars [52]. It has been estimated that the average time to respond back to the cyber-attack has gone up from a 24-day period to 32 days with a 55% increase in the daily recovery cost which is now approximately 32 thousand dollars on average [50]. Further, these attacks are targeted not only towards financials organizations or governments but, to the technology, healthcare, education, manufacturing, retail, insurance, media, transportation, hospitality and communication sectors to name a few [51] [53]. It is of critical importance to note that the healthcare was the most attacked industry in past year [46]. It grabbed one of the top three positions in 7 large breaches in 2015 alone [54]. As reported in [55], in 2017 threat actors are expected to cause damage to several nations by targeting their physical systems like critical infrastructure and consumer devices. This will cause disruption in government's control and operation which can be misused for political gains. Therefore, it can be concluded through the statistics that cyber-attacks are pervasive reality of the cyber realm.

1.2 Cyber threats - a brief discussion

Some of the significant types of the latest cyber threats and incidents are briefly discussed below:

1.2.1 Malware

Malware is a generic term which is used to represent a piece of malicious software which aims to interrupt the intended behavior of a computer system or device. It includes various different forms of viruses, trojans, worms, spyware, backdoor, keylogger, and ransomware to name a few. The sophisticated malware are capable of re-programming firmware of the target system in order to provide full control to the adversary. Also, with the rapid increase of mobile devices, embedded malicious code is finding its ways to attack mobile devices to harm the end user in terms of stealing sensitive data, and causing financial loss to name a few. Usually, the malicious code piggy backs on the application repositories (app stores) for its packaging and distribution.

1.2.2 Botnets

An important type of cyber-attacks which rely on the command and control server for its operation and execution is botnet. Often, hundreds or thousands of otherwise benign nodes are infected such that they may act as an adversary on behalf of threat-actor to achieve their evil goals. The technology behind these attacks have improved significantly from the usage of encryption and exploitation of protocols to abusing algorithms for evading detection. This is the first type of attack that matured as a service in the realm of cyber-crime. Also, this is being used in a number of ways for monetary benefits. Some of these include selling of the bot kits, downloading of the malware on the victim's machine, locking sensitive files to gain money, and fixing the available bot kits for immature hackers by taking money. Recently, bots have also been used for carrying out Denial of Service attacks as the involved nodes are used to amplify the attack bandwidth and impact [56].

1.2.3 Web based attacks

Web based attacks exploit the vulnerabilities of a web-server and a web-client to download and install a malicious program into the victim's computer. A number of strategies are utilized to achieve the malign targets of the adversaries. Some of these include exploiting the browser vulnerabilities, using malicious URLs, re-directing to compromised webpages, and drive-by download attacks. Another similar attack is based on web applications which has become very popular with the usage of mobile devices. Threat actors often manipulate Software Development Kits (SDK) to inject malwares. They try to exploit the inherent vulnerabilities in the software and web-stores to evade detection and steal information. Moreover, malvertising is also observed to be increasing during past few years. This type of attacks rely on the browser plugin to bundle the unwanted software in the package. The most exploited website to launch web-attacks is technology [57].

1.2.4 Denial of Service (DoS) attacks

This category of attack consists of malicious activity which results in a denial of internet service to a legitimate user. This is carried out by exhausting the target's resources. Commonly, these attacks are targetted against webservers where multiple clients or nodes take part in augmenting the attack impact and thus, is often termed as Distributed Denial of Service (DDoS). Recent trends in DDoS attacks are focused on enhancing the underlying attack tactics and technology. One such method is to abandon the centralized single server based attack and exploit pervasive Internet of Things (IoT) devices based attacks. Often, attackers demand money from the victim by black mailing them. The targets are threatened to pay or have their resources under attack. In general, these attacks are targeted towards internet providers, software industry, gaming and financial sectors [47]. It has also been noted that the massive DDoS attacks are often initiated by state actors [46]. Moreover, two common approaches are used in these attacks. Either the attack is directed towards single component of the target infrastructure or it may be directed towards multiple components of target infrastructure. In the former case, it could be short, less harmful attack however, if multiple components are being targeted the attack itself is expected to be slow and long but deadly in terms of both critical access loss and difficulty in detection due to threat actor's anonymity.

1.2.5 Spam and phishing attacks

Spam and phishing attacks serve as a backbone for a number of advance cyber threats. Commonly, they are used for malware distribution and credential harvesting. Both of these attacks exploit the trust of the end user and lure them to believe malevolent webpage, email, messages and organisation as trust-worthy. Usually these attacks are launched in the form of campaign which achieve highest success in the first few days. In the past few years, spam and phishing campaigns are excessively used to abuse international events like earthquake in Nepal, 2016 Olympic Games in Brazil, United States elections and Google's algorithm update [58].

1.2.6 Insider threat

The biggest security risk an organization faces is from an insider which could be a direct player in the malicious activity or it can be someone who is an inadvertent actor in the bigger picture. It implies that insider threats include all unintentional security incidents as well. However, irrespective of the fact that the insider is naïve or cunning, the opportunities it can provide to exploit a system vulnerability is massive and impactful. In a typical organization, it may include employees, corporate-partners, vendors, customers, and share-holders. These people have access to the sensitive information depending on their role and level in the organization. Some of the basic loop-holes which are exploited by the threat-actors are lack of policies and care to implement security, non-serious attitude of insiders towards security, stressful workload that leads to ignorance of security, grudging employees and inconvenience in abiding the security related policies. The insider attacks are not only targeted towards causing financial loss but, damaging the reputation as well [59].

1.2.7 Exploit kits

These are automated software programs which scan for an available computer vulnerability and are responsible to select appropriate malware accordingly. It then ensures successful delivery and execution of the malware using malicious URLs, droppers, and command and control servers. The realization of these exploits require interconnection between malicious tools, attack-actors and infrastructure which has not only increased sophistication of the attack but, has also created a network of threat-actors. Moreover, these kits are capable of obfuscating by encryption and morphing in addition to the advance techniques which are used to evade signature detection. The adaptability and impact of these programs is far-reaching [47].

1.2.8 Targeted attacks

This is a form of cyber-espionage often backed by state actors which devise smart threats that can go undetected for years even with the usage of state-of-the-art detection methods. They are often known as APT (Advanced Persistent Threat) which refers to them being; (i) *Advanced* implying the usage of innovative and sophisticated technology to exploit vulnerabilities in the system, (ii) *Persistent* for its ability to evade detection for a long time while being connected to external command and control communication setups, (iii) *Threat* refers to the financial loss, reputation damage, critical information theft to name a few which the victim has to bear [60] [61].

1.3 Cyber threats mitigation strategies – a review

Some of the strategies that can be followed to promptly detect and proactively prevent cyber threats from creating disasters are as follows:

1) The regular patching of firmware and software is critically important. This includes patching of vulnerabilities in browsers to automatically block installation of Potentially Unwanted Programs (PUPs) and modifying the default settings to tailor it for more secure usage which otherwise may allow malicious programs to gain access to the users' device.
2) A combination of host and network based anomaly detection methods should be implemented which may involve creating blacklists and whitelists for software applications, network traffic, and IP-address through filtering of files, programs, email attachments and web-contents.
3) Multiple blocking mechanisms should be enforced as well to make sure that these harmful pieces of code are not able to victimize user due to limited capability of one blocking device. For example, automatic execution of code macros and rendering of graphics should be disabled.
4) Also, continuous monitoring of inbound and outbound network traffic not only at the gateway but, at the host machine is necessary to detect obfuscated attack patterns.
5) Further, it is also imperative to devise strategies for systematic updates of the cyber threats detection systems. This includes training of the Incidence Response (IR) and Threat Intelligence (TI) teams to plan-ahead the strategies in case of zero-day attacks [47].
6) In case of DoS attacks, a plan should be at hand to combat the threat at multiple stages. The efforts should be focused towards stopping the attack close to its source.
7) There is a need to increase the awareness about the lethal consequences of naïve/inadvertent user behavior especially in the context of phishing and spam attacks.
8) A standardized security policy should be drafted and implemented in an organization to minimize the risk of insider threats. For example, access to sensitive information should be

provided to a person based on their role and trust in an organization. Also, a logging mechanism should be in place to track every activity of the users accessing a resource.

9) To protect the mobile devices, a firewall should be installed to inhibit installation of malicious applications.

10) Moreover, a concise and collective framework needs to be established to standardize the mobile application development in order to protect the end users from malign software.

These are some of the critical steps which must be taken to ensure safety and privacy of the cyber-space. However, due to the complex nature of the cyber-realm, these measures are not sufficient enough to guarantee detection and protection of internet based data and services. Therefore, in this thesis, the author has presented the idea of incorporating cognitive analysis in machine learning algorithms. Further details in this regard are available in Chapter IV.

2. Cyber kill chain

The detection of these highly sophisticated attacks requires modelling of the entire attack-phenomenon in order to understand the motives of the threat-actor which helps in updating security measures accordingly. This notion is based on the military concept of *kill chain* which refers to the organization of a military attack and was introduced by Lockheed Martin in 2011. This model defines cyber-attacks in terms of unique sequential and progressing stages, and based on that proposes a smart framework of solution for the analysis, detection, and prevention of cyber threats [60]. The basic assumption in this model is that a threat can be stopped by obtruding it at any of the kill chain stages. However, the efforts of cyber security experts for example, Incidence Response (IR), Threat Intelligence (TI) and Security Operation Centers (SOC) team is to catch the threat closer to the origin by utilizing kill chain model as a guide. In this way, preventive measure can also be taken at every step [62].

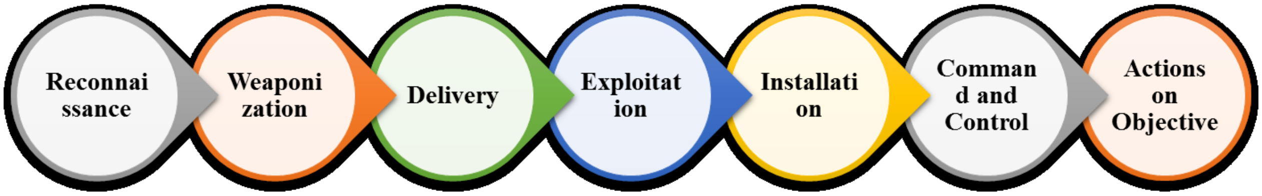


Figure 2: Cyber kill chain model [60].

2.1 Cyber kill chain stages – a brief overview

There are seven stages of a standard cyber kill chain model as illustrated in Figure 2 and are discussed below.

2.1.1 Reconnaissance

Similar to a military style attack where the first step is to gather information about the enemy, this stage in cyber-attack scenario requires surveillance to collect information about potential vulnerabilities, identification of target, and decision about possible attack methods [60]. For example, vulnerabilities which can help create a backdoor in the system are identified at this stage. It also includes investigating the best attack methodology to evade detection by the cyber defence mechanisms [63]. It is generally at this stage that the adversary finalizes the objective to infiltrate the target which is critical in deciding about the required communication channel which must be established during the attack and hence, is also a decisive factor regarding the choice of attack tool. For example, a botnet based attack requires malicious client to communicate back to the command and control server and therefore, can be used for stealing sensitive data and information. On the other hand, the phishing attack which is generally used to drop malicious programs like viruses and worms to the victim's computer is used to completely destroy the victim's resources. Hence, this step is fundamental in deciding the success of an attack which is based on the thorough analysis conducted by the attacker in order to illegally intrude the target system.

2.1.2 Weaponization

Based on the data and information gathered at the reconnaissance stage, the adversary analyzes the best method to launch an attack such that it is able to evade the malicious activity detection systems installed at the victim's node. In other words, this stage is used to prepare ammunition required to destroy enemy without getting detected. In terms of cyber-war, the threat actor choses the weapon through which the pre-set objective can be achieved utilizing the vulnerabilities and covert channels in the target system. It implies that this step is critical to determine *which* attack tool should be used and *how* to ensure its successful delivery at the target setup. This preparation also involves creating a backdoor and a penetration plan [64]. Generally, the victim remains unaware at this stage of the attacker's activity and motives. However, it could be caught at the previous stage of reconnaissance when the attacker was trying to determine a vulnerability in the system. For example, port scan is a way to find open ports in a victim's system which can be abused by the attacker. This activity can be captured with careful investigation on target side.

2.1.3 Delivery

Once the complete attack strategy is devised, the actual malicious payload or software is delivered to the victim's side. It is important to understand that at this stage the malignant piece of code should not be detected by the cyber defenses. To ensure this, either the delivery traces should be removed from the system or if it is not possible then it must not reveal the identity of the threat-master at all. In addition, multiple attempts through different routes are taken by the adversary to guarantee successful delivery without detection. For example, the malign program can be; embedded into an attachment [65], delivered through a backdoor [66], connected to a malicious URL [67], deployed by exploiting browser vulnerability [68], or distributed by a disgruntled employee (an insider) [69]. This means malware can be delivered to the target either through the user interaction or without it. The evasion of the delivery step from the defence radars is made possible through the usage of encryption [70] and behavioral or code based morphing [71]. This is achieved with the usage of programs like droppers [72] and downloaders [73].

2.1.4 Exploitation

Once the malicious program is delivered successfully to the target system, a software environment needs to be prepared to ensure its smooth installation. This implies that the delivered malware should have the access rights required to read, modify and write the required resource. Moreover, it should be able to deactivate any cyber threat defence system installed on the target system for example antivirus software. In addition, all the necessary files and libraries needed for full installation and execution of malicious software should be readily available on the system. In order to fulfill these requirement, a software or hardware bug also known as CVE (Common Vulnerabilities and Exposures) should be present in the system that can exploited to cause further damage. On a side note, a public database of the latest CVEs is available at [74]. It is used by the software developers to provide regular patches to secure their programs. However, there could be vulnerabilities which may not be listed and can be exploited by cyber-attacker for instance zero-day threat.

2.1.5 Installation

This stage is used to install the malicious piece of program on the victim's system and initializes infection. If the malicious program is in the form of an executable file, code injection or insider threat, this stage may not be needed. However, in other situation where installation is deliberately required, this step updates the resource accessing mechanism of the operating system. It also morphs itself in a form that cannot be detected using sophisticated anomaly detection mechanisms. This implies altering memory foot print, modifying process trees, and changing operational or execution behavior. The objective is to mimic the pattern of a legitimate file, process, and network trace.

2.1.6 Command and Control

When a cyber-attack is targeted towards stealing sensitive information for example, user private information, critical government documents, intellectual property, and health records, a command and control server is required to steal information by sending command and collecting data from victim. It can be a single web-server, peer-to-peer network or social media server. Moreover, command and control is also required in multi-level attacks directed towards different parts of a system and thus, requires an adversary to be available as a regulatory entity. For example, botnet based attacks have a command and control channel established with the server which could be a standard LAMP based web-server or sophisticated peer-to-peer protocol based server cloud.

2.1.7 Actions on objectives

This final stage of the cyber kill chain is responsible of actually launching the targeted goal of the attack. It implies that at this stage, the malicious code would have started executing malign objectives as programmed. So, this is similar to the detonation stage in terms of military kill chain. If the attack utilizes a central command and control, then the malware client would have successfully established a backward connection and process of stealing critical information has already started. Most of the current cyber-defence mechanisms tend to detect attack at this state, if possible.

3. Relationship between cyber threat landscape and cyber kill chain

As described in the previous section, the cyber kill chain process provides a method to model the sequential steps taken by an adversary to infiltrate target system and either cause permanent damage or infiltrate data from it without leaving a trace. This model is important from the perspective of a cyber-attack response team which can focus the attention on a particular event based on the threat found in the system. It implies that more detailed and result oriented schemes can be devised to detect and prevent any kind of attack. For example, port scan and social engineering based exploits can be grouped under reconnaissance stage where the objective is to discover information about the target. It can be stopped by blocking non-standard ports and obtruding phishing and spam attacks based on social engineering methods. However, when the malicious code is delivered to the system, it is required to execute host and network based signature and behavioral analysis to determine any anomalous activity. Moreover, when the malware tries to establish connection with the command and control server, the traces of C&C communication can be helpful in determining any invasion which might have occurred. Also, a detailed memory analysis can be used to find an attack trying to evade standard detection mechanisms. The basic idea is that it is not possible to kill multiple birds with single stone rather, different strategies need to be implemented concurrently to tackle the extremely intricate cyber threat domain.

Moreover, these attacks can be categorized in terms of different execution approaches which eventually decipher the motivation behind the attack and thus, a threat-actor can be traced back. This theme is important in generating detection and prevention measures accordingly. For example, a state-sponsored attack will involve establishing a secure command and control channel through which sensitive information of the target country can be exfiltrated. In addition, each step of the attack process will be extremely sophisticated and will exploit new and unknown vulnerabilities. Moreover, these attack will persist over long term and cannot be detected with simple defence mechanisms. Contrary to this, an attack launched by an individual is usually for some personal gain for instance money fraud. Mostly, these attacks are directed to achieve the malicious objectives as quickly as possible like locking files of a victim computer for monetary benefits and therefore, may not be as stealthy and sophisticated as those backed by organized entities.

In the context of current threat landscape where cyber-attacks are multi-faceted when an adversary is able to successfully infiltrate, the cyber-security teams can still stop the attack. In other words, the attacker will be successful when all the kill chain stages are completed which means that the exfiltration of data is yet to be performed in the described scenario and the malign agents can still be caught to stop the attack. It implies that instead of focusing on prevention only, the cyber-kill chain model helps in predicting the threat, preventing attacks based on it, detecting them and appropriately responding at each stage. This is especially useful for advanced targeted attacks where multiple kill chains are executed in parallel [47]. Thus, it can be concluded that following a cyber-kill chain model can help in analyzing and appropriately detecting threats of varying intensity and scale which are shaping the current cyber threat landscape.

Chapter IV: Cognitive cyber intelligence

The state-of-the-art cyber threat detection platforms utilize either the pre-known attack signatures or heuristics based behavioral analysis to determine the presence of any malicious activity. However, the detection efficiency and performance of these cutting-edge systems is thwarted due to the transmutation property of the latest cyber threats which further add to the already intricate nature of cyber-realm. Therefore, cyber-space defence mechanisms rely on human capabilities to analyze and ascertain the actual malicious incidents from the massive available data. This implies that to enhance the detection reliability, an overhauling in the existing cyber-defence systems is required from the perspective of human intelligence. Therefore, the idea of cognition based analysis in the cyber-security is discussed from two different approaches in the presented chapter. The first is the proposed modification to the standard cyber kill chain described in detail in the previous chapter, while, the second approach is the introduction of multiscale analysis for the cyber threats.

1. Cognitive Informatics and Computing

Literature survey [75] [76] [77] [78] indicates that there are considerable research efforts directed towards understanding of human brain and the mental process of how humans decide, perceive, judge, read, learn, reason and do all such tasks which are known to be cognitive. Cognitive science is a broad field and encompasses psychology, neurosciences, linguistics, intelligence and brain sciences, to name a few [79]. According to the authors of [76], mapping mental cognitive models to machine is broadly classified as cognitive informatics and cognitive computing. Cognitive informatics is a combination of various disciplines and domains including information theory and sciences, computer science and intelligence science that investigates the internal processing mechanism of human brain and intelligence. Cognitive computing is an active research field aimed at applying the combination of knowledge from information and cognitive sciences to engineering applications. This implies investigating various approaches to learn the human mental capabilities and subsequently implementing it to improve artificial learning methods [76]. This involves introducing increased understanding, better awareness and thorough comprehension capabilities in existing engineering concepts and methodologies to devise more reliable and robust solution. It is achieved by imitating tactics and connections which are stimulated in a human brain as a result of an activity. This process requires hybridization of theories and knowledge from multiple domains including but not limited to probability and statistics, signal processing, dynamical systems, machine learning, wavelet analysis, fractals theory, and chaos engineering.

The fundamental goal of research in this domain is to improve the adaptive learning capabilities of the traditional machine learning algorithms. For the presented research thesis, it involves using mathematical tools and models to learn through both experience and evolution and applying them in the context of cyber-security. The experience based learning comes from the pattern analysis of historical events while the evolutionary learning is rooted in adapting to the contextual changes like those found as a result of a stimulus-response mechanism in human brain. It is well established that the humans have an impressive threat sensing and fighting abilities analysis [80], and mimicking them further to refine the existing cyber-defence technology is, hence, the need of time. This research work achieves this notion of cognition through the application of complexity and complexity based measures in threat analysis and detection methods which are discussed in detail in the following sub-sections.

1.1 Complexity and measures

Complexity analysis and measurement is a significant aspect of human cognitive abilities and operations [76]. It is an estimate of the resultant of the interconnections and interaction among the system's components. If there are multiple different synergies in a system and these cannot be further simplified into smaller connections, then such system is considered as complex [81]. Contrary to this, if there are smaller number of interconnected components and relationships which can be analyzed independently, then the system is considered as simple [81]. In a more generalized sense, it can be stated that simplicity refers to the ease in comprehending a system or a process whereas, the difficulty in doing so is considered complexity. An example of a complex system is metamorphic cyber-attack which is capable of altering its behavior and code. These threats often appear to be benign under scrutiny in a sandbox but, morph into a dangerous piece of code when inspection is removed. Such attacks avoid detection by changing their behavior in a system and inside memory, in addition to the modification in malware source code. It is important to understand that these advanced malwares have higher interconnected components and relationships which helps in sensing the environment and adapts their nature accordingly. Even deciphering the source code of these attacks is almost impossible due to the complex modular code-writing approach generally utilized in their development. Comparatively, the first MS-DOS virus written in 1986 which was aimed at replacing the copy of the boot sector file with the virus. This virus was relatively simpler as it did not take into account hard disk partitioning and other critical details. Also, it wasn't able to infect systems where the most significant bit of the BIOS drive was clear. Thus, the cyber-virus and attacks have evolved from simpler to a much complex system. It should be remember that it is impossible to further simplify a complex system without actually damaging its dynamics [82]. For instance, in a ransomware case the adversary may use a phishing email as a tool to achieve their goal of luring at least one user out of millions to click on the malicious link and start the malware delivery process which when executed can encrypt the system. This will enable the threat-actor to demand for money in return of unencrypting the system. The basic motive is to reach a large population to increase the probability of successful attack. Phishing email is one such way of achieving this and it cannot be further simplified. Although, there are alternate methods for malware delivery like insider attack but, the one based on phishing attack cannot be further simplified. Therefore, this is the simplest level of complexity and it is useful from the aspects of tractability and robustness of the system's cognitive ability [83].

A detailed discussion on different aspect of measuring complexity is available in [84] where author has provided a method of classifying objects, systems and processes with respect to the involved complexity in terms of the nature of interaction among system components. It is based on investigating whether (i) the system depicts an order or a definite pattern or, (ii) it lacks any pattern and is completely random. Further, if a pattern involves relatively smaller number of interconnected components or relationships then, it is simple in nature. Contrarily, the pattern which comprises of many interactions with itself and with other patterns leading to an emerging new order, behavior or arrangement is termed as complex [84]. Similar approach is adopted by author in [85] to describe complexity in the context of dynamical systems theory. It is stated that if the collective interactions in a system results in an evolutionary behavior which displays previously unknown patterns then, such system can be considered complex provided that it is not possible to further break it down into smaller components [85]. From cognitive point of view, complexity can be classified as (i) static referring to the limited patterns in the system structure or behavior; (ii) dynamic based on varying temporal behavior; (iii) functional which indicates system's functional components ; (iv) organization which points to the interaction level between various components , and (v) design referring to the underlying structural attributes [86]. Moreover, there are different complexity measures which can be described based on the system model. For example, (i) mono or multi scale; (ii) algorithmic or probabilistic; (iii) local or global; (iv) average or asymptotic; (v) arithmetic or logical; and (vi) absolute or differential.

2. Cognitive cyber kill chain

The latest high-tech cyber-defences employ multiple layers of security to increase the probability of catching and inhibiting an intrusion in systems having critical importance. However, the threat-actors are still able to infiltrate the network and persistently stay undetected in the target system. These systems are generally backed by organized groups or state-actors and tend to cause a damage to the credibility of the victim while successfully exfiltrating the confidential information. For example, the tech-giant Yahoo declared in December 2016 regarding the compromise of its 1 billion user email accounts in 2013 which is the biggest cyber-attack in the history of the internet [87]. This attack was enacted by the same actor who earlier stole the proprietary source code of the company [88]. The news was broken at the time the deal with Verizon Inc. to sell Yahoo at a cost of US$4.8 billion was in progress and perhaps hurt it as well [88]. In the context of the previous example, it can be concluded that there is a need to introduce adaptive strategies for scanning threats along with the layered security approach.

2.1 The limitation of the traditional cyber kill chain

The fundamental problem in the traditional cyber kill chain is the assumption of single and similar intrusion attempt at each stage which is not valid with the sophisticated attacks like Advanced Persistent Threat (APT) [89]. In these attacks multiple kill chains can be formed at a time and each one of them can have numerous indicators of attacks. It implies that it is required to correlate all these information in order to detect the presence of a persistent attack. Basically, an APT uses multiple different schemes and methodologies to enter a network, evade detection, collect valuable information and illicitly send it to the threat-actor. It means that it does not only completely covers all stages of a cyber kill chain but, implements several copies of these chains in parallel and over long time duration to achieve the desired objective [47]. It means that in an organized targeted attack, there are multiple possible permutations of the kill chain which can be in effect simultaneously using lateral movement [90]. For example, an APT attack can start with an initial reconnaissance stage but, instead of preparing for the weaponization, it may find another exploit to gain privilege access like an insider. Then, it may re-perform the reconnaissance internally while moving to the more valuable node. This process can be repeated multiple times such that it masquerades the behavior of a legitimate process or network flow. Finally, it can install and execute the malicious software while establishing communication with the command and control server [90].

2.2 Cognitive intelligence in cyber kill chain

It is vital for the security researchers to consider the cyber-defence strategies from a holistic point of view rather than the conventional single kill chain model. This can be done by taking into consideration the purpose and objective of the attacker. For instance, in case of an attack targeted toward data or information stealing, communication with a command and control server is a must which can be exploited to detect the presence of a threat. Another suggestion by the author in [91] was to focus detection on the internal cyber kill chain stages instead of covering the network perimeter. However, this approach lacked the fundamental network based layered security approach which is imperative to keep the network threats out of the system by stopping them at the network boundary. Similarly, research in [92] proposed the lateral movement strategy in the cyber-kill chain and removed the weaponization stage all together. This scheme considers utilization of multiple nodes (tactics) to access the system's resources. Although realistic, this idea did not take into account the movements of command control communication channels as discussed in [91]. Another intelligent approach was presented in [93] where, authors have argued about a generic model which considers multiple aspects of an attack including but not limited to technical, legal and policy matters. Nevertheless, this discussion was aimed at illustrating the effect of threats from the perspective of shared impact on an organization's stakeholders.

Further, with the exploitation of polymorphism and metamorphism concepts in the cyber threat designs, the concurrent analysis of the threats is inevitable [94]. This simultaneous analysis idea is of paramount importance because of its conceptual resemblance to the methodology adopted by security professionals to analyze a malicious activity. Concurrency in this context refers to reflect on not only the current indicator of compromises but, to investigate in parallel the tactics adversary may have adopted to ensure successful delivery and execution of the malware in the network previously. In addition, the future strategies and the corresponding probable compromises are as well studied. Thus, a collective view of the security measures of the system is investigated and the resultant cyber kill chain model has the following states as shown in Figure 3 [95]:

1. Internal and External reconnaissance to investigate and exploit security weakness.
2. Malicious content delivery.
3. Behavioral changes to ensure persistence.
4. Establishing command and control channel while moving between connected endpoints.

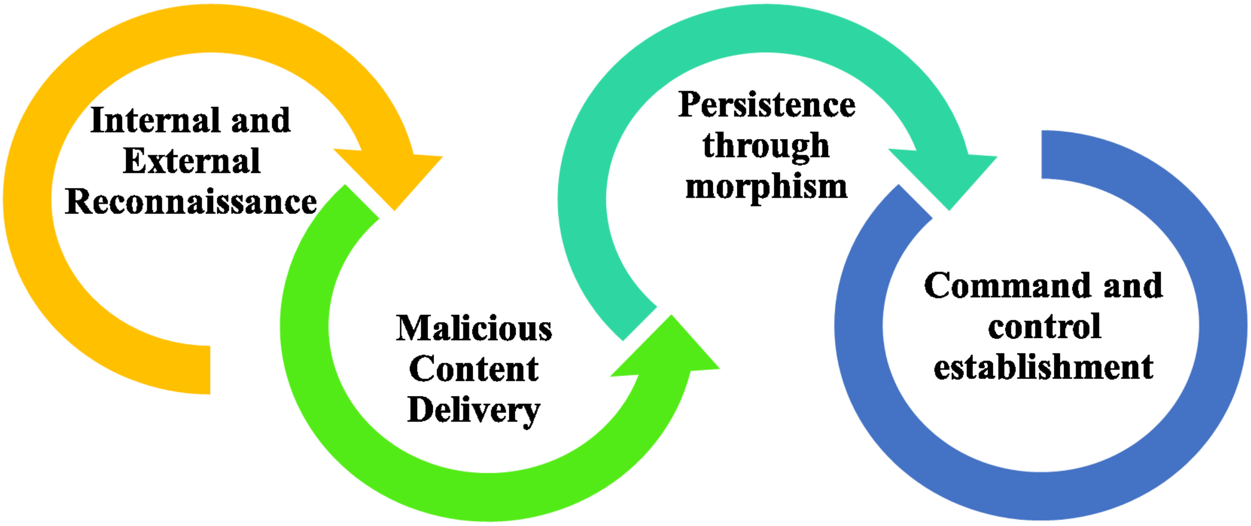

Internal and External Reconnaissance
Malicious Content Delivery
Persistence through morphism
Command and control establishment

Figure 3: Modified cognitive cyber kill chain [95].

3. Significance of cognitive analysis in cyber-security

The recent massive transmutation of cyber threats has challenged the security researchers in an unprecedented manner. They are presented with the daunting task of extracting a reliable and robust set of features which is able to detect multiple different types of attacks which are diverse in their construction, tactics, motives, and targets. So, the problem of selecting n features out of a list of available M attributes where $n \ll M$, implies finding those characteristics of a network or host system which are clearly distinct between malicious and benign samples. In other words, these properties or features help the threat-detection algorithm maximize the classification performance of the system. However, selecting an optimal feature set is intractable [96]. In addition, an optimal solution for the classification problem is only possible when a monotonic criteria for feature evaluation exists [97] which is not the case in most of the real world problems like cyber threat detection. Therefore, selecting appropriate feature set which fully represents the data at hand and provides optimal performance when fed to a machine learning system is intimidating but, critical.

Moreover, these threats are ever-evolving and as a result, a single set of features cannot be attributed towards successful detection of a variety of threats. Research literature suggests that the feature selection problem is NP-hard [98] [99] [100] [101] . Further, if we assume that there exists a set of features which can detect latest cyber threats then, it is merely a matter of few minutes if not seconds that this set will become obsolete in terms of detecting malicious activities. The primary reason is the ability of threat-masters to alter the threat signature rapidly, merely in 15 seconds [10]. Most of the machine learning systems employ the deviation measure between the observed data and the expected or target data. However, with the observed data changing rapidly, it is exceptionally difficult to detect such attacks even with the utilization of cutting-edge defence mechanisms which not only rely on the signatures but, employ the behavioral analysis knowledge. Further, a human expert is also needed to ascertain the anomaly and activate the appropriate policy for the system which is another major limitation of the current cyber-defence technology.

It is worth-considering that with the latest obfuscation techniques the normal and malicious samples from the network and host system look similar. This is because the adversaries attempt to design malware which masquerade the behavior of a legitimate flow or process. For example, about 92% of the attacks are carried out using the HTTP protocol [4] exploiting some web application. However, it is not possible to completely block HTTP otherwise, users will not be able to proceed with their daily browsing activities. Nevertheless, in such scenario the negative and the positive samples will overlap and will become inseparable on the feature space. This requires introduction of scale based analysis discussed in the next section.

4. Scale based cognitive analysis for cyber threats

Scale is defined as the quantifiable aspect of a system, process or object. It represents the fragility or coarseness of various characteristics of the object or system being studied [18]. It is a reference frame that provides the distribution of patterns with respect to the variation in measurement. In other words, it is a type of window or filter for the perception about the observed data by providing a foundational ground to observe and analyze an experiment or event. It implies that increasing the window size will extract details from the coarser level and decreasing it will provide the finer details of the system. Therefore, scale defines the limit or extent to which the information about a system can be observed and further analyzed. For example, a host based cyber-security defence system can provide top level information like process names and the corresponding module names at the coarser scales. Consequently, the details of each module and the threads related information can be extracted at the next refined level. Moreover, different states of these modules and threads can form another level of information. Thus, the underlying relationship between processes, modules, and threads together can provide a holistic and more complex picture of the system. Another example from the cyber threat landscape is of the multiscale attack distribution. First, different types of communications, flows and information can be categorized as normal or anomalous based on the available samples. These anomalous labelled samples of attributes can be further considered as triggered by a host based event or network based activity. Moreover, each of the host and network related information can be separately divided into multiple layers of attributes. Some of the attributes for a network include total duration, number of bytes exchanged, inter-arrival time, and round trip time. Similarly, for the host system these may comprise of process, module, thread, memory and operating system information [102].

The inclusion of scale at the feature extraction and selection stage is critical to evaluate new and emerging structural patterns and procedural relationships. These could be distributed across temporal or spatial scales of measurement. For example, a change in the packet flow duration and the corresponding number of bytes exchanged at a finer level can affect the behavior of the attack at the coarser scale. Generally, more advanced and sophisticated attacks tend to stay slow and low to evade detection [103]. Therefore, finer scales based analysis is a must in such situation otherwise, many critical malicious activities cannot be detected. It is primarily because of the ability of finer scales to reveal convoluted associations in a feature space. It is worthwhile to consider that scale can be used as a modelling tool to reveal hidden patterns in an observation and can also be the characteristic of the process itself [6]. For example, the samples of the attributes like packet inter-arrival time and number of bytes in a flow collected by a cyber defence system can be observed at different scales. On coarser levels, these patterns for normal and anomalous samples may look similar but, as scaled down new patterns can emerge indicating the subtle differences which lead to the identification of an attack. On the other hand, the host computer system information is itself distributed in multiple scale levels; the process names form the first level, the associated module is part of the second level of hierarchy and the thread information is the third level. Each of these levels can be expanded further horizontally depicting the multiscale nature of the system itself. Therefore, it can be concluded that scale based analysis is vital for the detection of highly sophisticated attacks.

4.1 Multiscale analysis using fractals

In order to quantify the complexity of an object or process mathematically, its dimension is determined. Generally, the notion of dimension is limited to the Euclidean space in which the number of independent coordinates required to embed the object represents its dimension. This concept is slightly different from the topological dimension which signifies the form or topology of an object under transformation [84]. To further elucidate, consider a line which has the topological dimension of 1 but its embedding dimension depends on the space in which it is represented. So, in a 2-dimensional space, the embedding dimension of a line will be 2 [86]. It is important to understand that the dimensions associated with platonic objects are all integer dimensions and therefore, lack the ability to comprehend the complexity for a multiscale analysis of an object or a system which requires non-integer dimensions and scales to be considered in evaluation.

Fractals are objects which exhibit their complexity in terms of non-integer dimensions. For instance, the coastline of Britain has a topological dimension of 1 but, a fractal dimension of 1.24. The non-integer number signifies the rough and irregular nature of the coastline contrary to the perceived straight line. Basically, fractal objects are non-differentiable everywhere [104] and have a unique scaling characteristic according to which each portion of an object is a scaled down version of the whole itself; and Britain coastline is one such example. So, the fractal dimension of an object is directly proportional to its complexity [105]. Moreover, fractals are invariant to magnification along symmetrical or asymmetrical scale and therefore, possess the self-similarity or self-affinity property respectively. The theory of Fractals was formally introduced by Mandelbrot [106] who referred to it as a set for which topological dimension is exceeded by the Hausdorff-Besicovitch dimension. Further, Robert L. Devaney [107] introduced the concept of self-similarity of an object in n-dimensional Euclidean space to the Mandelbrot's definition. Thus, self-similarity is the foundational block of the fractal dimension.

To calculate the fractal dimension of an object, the relationship exponent of different measurements like similarity, information and correlation of an object is determined at various magnification scales. To put it simple, for Platonic objects which have a smooth boundary like a circle, square or a line this exponent is proportional to the topological dimension. However, when the system or object have irregularities this exponent exceeds the topological dimension [108]. Mathematically, self-similarity or self-affinity based fractal dimension is expressed as [109],

$$D_s = \lim_{r_k \to 0} \frac{\log(N_k)}{\log(1/r_k)} \qquad (1)$$

The above equation implies that self-similarity based fractal dimension can be determined by taking the logarithmic quotient between a measure of object N_k at a given measurement scale $1/r_k$, and the measurement scale itself when the scale tends to zero. However, the measurement itself can be deterministic in nature like morphological or probabilistic like information, correlation, and variance. Some of the examples of dimensions which express the complexity of an object in terms of its geometry include Hausdorff dimension, mass dimension, and gyration dimension. These dimension do not involve any notion of statistics and therefore, are also called monofractals. Contrarily, entropy based dimensions are defined from statistical self-similarity or self-affinity point of view and therefore are probabilistic in nature for example, information dimension, correlation dimension and Renyi dimension [104]. A real-world example of a statistical self-similar process is the fluctuations generated by human heart which are self-similar on time scale [110]. Also, in equation (1), the significance of scale is represented through subscript k which indicates the variation in the measurement with the change in scale. This scale factor in the analysis is responsible to determine the complexity of the objects and hence, is termed as multiscale analysis. Fractals are mathematically elegant because of their self-similarity property which ensures that the fractal dimension is always embedded within the topological dimension.

As established already, feature space selection is challenging due to the multi-faceted threat landscape and therefore, fractal based multiscale analysis is a promising technique to detect advanced threats. The fundamental reason is the ability of the fractals to determine the correlation between different scales of the object under study and how it contributes to the final sophisticated patterns observed in latest cyber-attacks. Moreover, fractals are mathematically tractable and a good variety of implementation algorithms are also available. For instance, fractal dimension of a DNS time series follows the attack pattern in a way that the dimension is increased with the introduction of attack pattern which in turn point towards the increased complexity of the system [111]. Moreover, in this thesis a correlation fractal dimension based variation of the k-Nearest neighbours (k-NN) algorithm has been presented which was able to successfully detect advance persistent threats [103] with high reliability. In addition, information fractal dimension based neural network is also discussed which is able to detect threats which masquerade the behavior of legitimate internet traffic [17]. This is done by employing the multiscale analysis of the error curve and using the fractal based values to reflect back the error in the feedback part of the network. Another interesting approach has been used by researchers in [45] [44] [43] where the structure of the neural network has been transformed to that of a fractal object and then, re-construction and re-organization properties are explored to achieve multiscale analysis based results. Thus, algorithms having cognitive roots are proving to be much more robust perhaps because of the multiscale information distribution or complex data analysis requirement.

4.2 Multiscale analysis using wavelets

One of the challenges while analyzing multiscale objects is to ensure the stationarity both in the temporal and spatial domains [112] which is difficult to achieve over the entire dataset and therefore, a window is selected such that the data within that window is considered stationary. However, this implies a window estimation technique is needed which led to the development of wavelets. Basically, wavelet transform is a mathematical analysis tool which has the ability to simultaneously consider time and space effects and as a result, the stationarity issues are addressed satisfactorily [113] [114]. It should be emphasized that this idea is contrary to the Fourier transform which works on the strict assumption of stationarity of the data whereas, wavelet transform determines the frequency content as a function of time. This is attained through the usage of a suitable wavelet which is able to extract the desired information from the data. In other words, the wavelet will play the role of a microscope which is able to reveal the singularities in fractal measures which may be distributed across several scales. This is possible because of the tendency of a wavelet to remove regular behaviors which exposes the underlying irregularities. Therefore, contrary to the Fourier analysis which focus on the global aspect of the data evaluation, wavelets are predominantly local in their approach of analysis and hence, suitable for multiscale analysis of objects [115].

Various domains of cyber-security like forensics, cryptography, and biometric finger printing have benefited from the usage of wavelet transform as an analysis tool [116]. Moreover, this concept has also been extended to the anomaly detection in cyber-world [117] [111] [118]. In this thesis, a novel hybridization of wavelets with Hebbian learning rule is proposed. Hebbian rule is inherently single scale and linear in its analysis. However, when combined with wavelets the resultant algorithm is able to extract the non-linear and inseparable anomalous samples from the legitimate data samples. It implies that this scale-based technique is useful in not only converting the linearity in algorithm to non-linearity but additionally enable it to extract overlapping data points [119].

An important aspect of wavelets in the performance evaluation is the choice of a particular wavelet window [120] [121] included but not limited to Haar, Biorthogonal, Morlet and Mexican Hat. In this thesis, this particular choice of wavelet is based on the promising empirical results obtained due to compact support of the orthonormal family of wavelets called Daubechies wavelets [121]. These are characterized by their order 'm' which represents the number of vanishing moments that indicate the regularity and support of the wavelet function. Principally, the higher the value of m, the greater will be the regularity and larger the support. The number of vanishing moments tested for this research work were 1, 2, and 4. However, all of them provided approximately similar results and hence, Daubechies db1 wavelet is selected based on computational ease i.e. only 2 coefficients are required in computation.

Chapter V: Dataset and feature extraction

The dataset collection and feature extraction process revealed the multiscale nature of information embedded in the contemporary cyber threats. Whether the data is collected from the honeypot directly or is synthesized carefully in the laboratory, the resultant sample space is overlapping and can be only be analyzed using scale based approaches. This chapter elucidates the data collection process in detail while, highlighting their source of origin. A detailed discussion on standard guidelines for feature extraction is what follows. Further, statistical analysis of the selected feature vector is presented and discussed using illustrations.

In this research work, two publicly available datasets are utilized; (i) IMPACT cyber trust data set and (ii) UNSW dataset. Further, IMPACT dataset is synthetically mixed with APT threats from another publicly available repository of advanced malware maintained by Mila Parkour. Both data sets comprise of packet capture files and fall under the category of network datasets.

1. Advanced Persistent Threats (APT) dataset

The dataset used in the first experiment which was targeted towards the detection of Advanced Persistent Threats (APT) was synthetically combined from two different sources of packet capture files. The first one was the non-malicious or normal data obtained from PREDICT now renamed as IMPACT Cyber Trust internet data set repository [122] available under the category of "DARPA Scalable Network Monitoring (SNM) Program Traffic". The second is the anomalous dataset which has traces of APT and was collected from Contagio malware database [123] contributed by Mila Parkour. The primary reason for using this manual synthesis approach was the fact that no complete real or synthetic dataset that contains both the normal internet flows and anomalous APT flows was available for open research, to the best of author's knowledge. Further details about the collection, filtering and processing of dataset is available in next sub-sections.

1.1 Synthetic combination of packet capture files

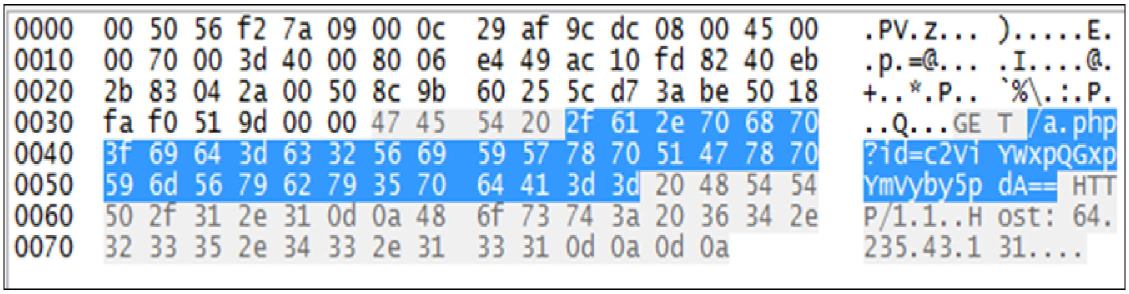
The packet capture files provided by PREDICT were filtered to extract normal TCP packet flows. The APT traces were then embedded into these normal flows to generate a final dataset that mimics the APT attack methodology i.e. slow and low. Various types of APT's used in this experiment is summarized in Table 1. Also, screenshots of the packet capture files for related APT is available in Figure 4, Figure 5 and Figure 7 . Further, it should be noted that this data is not available directly anymore due to the internet safety issues and can therefore, be obtained through a Dropbox link provided by Mila Parkour on request [123]. Nevertheless, the methodology followed to synthesize this dataset was based on a report by Carnegie Mellon University [124] and the related literature [61] [125] [60], which discusses different possibilities of APT detection through the traces of command and control server communication. In addition, [123] and [126] elaborated through examples the APT behavior that ensures low activity over long duration to evade cyber-defence systems like firewalls and has therefore, been helpful for in-lab experimentation carried out to produce a combined dataset mimicking a real APT. It is note-worthy that TCP based command and control APT communications are considered for this experiment. Most of these are based on HTTP protocol and hence, are unable to be blocked by the system administrators.

In addition, the IMPACT DARPA SNM dataset was selected as a benign set of internet flows because it only consists of traces from year 2009 when APT was not a common cyber-attack. It actually surfaced in commercial cyber technologies in 2010 when Stuxnet was discovered [127]. Also, it was deliberated that the normal dataset should not be too old to be considered reasonable for analysis in terms of latest internet traffic patterns. Moreover, this dataset was synthesized by DARPA to simulate real world internet traffic for 10 days from November 3 to November 12, 2009. The total size of the capture file is 6TB and contains HTTP, SMTP and DNS traffic. However, for this work only 3GB of data was used from November 3, 2009 due to resource limitations. Also, files of size 603.2 MB consisting of various APTs were also used.

#	Threat	Protocol	Method	HTTP header Pattern (Signature)
1	Darkcomet	GET	HTTP	/a.php?id=c2ViYWxpQGxpYmVyby5pdA==
2	DNS Watch	GET	HTTP	/dns/dnslookup?la=en&host=vcvcvcvc.dyndns.org&type =A&submit=Resolve
3	Gh0st-gif	GET	POP	/h.gif?pid =113&v=130586214568 HTTP/1.1
4	Gh0st v2000 var	-	FTP	v2010 … Service Pack 2 …? ... |…| … |0.@ ...
5	IXESHE	GET	SSL	/AWS96.jsp? baQMyZrdI5Rojs9khs9fhnjwj/8mIom9jOKyjnxKjQJA
6	LURK	GET	SOCKS	LURK0……..x.kf.e.apgpgba0c..#..
7	Pingbed	GET	HTTP	/Default.htm
8	Taidoor	GET	SSL	/gmzlk.php?id=031870111D309GE67E
				/query.jsp?tb=eecnhr111D308CB9EB
9	Vidgrab	POST	HTTP	(172.16.253.130)|1067|WinXP|D|L|No| 0..0....1..52..|No|V2010-v24|2184|0|3111947|0|1|.
10	Xtreme Rat	GET	HTTP	/1234567890.functions
11	9002	POST	TCP	9002.................wx....9002.................wx....9002.............. ....
12	Poison Ivy	GET	-	256 bytes of seemingly random data after a successful TCP handshake, then 48 byte "keep-alive" requests
13	Variant Letsgo	GET	HTTP	/index.htm
14	Mediana	GET	HTTP	/index.htm?n763t4OPm*rs6fXq7fXp7uj16e-r&Length=0
15	Hupigon	-	SOCKS	....................................;... Windows XP 5.1 (2600.Service Pack 3)......................... ....$...DELLXT................................. .. .. 4s.love.......HACK..
16	Scieron	-	TCP	packet data <p>0000 16 03 01 00 41 01 00 00 3d 03 01 54 c1 2a fa 82
				<p>0010 a5 0b 00 4c 7b 26 c9 33 81 bd 63 34 08 ab b3 38
				<p>0020 3a de 83 db b1 9c 95 02 3e c3 34 00 00 16 00 04
				<p>0030 00 05 00 0a 00 09 00 64 00 62 00 03 00 06 00 13
				<p>0040 00 12 00 63 01 00
17	Sanny	POST	HTTP	/write.php
18	Netraveler	GET	HTTP	/fly/2013/2011/nettraveler.asp?hostid=E81B9088&hostname=DellXT&hostip=172.16.253.130&filename=travlerbackinfo-2013-1-14-0-29.dll

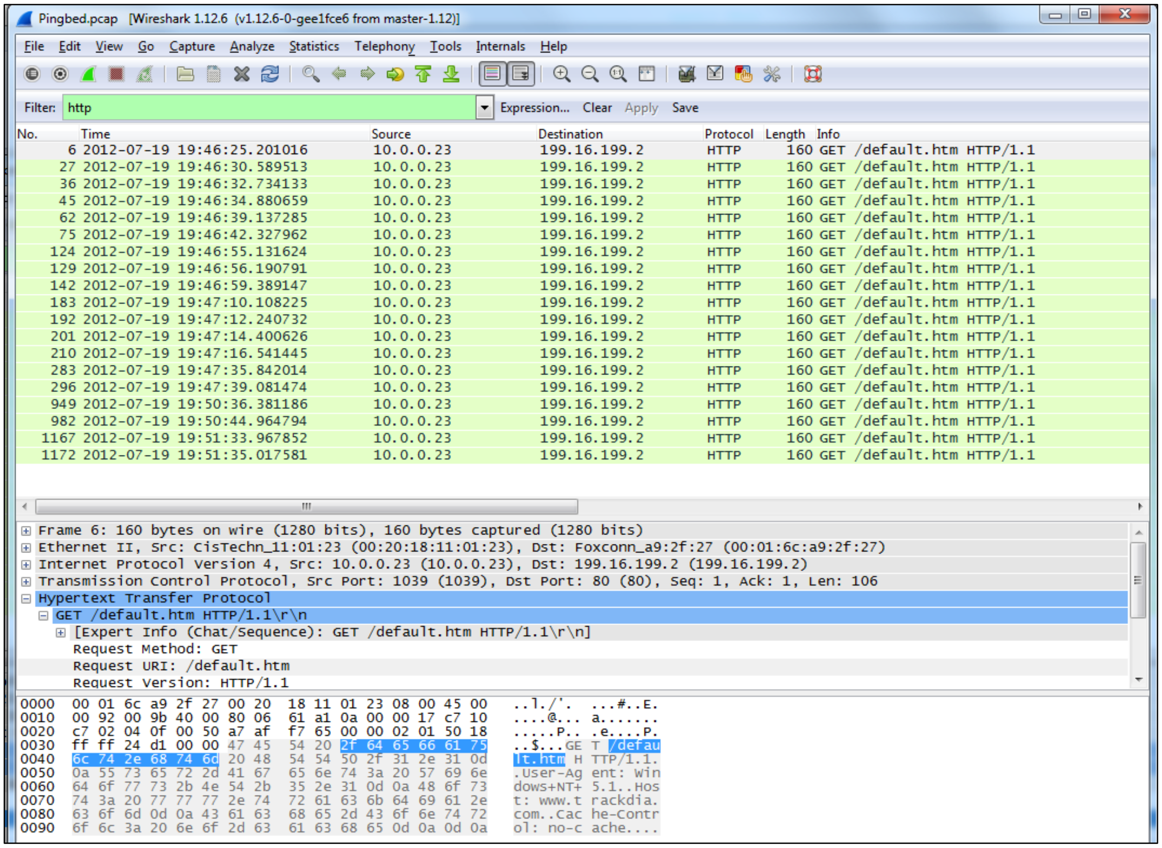
Table 1: Summary of APT flows used from Contagio malware repository [103].

An abstract view of the combined capture files is provided in Figure 6. This particular arrangement was considered to reflect the characteristic of the APT which are low speed and have smaller share in the total internet traffic which makes them difficult to be detected. In addition, APT attacks are often spread over years and are backed by state-actors which employ sophisticated evasion techniques. One such technique is to use a variation of the known-APT like Gh0st RAT which has numerous mutated copies utilized in cyber-espionage. Therefore, the combined dataset also includes the variants of Gh0st RAT to ascertain the performance of proposed Fractal based k-Nearest Neighbourhood algorithm discussed in the next chapter. Moreover, the analysis of the APT data revealed that the attacks do not only use port 80/8080 but, as well utilize ports 443, 110, 21, etc. Hence, the final capture file comprised of an assorted set of APTs that was further utilized for feature extraction.



```
0000  00 50 56 f2 7a 09 00 0c  29 af 9c dc 08 00 45 00   .PV.z... ).....E.
0010  00 70 00 3d 40 00 80 06  e4 49 ac 10 fd 82 40 eb   .p.=@... .I....@.
0020  2b 83 04 2a 00 50 8c 9b  60 25 5c d7 3a be 50 18   +..*.P.. `%\.:.P.
0030  fa f0 51 9d 00 00 47 45  54 20 2f 61 2e 70 68 70   ..Q...GE T /a.php
0040  3f 69 64 3d 63 32 56 69  59 57 78 70 51 47 78 70   ?id=c2Vi YWxpQGxp
0050  59 6d 56 79 62 79 35 70  64 41 3d 3d 20 48 54 54   YmVyby5p dA== HTT
0060  50 2f 31 2e 31 0d 0a 48  6f 73 74 3a 20 36 34 2e   P/1.1..H ost: 64.
0070  32 33 35 2e 34 33 2e 31  33 31 0d 0a 0d 0a         235.43.1 31....
```

Figure 4: “Dark Comet” malware pattern [103].


Pingbed.pcap [Wireshark 1.12.6 (v1.12.6-0-gee1fce6 from master-1.12)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Expression... Clear Apply Save

Filter: http

```
No.     Time                        Source         Destination      Protocol Length Info
      6 2012-07-19 19:46:25.201016  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
     27 2012-07-19 19:46:30.589513  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
     36 2012-07-19 19:46:32.734133  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
     45 2012-07-19 19:46:34.880659  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
     62 2012-07-19 19:46:39.137285  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
     75 2012-07-19 19:46:42.327962  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
    124 2012-07-19 19:46:55.131624  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
    129 2012-07-19 19:46:56.190791  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
    142 2012-07-19 19:46:59.389147  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
    183 2012-07-19 19:47:10.108225  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
    192 2012-07-19 19:47:12.240732  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
    201 2012-07-19 19:47:14.400626  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
    210 2012-07-19 19:47:16.541445  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
    283 2012-07-19 19:47:35.842014  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
    296 2012-07-19 19:47:39.081474  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
    949 2012-07-19 19:50:36.381186  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
    982 2012-07-19 19:50:44.964794  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
   1167 2012-07-19 19:51:33.967852  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
   1172 2012-07-19 19:51:35.017581  10.0.0.23      199.16.199.2     HTTP       160 GET /default.htm HTTP/1.1
```

```
⊞ Frame 6: 160 bytes on wire (1280 bits), 160 bytes captured (1280 bits)
⊞ Ethernet II, Src: CisTechn_11:01:23 (00:20:18:11:01:23), Dst: Foxconn_a9:2f:27 (00:01:6c:a9:2f:27)
⊞ Internet Protocol Version 4, Src: 10.0.0.23 (10.0.0.23), Dst: 199.16.199.2 (199.16.199.2)
⊞ Transmission Control Protocol, Src Port: 1039 (1039), Dst Port: 80 (80), Seq: 1, Ack: 1, Len: 106
⊟ Hypertext Transfer Protocol
  ⊟ GET /default.htm HTTP/1.1\r\n
    ⊞ [Expert Info (Chat/Sequence): GET /default.htm HTTP/1.1\r\n]
      Request Method: GET
      Request URI: /default.htm
      Request Version: HTTP/1.1
```

```
0000  00 01 6c a9 2f 27 00 20  18 11 01 23 08 00 45 00   ..l./'.  ...#..E.
0010  00 92 00 9b 40 00 80 06  61 a1 0a 00 00 17 c7 10   ....@... a.......
0020  c7 02 04 0f 00 50 a7 af  f7 65 00 00 02 01 50 18   .....P.. .e....P.
0030  ff ff 24 d1 00 00 47 45  54 20 2f 64 65 66 61 75   ..$...GE T /defau
0040  6c 74 2e 68 74 6d 20 48  54 54 50 2f 31 2e 31 0d   lt.htm H TTP/1.1.
0050  0a 55 73 65 72 2d 41 67  65 6e 74 3a 20 57 69 6e   .User-Ag ent: Win
0060  64 6f 77 73 2b 4e 54 2b  35 2e 31 0d 0a 48 6f 73   dows+NT+ 5.1..Hos
0070  74 3a 20 77 77 77 2e 74  72 61 63 6b 64 69 61 2e   t: www.t rackdia.
0080  63 6f 6d 0d 0a 43 61 63  68 65 2d 43 6f 6e 74 72   com..Cac he-Contr
0090  6f 6c 3a 20 6e 6f 2d 63  61 63 68 65 0d 0a 0d 0a   ol: no-c ache....
```

Figure 5: Screen shot - packet capture of “Pingbed APT” - The pattern *‘/default.htm’* is visible [103].

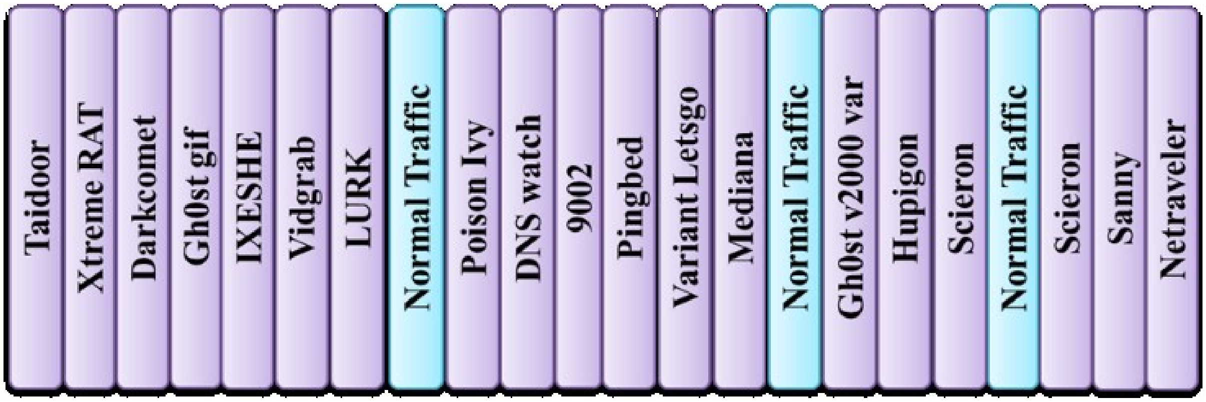


Figure 6: Mixed capture files with anomalous and normal packets - abstract view [103].

1.2 Feature extraction and analysis

An in-depth analysis of the TCP session was carried out for the final packet capture file and high correlation between the two attributes and APT instance was found [103]. The two metrics are: total number of data packets exchanged during a single TCP session and the complete duration of the TCP session. It is interesting that [126] [125] already reported that the activity of an Advanced

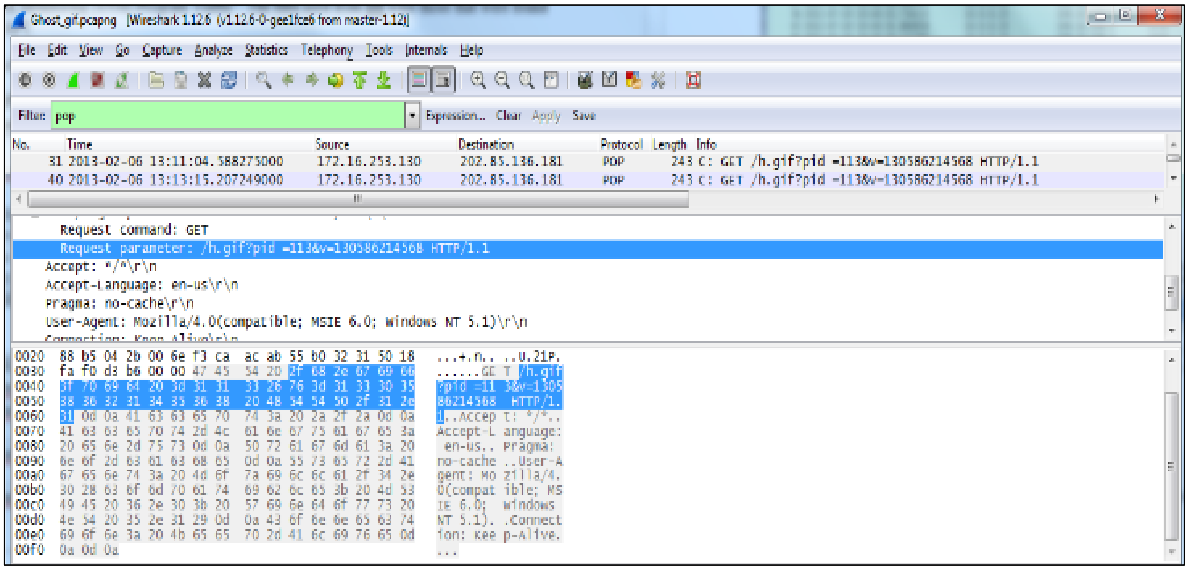
Persistent Threat consists of smaller number of packets in either short lived TCP sessions or long TCP sessions. Contrarily, the normal internet traffic exhibited patterns with high packet count in short TCP session duration because of TCP being connection oriented. Therefore, the two mentioned attributes were selected as a feature vector for our classification system.

A command line tool for manipulation of packet capture files is Tshark [128] which was used to obtain the required features from the raw data. The extracted data was later labelled as positive for an attack and negative for a normal sample. The ratio of the training, cross validation and testing data was 40%, 25% and 35% respectively.

1.3 Data filtering

The data pre-processing step required removal of any noise from the extracted features to ensure robust training of the classification system. The noise for the selected feature vector consisted of two types of packets which are removed before feeding the data into the learning engine.

1. The zero length TCP packets were removed as they did not contribute towards the actual data exchange occurred between the source and destination.
2. The re-transmitted packets were removed as they do not affect the total packet count. This is because at the receiver end, the re-transmitted packet is either discarded if the original packet already reached the destination or it is kept in queue if the actual packet did not make it to the receiver. In either case, the total packet count remains the same.


Ghost_gif.pcapng [Wireshark 1.12.6 (v1.12.6-0-gee1fce6 from master-1.12)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: pop
Expression... Clear Apply Save

```
No.  Time                           Source          Destination     Protocol Length Info
  31 2013-02-06 13:11:04.588275000  172.16.253.130  202.85.136.181  POP         243 C: GET /h.gif?pid =113&v=130586214568 HTTP/1.1
  40 2013-02-06 13:13:15.207249000  172.16.253.130  202.85.136.181  POP         243 C: GET /h.gif?pid =113&v=130586214568 HTTP/1.1
```

```
      Request command: GET
      Request parameter: /h.gif?pid =113&v=130586214568 HTTP/1.1
    Accept: */*\r\n
    Accept-Language: en-us\r\n
    Pragma: no-cache\r\n
    User-Agent: Mozilla/4.0(compatible; MSIE 6.0; Windows NT 5.1)\r\n
    Connection: Keep-Alive\r\n
```

```
0020  88 b5 04 2b 00 6e f3 ca  ac ab 55 b0 32 31 50 18   ...+.n.. ..U.21P.
0030  fa f0 d3 b6 00 00 47 45  54 20 2f 68 2e 67 69 66   ......GE T /h.gif
0040  3f 70 69 64 20 3d 31 31  33 26 76 3d 31 33 30 35   ?pid =11 3&v=1305
0050  38 36 32 31 34 35 36 38  20 48 54 54 50 2f 31 2e   86214568  HTTP/1.
0060  31 0d 0a 41 63 63 65 70  74 3a 20 2a 2f 2a 0d 0a   1..Accep t: */*..
0070  41 63 63 65 70 74 2d 4c  61 6e 67 75 61 67 65 3a   Accept-L anguage:
0080  20 65 6e 2d 75 73 0d 0a  50 72 61 67 6d 61 3a 20    en-us.. Pragma: 
0090  6e 6f 2d 63 61 63 68 65  0d 0a 55 73 65 72 2d 41   no-cache ..User-A
00a0  67 65 6e 74 3a 20 4d 6f  7a 69 6c 6c 61 2f 34 2e   gent: Mo zilla/4.
00b0  30 28 63 6f 6d 70 61 74  69 62 6c 65 3b 20 4d 53   0(compat ible; MS
00c0  49 45 20 36 2e 30 3b 20  57 69 6e 64 6f 77 73 20   IE 6.0;  Windows 
00d0  4e 54 20 35 2e 31 29 0d  0a 43 6f 6e 6e 65 63 74   NT 5.1). .Connect
00e0  69 6f 6e 3a 20 4b 65 65  70 2d 41 6c 69 76 65 0d   ion: Kee p-Alive.
00f0  0a 0d 0a                                           ...
```

Figure 7: “Gh0st RAT”, a second stage malware used by the Gh0stnet attackers [103].

1.4 Statistical information

The statistical analysis of the data after the pre-processing step is available in Table 2. The table provides an abridged information about the normal and anomalous samples. Nevertheless, it should be noted that data packet counts per TCP session for each flow is used in the experiment. Combining both the benign and malicious data, there are 38,358 sessions for each set of source and destination IP addresses. The APT session count accounts for only 378 sessions while the rest 37,980 sessions belong to the PREDICT dataset.

Moreover, a distribution of session is pictorially represented in Figure 8 such that each point on the plots corresponds to the total packets exchanged in a TCP session for APT and normal traffic respectively. A close analysis of these figures reveals that these data samples are predominantly over-lapping. In other words, the benign and malicious samples are indistinguishable on selected feature space because of the sophisticated APT design which try to follow the normal traffic pattern to evade detection. Also, it is observed that the infected system creates and deletes multiple short lived TCP connections with the command and control server. Each of these sessions have very small data exchanges and therefore, are difficult to get detected. On the other hand, normal TCP communication tend to live longer with as much data exchange as possible [129].

S. No.	**Attack Name/ Normal Traffic**	**Total Packets (after noise removal)**	**Total TCP Session Count (after noise removal)**	**Total Duration (sec) for all TCP sessions (after noise removal)**
1	Taidoor	116	63	14.9623
2	Xtreme Rat	2578	8	523.5481
3	Dark Comet	2	1	0.1249
4	Gh0st-gif	3	3	0
5	IXESHE	3	3	0
6	Vidgrab	245	1	0
7	LURK	210	42	1156.4667
8	Normal Traffic	658677	12796	22716.4547
9	Poison Ivy	89	1	2888.3428
10	DNS Watch	247	2	1202.648
11	9002	3585	1	369.6251
12	Pingbed	19	19	363.6382
13	Variant Letsgo	277	1	0
14	Mediana	266	133	12.1138
15	*Normal Traffic*	*662219*	*12812*	*41.466*
16	Ghost v2000 var	26	25	2013.0399
17	Hupigon	66	33	.2886
18	Scieron	60	3	0.0093
19	Normal Traffic	667195	12372	69.9476
20	Scieron	64	3	1554.7916
21	Sanny	531	8	124.8862
22	Netraveler	68	28	47.7137

Table 2: Processed statistics of APT+Normal packets from Contagio + IMPACT dataset [103].

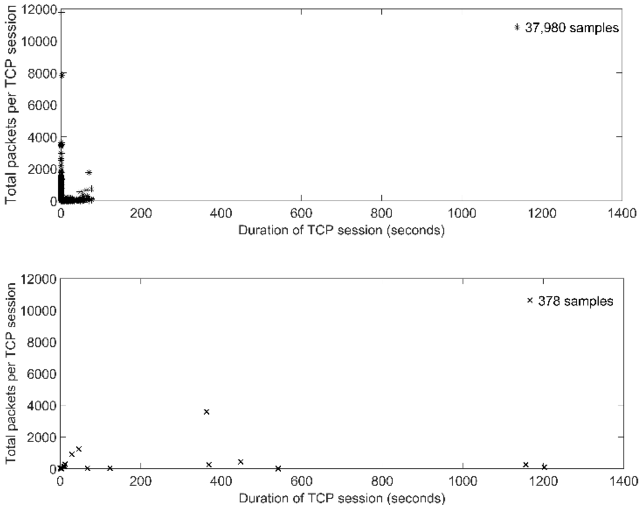


Figure 8: Packets vs. TCP session's duration Normal (top) and APT (bottom) [103].

2. UNSW dataset

The second and third experiment presented in this thesis is related to the detection of anomalous events which overlap the normal data points on a feature space. Therefore, multiscale analysis was introduced into neural network and Wavelet based Hebbian learning algorithm; the algorithmic details are available in next chapter. The first challenge of this research study was to obtain a real-world dataset which ascertain our observation of coinciding benign and malignant samples over a feature space. The **UNSW-NB15** dataset which was generated and processed at Cyber Range Lab of the Australian Centre for Cyber Security (ACCS) [130] was selected for testing the performance of algorithm. This data set was captured in year 2014 and is available in **PCAP** (packet capture) and **CSV** (Comma Separated Values) format.

The dataset consists of 49 different attributes of internet communication. Some of which were obtained directly from the raw data like source and destination IP, protocol and states. The other parameters were derived from these e.g. number of last 100 connections with same destination and source IPs. Complete details about the data capture setup and files information is available at [131] [132]. Contrary to the previous dataset which comprised only of APT, this dataset has multiple attacks as illustrated in Figure 9. However, this work was not targeted towards the multiclass detection of threats. Rather, all of these attacks were considered as part of the single threat class and were labelled positive. On the other hand, the normal class was labelled as negative thus, transforming the problem into a binary classification challenge.

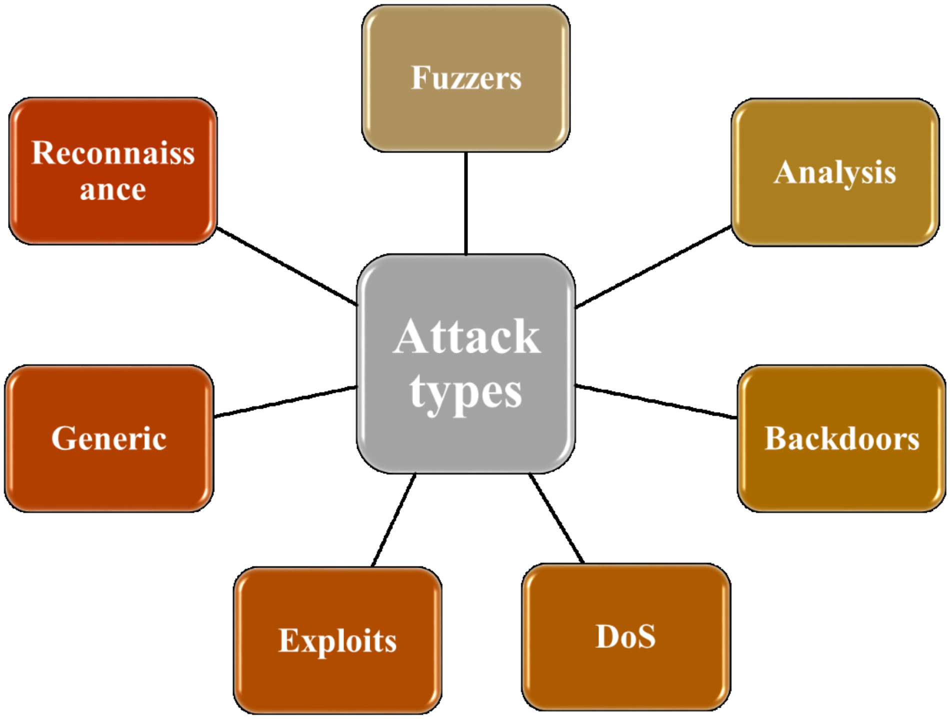


Figure 9: Attack types in UNSW dataset.

Further, the UNSW dataset has multiple internet protocols, however, this work is concentrated towards HTTP based flows. Statistically, it was found that 92% of the cyber threats are exploiting HTTP protocol in one way or the other [4]. This is plausible because of the extensive utilization of HTTP protocol in various forms of internet based communication thus, acting as a critical backbone of internet infrastructure that cannot be blocked by system administrators. This has provided threat actors with an opportunity to exploit the protocol without getting detected. Moreover, in some situations the protocol is not abused however, still it functions as a cornerstone in the attack success. An example of such a scenario is the HTTP based re-direction to download any malware. Another common example is the communication of a bot with the command and control server which piggybacks on HTTP protocol at its core. Thus, from the perspective of [133], it can be considered an indicator of attack (IoA).

There are 206273 total flows out of which 9.13% are anomalous. Table 3 depicts a detail information of benign and malignant HTTP sample in four filtered UNSW files.

UNSW File No.	Anomalous Samples	Normal Samples	Total Samples	Ratio of Anomalous to Total Samples
1	1971	53887	55858	3.52%
2	4163	61661	65824	6.32%
3	7866	43408	51274	15.34%
4	4847	28470	33317	14.54%
Total	18847	187426	206273	9.13%

Table 3: Detailed breakdown of HTTP samples - UNSW-NB15 dataset [17].

2.1 Feature selection criteria

Once, the HTTP based samples were filtered, the feature extraction process began. Mainly four features were selected out of the available 49 different attributes. The feature selection criteria was based on the following guidelines [134] [135]:

- Ideally, selected features should be independent and thus, uncorrelated. In real-world scenarios this is not a feasible condition to be fulfilled and hence, a weak independence is the least expected characteristic of the feature vector. This is necessary to avoid over-fitting or under-fitting of the data and to achieve adequate generalization in terms of results.
- Feature vector should be representative of the system dynamics. For instance, a TCP packet based features should signify an end to end communication of the packet flow.
- Uniqueness should be ensured while selecting the features. It implies avoiding any possible overlap. For example, if the objective is to identify various flows based on their protocol, one possible approach could utilize port information. However, there are protocols that use the same ports but, are inherently different.
- Features can be similar to the parameters or attributes of the system if there exists a strong correlation with the system dynamics. However, the selected features must adequately depict the subjective behavior of the system under consideration.
- A feature vector should represent all the classes in the dataset i.e. normal and anomalous class.

The dataset was distributed in 3 subsets namely, training, validation and testing sets. A detailed information about the anomalous and normal samples for each subset is available in Table 4. The ratio between the three sets is 70%, 15%, and 15%.

No.	Data category	Normal Samples	Anomalous Samples	Total Samples	% of anomalous data
1	Training	65153	4847	70000	6.9%
2	Validation	13654	1346	15000	8.9%
3	Testing	12611	2389	15000	15.9%

Table 4: Dataset division of training, validation and testing samples - UNSW-NB15 dataset.

2.2 Feature extraction and analysis

A thorough analysis of different data attributes led to the selection of four main features namely flow duration, round trip time, total packet inter-arrival time and total bytes count exchanged between source and destination. The feature vector selection process was based on considering the attack strategies by the adversaries to launch and carry out stealth attack. This requires malicious samples to be hidden under the normal data patterns to stay undetected. It is also common with latest threats to learn the behavior of the victim's system and follow the learned pattern to remain surreptitious. For example, if the current network state depicts a total of 1000 TCP flows with an average duration of 2-3 minutes per flow, then in order to avoid detection, the attack patterns should as well follow this behavior. Moreover, the data exfiltration will be carried out in smaller chunks rather than large detectable chunks. In short, the stealth objective of an attack will be to avoid generating any outlier or anomaly which may trigger any threat detection system. The data distribution for each of the selected features is shown in Figure 10, Figure 11, Figure 12 and Figure 13. The illustrations revealed the overlapping characteristic of the positive and negative samples embedded in the selected feature space.

Further, the 2-dimensional view of each of the features which form a 4-dimensional feature space reveals an over-lapping sample space. In cyber-realm, this problem reflects the sophisticated attacks which are able to mutate their characteristics with respect to the target system. This implies that the determination of a reliable and distinct, linear or non-linear classification boundary in single scale Euclidean space is not possible. This particular ineptitude of the traditional machine learning algorithms is independent of the dimensionality of the dataset. Rather, it can be attributed towards the phenomenon of emergence in scale-invariant internet datasets.

Moreover, this phenomenon of overlap was expected due to the heavy tailed distribution [119] produced by both classes of samples. Basically, the positive and negative samples are not only scattered but, have abundance of outliers. Therefore, the classical machine learning methods based on Euclidean distances which are used to determine a classification boundary in such scenario prove to be futile [17]. Also, Figure 10, Figure 11, Figure 12 and Figure 13 represent the histogram of each of the selected feature which as well has the heavy tailed distribution property. To ascertain our observation, Kolmogorov-Smirnov (KS) test was performed with the assumption that no particular heavy tailed distribution is being followed by the dataset. However, the test results produced a high mean and variance implying the presence of an underlying heavy tailed distribution. It signifies the fact that the sample values which are much larger than the mean occur frequently. This inference was in accordance with the web traffic behavior as explained in [136]. Further, the heavy tailed distributions have a fascinating attribute of scale-invariance which is observed in the internet traffic as well [137]. Mathematically, a dataset F is scale-invariant if there exists an x_0 and a g such that,

$$\overline{F}\,(\lambda x) = g(\lambda)\overline{F}(x) \tag{2}$$

for all λ, x such that $\lambda x \geq x_0$ [119]. It should be noted that this condition is satisfied only when the considered distribution is heavy-tailed like Pareto [138]. Moreover, authors in [139] proved that long range dependence is another feature of heavy tailed distribution which coincide with the scale-invariant property of a dataset. Hence, a multiscale based intelligence is proposed in standard neural network and Hebbian learning algorithms, the details of which are available in next chapter.

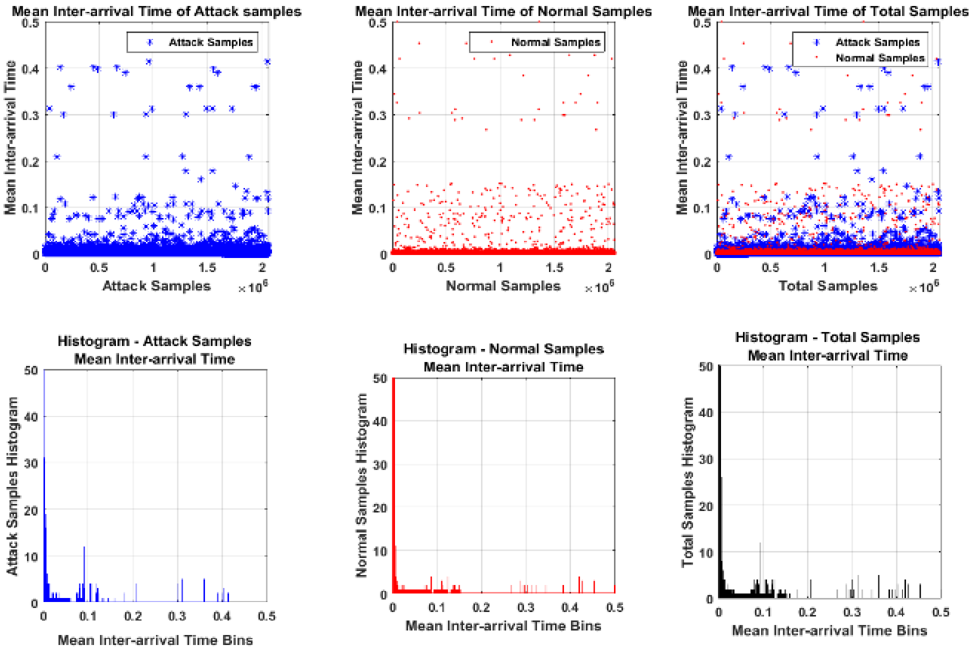


Figure 10: Packet interarrival time [17].

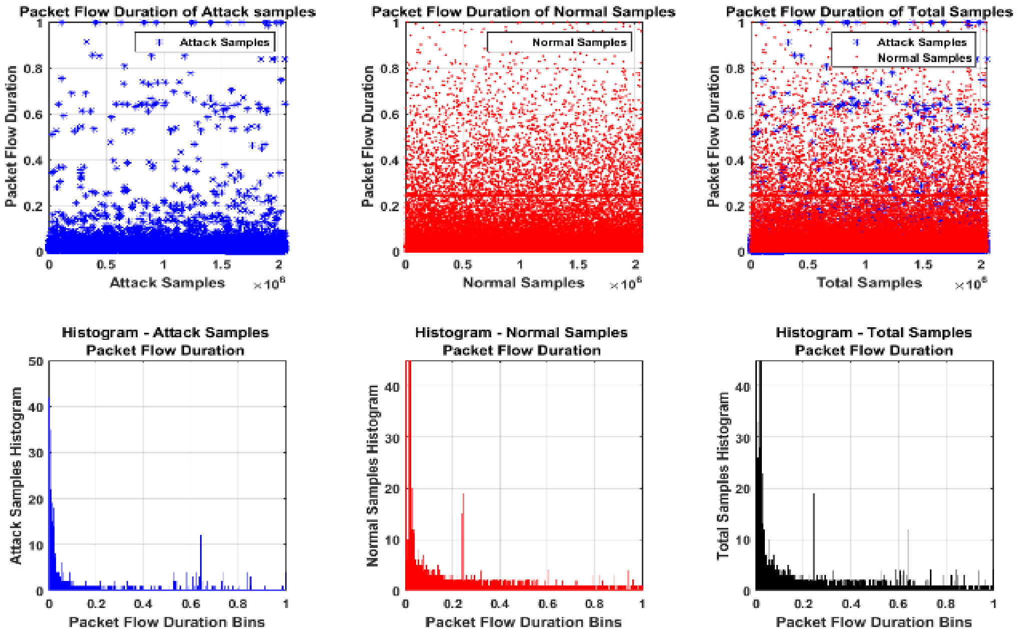


Figure 11: Packet flow duration [17].

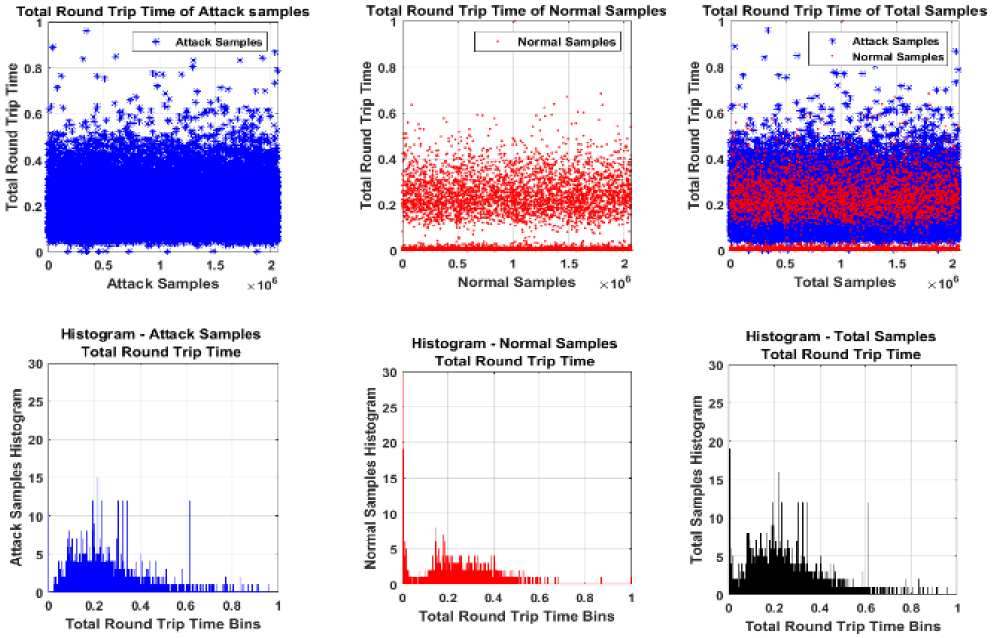


Figure 12: Total round trip time [17].

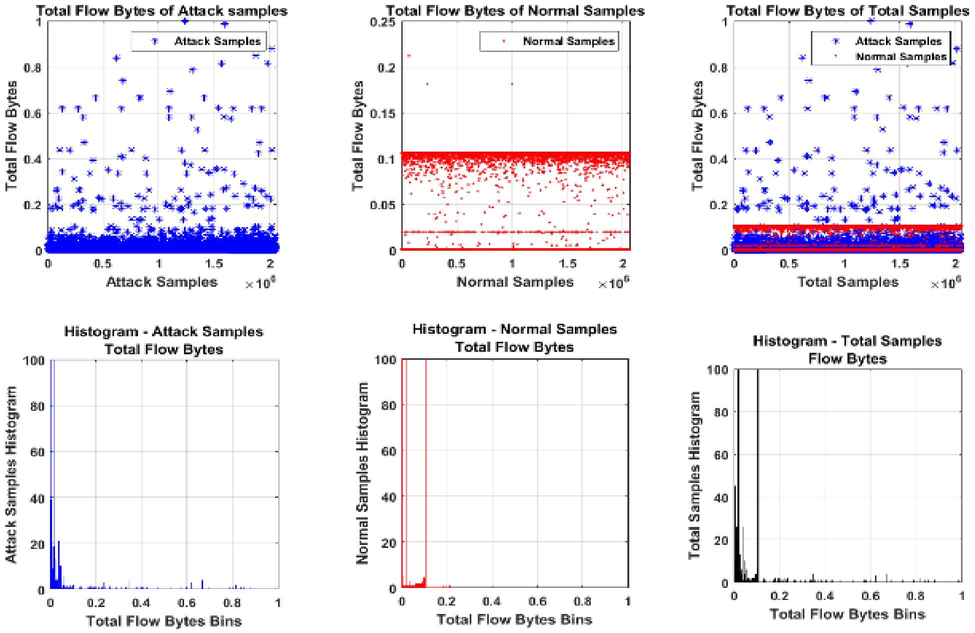


Figure 13: Total bytes exchanged in a flow [17].

Chapter VI: Cognitive machine intelligence algorithms

The single scale nature of the traditional machine learning algorithms made them susceptible to the dynamic changes in feature space which in turn result in misclassification. In cyber realm, it is getting difficult to find a reliable classification boundary because of the tendency of advanced threats to morph and follow the behavior of normal internet activity. This gives rise to the class overlap problem which is in addition to the class imbalance challenge generally observed in an internet dataset. This chapter addresses these problem by proposing multiscale modification in three of the existing machine learning algorithms using fractals and wavelets. Each of the section consists of a detailed description of traditional model, the multiscale modification and the pseudo code implemented in MATLAB along with introductory mathematical background.

1. Instance based learning algorithm – traditional and proposed fractal based cognitive algorithm

Instance based learning algorithms classify a particular sample based on comparison with the previously stored training examples in the memory. In other words, these algorithms assign a target value, normally discrete, to the presented data instance according to its resemblance strength with the training instances. It implies that contrary to the learning algorithms that tend to generalize their approach, these delay the data processing until a new sample is encountered and hence, are referred to as 'lazy' learning algorithms [140]. The primary reason for considering this category of algorithm is its local and distinct estimation approach for classification of each instance depending on the selected similarity index e.g. neighbourhood count.

The first experiment is focused on the performance comparison of the standard k-Nearest Neighbours (k-NN) algorithm and the proposed fractal based k-NN which is defined in detail in the subsequent sections.

1.1 Traditional k-Nearest Neighbors (k-NN)

One of the pre-dominantly used type of instance based learning algorithms is k-Nearest Neighbours (k-NN) which utilizes the training instances to classify a new sample instance. This approach does not have any generalized learning methodology rather it is the resemblance with the local training examples which define the classification outcome of the instance under observation [141]. The similarity metric used in the algorithm is often a distance in the selected sample space like Euclidean distance and accordingly the class labels are assigned. Specifically, the **k** in k-NN signifies the number of neighbours that are used in the computation of similarity measure. The decision about the class label is then based on majority class vote.

Mathematically, this estimation algorithm can be represented as [103]:

$$\widehat{\mathrm{y(x)}} = \mathrm{y}_{\mathrm{n}^*} \tag{3}$$

where, n^* is defined as,

$$\mathrm{n}^* = \arg\min \mathrm{dist}(\mathrm{x}, \mathrm{x}_\mathrm{n}) \tag{4}$$

It is important to note that k-NN is a non-parametric classification algorithm and due to its local nature, it is sensitive to the geometrical structure of the data under observation [140] [142]. Therefore, it is reasonably anticipated that it will perform well with the complex dataset which has multiple local approximations like that of fractals.

The fundamental limitation of instance based learners is the associated high computation cost because the learning computations are being performed during the classification time rather during the training stage [142]. Therefore, the storage and retrieval of the training instance is one of the major practical issue. Moreover, the determination of nearest neighbours for each instance involve building the model from scratch and therefore, requires enormous computations which affects classification speed. This is in addition to the high memory requirements which must be managed to save and query training data [143]. In addition, as the distance is calculated based on all the features, the redundancy or irrelevance in the feature vector causes the algorithm to suffer from the curse of dimensionality. Different modifications have been proposed to address the stated problems of high-dimensionality, time and memory scalability in the traditional k-NN in [144] [145] [146]. Nevertheless, k-NN performs well with the dataset having low variance and low bias which is generally not the case for cyber threat data. In other words, the advanced attacks try to mimic the behavior of the normal data and hence, conceal themselves under a benign sample on a feature space resulting in a high-variance and high-bias problem. Therefore, fractal based modification has been proposed in the standard k-NN to address the stated issue through multiscale analysis.

1.2 Proposed correlation fractal dimension based algorithm [103]

Instead of the traditional Euclidean metric, a correlation fractal dimension based similarity index is proposed in the standard k-NN algorithm. A labelled reference dataset of selected feature vector is a pre-requisite for testing this algorithm. It works incrementally by comparing the correlation fractal dimension of each cluster of data instances i.e. the negative and positive groups, after the addition of the new data point to the anomalous and benign class successively. The next step is to separately compute the correlation fractal dimension of the attack and normal reference datasets, which act as the prototypical measure of each class. For the classification of the presented sample, the prototypical correlation fractal dimension of the normal and threat data sets is compared with the newly computed correlation fractal dimension of each cluster after the addition of the new sample. The cluster with the minimum change in the fractal dimension is selected as the class of the particular instance. In short, this process involves determining the multiscale similarity index of the new data instance with each class and selecting the label to which the input data is most comparable [86].

The similarity metric selected for the proposed algorithm is the correlation dimension, D_c [111]. In order to compute this, the most important step is to fix a frame of reference such that it consists of all the available samples. Then, it is covered by volume elements called vels denoted by N_r and each element has a resolution scalable by r. The probability of the jth vel, p_j, is defined in terms of frequency of intersection of the jth vel, n_j, as follows [147]:

$$p_j = \lim_{N_T \to \infty} \frac{n_j}{N_T} \qquad (5)$$

$$N_T = \sum_{j=1}^{N_r} n_j \qquad (6)$$

If a power law relationship holds between the squared probabilities summed over all the vels having scaling diameter r [111] :

$$\left(\sum_{j=1}^{N_r} p_j^{2}\right)^{-1} \sim \left(\frac{1}{r}\right)^{D_c} \tag{7}$$

Then, the correlation dimension is given by the following relation [111] [147] [86]:

$$D_c := \lim_{r\to 0} \frac{-\log \sum_{j=1}^{N_r}\left(p_j^{2}\right)}{\log\left(\frac{1}{r}\right)} \tag{8}$$

An important aspect of the correlation fractal dimension computed for the presented research is that it is an estimate with finite values of the scaling factor r. However, to ensure accuracy of results the minimum selected value of scaling factor is at least 5 such that a reliable linear log-log relationship can be estimated [111].

1.2.1 Pseudo code [103]

```
1) Initialize a labelled reference dataset, R, comprising of at least 30-40% of the total data to be classified.
2) Compute the prior Correlation Fractal Dimension, fd_prior_anom, of only anomalous data samples in R.
3) Compute the Correlation Fractal Dimension, fd_prior_norm, of only normal data samples in R.
4) Load a set S of data points in the main memory.
5) For each point or sample p in S:
   a) Re-calculate the Correlation Fractal Dimension, fd_posterior_anom, by adding p to anomalous values of R. Compute the change in Correlation Fractal Dimension, fd_anom = abs(fd_posterior_anom - fd_prior_anom).
   b) Re-calculate the Correlation Fractal Dimension, fd_posterior_norm, by adding p to normal values of R. Compute the change in Correlation Fractal Dimension, fd_norm = abs(fd_posterior_norm - fd_prior_norm).
   c) If, fd_anom < fd_norm,
         classify p as anomalous.
   d) If, fd_norm ≤ fd_anom,
         classify p as normal.
```

2. Artificial Neural Network – traditional and proposed fractal based cognitive algorithm

Artificial Neural Networks (ANN) is one of the foundational techniques in artificial intelligence which have stem from the human curiosity to understand and mimic the cognition process in human brain. It has been defined as a highly connected and complex network of tiny machines capable of performing intense cognitive tasks when actuated by a signal [148]. It has striking resemblance with the human brain which consists of a connected system of neurons that can process information by mapping the output of each neuron with the other to create a distinct and impactful neurological signal.

Following the design of human brain [149], there are two main components which constitute an artificial neuron in ANN [150] i.e. the activation function and the weights of the connected links. These two components can be altered in different interconnected configuration to perform a variety of tasks such as classification, filtering, prediction and clustering. Further, to learn the behavior of the dataset dynamically, the ANNs reorganize themselves through weight adjustments on the connected links [151] [152]. The basic technique is to let the neurons improve their learning subsequently by feeding back the error in the previous iteration.

2.1 Gradient Descent based Artificial Neural Network (ANN)

A single hidden layer artificial neural network is considered for experimenting in this research work. There are total three layers involved; one input layer with four neurons, one output layer with single neuron, and one hidden layer with four neurons. The basic rationale behind the selection of a single hidden layer is to compare the classification performance of the single scale standard and proposed multiscale ANN. The addition of multiple layers may introduce unnecessary complexity in the context of comparing the two algorithms. The primary motivation was to observe and analyze the changes in the classification results by introducing multiscale error based feedback path keeping every other component in the model simple and constant. Moreover, the selection of the number of neurons have been done using the standard guidelines as discussed in the subsequent sections.

In this work, the artificial neural network model is based on the gradient descent algorithm and therefore, the activation function for hidden and output layers is one of the critical design considerations. This can be attributed to the usage of gradient of the activation function in the backpropagation algorithm which is responsible for updating the weights and bias values. A detailed list of transfer function which acts as the activation function is available in [153]. However, for the ANN model in this experiment, the hidden layer neurons have an activation function of $tanh()$ which is a rescaled version of the standard $sigmoid()$ function and has an output range of $(-1\,,1)$. This particular transfer function is selected based on its characteristics of stronger gradient values and avoidance of systematic bias values [154] [155]. As the main objective of the ANN in this problem is to classify the sample as a normal or benign, the output layer neuron has a transfer function of $hardlim(z)$. Generally, the steps from feeding feature vector into ANN till the final class estimation are referred to as feed-forward propagation from the perspective of learning in ANN.

The estimated output is then compared with the desired target value to measure the error which is then feedback to the ANN to incorporate the training aspect (error reduction) in the supervised learning algorithms. One of the most common types of error metric used in ANN is mean squared error (MSE) which is also modified in this experiment to include the multiscale aspect. Basically it is the second statistical moment of the Euclidean error measure and is defined as the average of the squared error between the desired and output value. The major drawback with this absolute error type is that the squaring cause larger values to have more weight compare to the smaller values which diminishes with the increasing power.

This backward error propagation is an optimization strategy which aims at minimizing the error in the subsequent iteration based on the previous iteration. The process is generally based on search methodology [156] that can be local or global. Algorithms like gradient descent and its derivative work on the principle of determining the steepest path through the minimum gradient calculation of the weight space [157]. There are two main problems which are addressed by these algorithms i.e. the search direction and the step size. These minimization techniques are combined with the backpropagation algorithm to achieve desired results as can be found in [158], [150] and [159].

2.1.1 Selection of the number of neurons

One of the critical design consideration in ANN is to select the suitable number of neurons for the hidden layer. The significance of this decision comes from the fact that using more neurons than requires may result in over-fitting of the data which implies that the system will not be able to generalize its performance over variety of datasets. This is attributed to the memory bank behavior depicted by the neural network in its efforts to match the input features with the training dataset. Contrarily, using less than the required neurons in the hidden layer results in the under-fitting of the data reflecting the inability of the system to learn the complex relationships of the features. Hence, it is imperative to understand and follow the best practices while choosing the hidden layer size or the number of neurons. Also, few of the rules that can be a starting point are as follows [160] [161].

1) Total number of neuron in the hidden layer should be in between the feature vector size and the output result.
2) Hidden layer size should not be greater than the number of features. Approximately, the hidden layer size should be two-third of the input layer size plus the output layer size.
3) Mean of the input and output layer size is a good estimate for the selection of number of neurons.

Considering the above stated 3 rules, 4 neurons in hidden layer have been selected for the gradient descent based neural network and proposed cognitive ANN.

2.2 Proposed fractal based Artificial Neural Network (ANN)

The estimated output is generated by the ANN during the forward propagation stage and the error correction is employed in the backward propagation as discussed in the previous section. This work is focused on the modification of error measurement and the corresponding changes in the backpropagation algorithm. Basically, the traditional single scale MSE is replaced with a multiscale information fractal dimension based error calculation to minimize the training error. The introduction of multiscale error complexity measure into the learning paradigm of artificial neural network is a novel change [17]. Instead of using the traditional single scale Platonic Euclidean distance, ANN uses cognitive learning approach to minimize error based on the complexity of objects. Author believes that this concept is analogous to the measures employed by human and other higher life creations to differentiate or assimilate objects.

The information fractal dimension is a measure of self-affinity in the probabilistic sense and is also referred as Shannon information/entropy fractal dimension. To define it mathematically, consider the probability p_{kj}in terms of relative frequency n_{kj} with which an object is intersected by the jth vel and the total number of intersections N_{tk} at scale k,

$$p_{kj} = \lim_{N_{tk}\to\infty} \frac{n_{kj}}{N_{tk}} \tag{9}$$

This probability measure is used to define the Shannon Entropy H_k and is expressed as,

$$H_k = \sum_{j=1}^{N_k} p_{kj} \log p_{kj} \tag{10}$$

The Shannon Information dimension can be defined as:

$$D_I = \lim_{r_k\to 0} \frac{H_k}{\log(\frac{1}{r_k})} \tag{11}$$

This concept of scale based analysis helps in evaluating the cost function at different scales. This is required since the error curve itself is multiscale in nature as the normal and anomalous data samples constitute different dimensions. It signifies the fact that the two classes have different error functions as they have different underlying complexity which becomes evident only through the multiscale analysis.

The application of this multiscale analysis concept can be further extended to the class imbalance problem often encountered in the cyber security dataset [17] [103]. Class imbalance indicates the presence of one class samples much more than the other such that the generalization becomes difficult in terms of learning the minority class. This kind of scenario leads to the biased classification results in favor of majority class. The UNSW-NB15 dataset used in this experiment also exhibits this property as out of total samples only 9.13% are anomalous samples. Further, literature suggests that there are techniques to solve this problem like assigning different classification error costs or re-sampling of the data until equal class representation is achieved [162] [163]. Nevertheless, this optimization is impractical in real world scenarios where the attacks have the capability to rapidly alter their behavior which requires introduction of relevant dynamism in cost function.

The mathematical definition for mean squared error (MSE) for total N samples is represented by t target instances and corresponding y estimate is as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(t-y)^2 \tag{12}$$

For a typical binary classification problem which has only two classes, anomalous and normal, the mean squared error can be represented as the sum of the individual class errors.

$$MSE = MSE(Normal\ Class) + MSE(Anomalous\ Class) \tag{13}$$

As observed in a real world cyber dataset discussed in the previous chapter, the ratio of anomalous to the total samples is fairly low which implies that

$$MSE(Anomalous\ Class) \ll MSE(Normal\ Class) \tag{14}$$

Hence, the total error can be approximated as the error of the majority class which is here the class of normal or benign samples in contemporary internet datasets [164].

$$MSE\ \approx MSE(Normal\ Class) \tag{15}$$

This in turn effects the gradient of the error which can be mathematically expressed as,

$$\left||\nabla MSE(Anomalous\ Class)|\right| \ll ||\nabla MSE(Normal\ Class)|| \tag{16}$$

$$\left||\nabla MSE|\right| \approx ||\nabla MSE(Normal\ Class)|| \tag{17}$$

Hence, the majority class takes the lion's share in controlling the MSE which indicates that classifying all the samples as normal will lower the MSE which however, will cause the system to miss the attack patterns. Moreover, when this error will be propagated back to the system for weight and bias adjustment, the system will tend to learn the pattern that can help classify majority class only. To address this problem, an information fractal dimension based cost function is introduced that estimated the error of each class with respect to its dimensionality. Mathematically, it can be expressed as,

$$\begin{aligned} MSE = {} & FractalDimension_{normal} \cdot MSE(Normal\ Class) \\ & + FractalDimension_{anomalous} \cdot MSE(Anomalous\ Class) \end{aligned} \tag{18}$$

The empirical results revealed that better classification performance can be achieved along with the faster convergence using the proposed modification.

2.2.1 Pseudo code [17]

In order to train and test the dataset with gradient descent based ANN, neural network toolbox of MATLAB has been used [165]. The proposed Information fractal dimension based modifications in the error calculation and the corresponding changes in back-propagation algorithm has been accomplished using MATLAB code. Also, the training phase of the algorithm consists of computing the fractal dimensions of normal and anomalous data separately. Moreover, to address the variance and bias in the data [166], 5x2 cross validation has been used. This means that there are 5 cycles of cross validation while each of them has 2 sub-cycles that utilize alternate batches of the same data.

The algorithmic details of fractal dimension based neural network are as follows:

```
$function$ $\boldsymbol{main}()$
    FOR LOOP (5 loops)
        // During each of the 5 supervised training and testing iterations, data sets $S_1$and $S_2$ are used alternatively as training and testing data sets. (Comment)

        1. Divide the dataset into two parts with the ratio of first to second part as 70% and 30%. Label the sets as $S_1$ and $S_2$ respectively.
        2. $\boldsymbol{FractalNN}(S_1, S_2)$
        3. $\boldsymbol{FractalNN}(S_2, S_1)$
$end$ $\boldsymbol{main}()$
```

```
$function$ $\boldsymbol{FractalNN}$ $(Train\ Dataset\ \boldsymbol{S_{tr}}\ ,Test\ Dataset\ \boldsymbol{S_{ts}})$
// $F_{tr}$ and $T_{tr}$ are the feature set and target labels of data set $S_{tr}$. Similarly, $F_{ts}$ and $T_{ts}$ are the feature set and target labels of data set $S_{ts}$. (Comment)

// Training Phase (Comment)
a. Compute the Information Fractal Dimension, $fd_{ref_{anom}}$ of anomalous data labels in $T_{tr}$.
b. Then, compute the Information Fractal Dimension, $fd_{ref_{norm}}$ of normal data labels in $T_{tr}$.
// Testing Phase (Comment)
c. Initialize a neural network with the input layer size of $n$ and a hidden layer size of $m$. The output layer has a single neuron to classify the samples as normal or anomalous.
d. Initialize the input layer weight matrix $W_{in}$ of size $(nxm)$ by computing cross correlation of $F_{ts}$ and a uniform random vector of size $(nx1)$ that carries input bias values $b_{in}$.
e. Using uniform random number generator, initialize hidden layer weight vector $W_l$ of size $(mx1)$ and a scalar output bias value $b_l$.
f. Set the activation function for the neurons in hidden layer as hyperbolic tangent function which takes weighted features $z_1 = W_{in}.F_{ts} + b_{in}$ as input and is expressed as follows:
    i. $a_1 = \tanh(z_1)$
g. The output layer neuron classifies the weighted hidden layer output samples expressed as follows:
    i. $z_2 = W_l\ .a_1 + b_l$
h. Classify samples into anomalous and normal category using the following equation:
    i. $a_2(z_2) = \text{hardlim}(z_2)$
    ii. $y = a_2(z_2)$

i. For the predicted output $y$, compute the Information Fractal Dimension, $fd_{cur_{anom}}$ of anomalous data labels and find the error in the fractal dimension i.e.

    i. $fd_{err_{anom}} = fd_{ref_{anom}} - fd_{cur_{anom}}$

j. Similarly, compute the Information Fractal Dimension, $fd_{cur_{norm}}$ of only normal data labels and find the error in the fractal dimension i.e.

    i. $fd_{err_{norm}} = fd_{ref_{norm}} - fd_{cur_{norm}}$
```

```
k. Find the total fractal dimension error i.e.
    i.   $fd_{err_{tot}} = fd_{err_{norm}} + fd_{err_{anom}}$
l. Based on the computed errors, update the weights and bias values as follows:
    i.   $delta_{out} = \left[fd_{err_{norm}} \cdot \left(T_{2_{norm}} - y\right)\right] + \; [fd_{err_{anom}} \cdot \left(T_{2_{anom}} - y\right)]$
    ii.  $delta_{in} = (1 - tanh^2(z_1))\,.\; delta_{out}W_l^T$
    iii. $W_l = W_l + (delta_{out}\;.\,a_1(z_1))$
    iv.  $b_l = b_l + delta_{out}$
    v.   $W_{in} = W_{in} + F_{ts}\;.\; delta_{in}^T$
    vi.  $b_{in} = b_{in} + delta_{in}$
m. Repeat until $fd_{err_{norm}}$ and $fd_{err_{anom}}$ is less than 0.001 (stopping criterion).
n. Computer the classification performance metric.
$end\ function$
```

3. Hebbian rule based ANN - traditional and proposed wavelet based cognitive modification

Biologically plausible cognitive phenomenon have been at the heart of the Hebbian theory which itself has stemmed from the notion that "*The Cells that fire together, wire together*" [167]. Basically, the theory serves as the foundational model for understanding the development process in human beings including but not limited to behavioral changes, learning capabilities, information storage and retrieval, and activity skills [168] [169]. In the context of learning, "connectionism" is the basic concept of Hebb's theory which helps human brain generate distinct signals through simultaneous stimulation of neuronal activity. Formally, the Hebb Synapse [170] is stated as "*When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased.*" In other words, the neuronal assembly is highly connected and the signal strength of a cell increases with the continuous activation of another cell which indicates a growth process in both cells. This is a form of associative learning through reinforcement and evolution and takes place due to the highly localized, temporally dependent and spatially connected activity of neurons. To put it simple, the correlation between time dependent activity of neurons leads to different forms of learning [171]. Also, for learning complex system, there could be multiple neuron assemblies that can have local and global connection with each other and within themselves. This theory paved the path for the design of early artificial neural networks.

3.1 Traditional Hebbian rule based ANN

The artificial neural network relies heavily on the error-correcting mechanisms which are responsible for tuning the weights and bias values in order to improve the classification performance iteratively. Basically, it is an optimization problem in which the cost function is minimized such that the estimated output follows the target. This rule in general is known as the "*Delta rule*" or "*Widrow-Hoff rule*" [171] and is mathematically expressed as follows:

$$\Delta \mathrm{w_{ij}(n)} = \ \varepsilon \mathrm{e_j(n)a_i(n)} \tag{19}$$

where $\Delta w_{ij} = w_{ij}(n) - w_{ij}(n+1)$ and is the change in weight from neuron i to neuron j, e_j is the error signal for neuron j, $a_i$is the activation level of neuron *i* at time step *n* and ε is the learning rate. This equation signifies that the weight modification is local as it relies on the error of a particular neuron and the signal input to it. However, the learning rate ε drives the stability and convergence of the iterative process and hence, is a critical parameter in defining the classification performance of the delta rule based ANN. The cognitive rationale behind the delta rule is the ability of humans to learn and correct their behavior, perceptions and actions through errors. One of the theories in this domain compares backpropagation algorithm employed in the ANN with a Hebbian learning called Contrastive Hebbian Learning [172] [173] which is presumed to be the human brain learning strategy. It clamps the output signal to a desired level and let the error signal propagate back to the network.

Human learning continues even in the absence of the error signal through the processing of information from the surroundings. This type of learning resembles the Hebbian rule based ANN which are activated as a result of the neuronal activity triggered by the signal which itself is a weighted sum of activations. The weight adjustment in these networks is dependent on the simultaneous activity of sending and receiving neurons and this rule is mathematically expressed as,

$$\Delta \mathrm{w_{ij}(n)} = \ \varepsilon \mathrm{f\big(a_i(n)\big)f(a_j(n))} \tag{20}$$

It should be noted that the change in weight from neuron i to neuron j is Δw_{ij}, a_i and $a_j$are the activation level of neuron i and neuron j respectively, ε is the learning rate and $f(.)$ is a (non-linear) function. This equation implies that the weight update mechanism is dependent on the nearby neuronal activity thus signifying the concept of "connectionism."

This rule has two fundamental limitations. The first of them is that the weights may grow infinitely large causing the system to become unstable due to the underlying positive feedback loop. Each successive update will cause the weight to increase in the direction of largest eigen value [174]. Also, this model fails to incorporate the supplemental rule of weight decrement. This is possible if the neurons are activated asynchronously on either side of a connection. As a result Oja's rule [175] was introduced which normalized the weight update rule and is a stable form of Hebb rule. Mathematically, it is expressed as,

$$\Delta \mathrm{w_{ij}(n)} = \ \varepsilon \mathrm{f(a_j(n))(f(a_i(n))} - \mathrm{f\Big(a_j(n)\Big)\, w_{ij}(n)} \tag{21}$$

This equation ensures that the large modification in the weight is not equally propagated to the next update and hence ends up in providing soft weight boundary.

3.2 Proposed multiscale Hebbian rule based ANN

A multiscale Hebbian learning algorithm has been proposed. To introduce the notion of multiscale, a dyadic wavelet decomposition technique has been used in combination with the traditional Hebbian rule. Basically, wavelet is a space-scale representation and is used to express a function as a weighted sum of selected basis function. This technique is suitable for the analysis of non-stationary signal as it represents the localized behavior of the signal in terms of time and frequency [121]. Mathematically, it is defined as zero average function which is dilated with a scale parameter a and translated by x_0 and is expressed as follows:

$$\psi_{\mathrm{x_0,a}}\,(\mathrm{x}) = \frac{1}{\sqrt{\mathrm{a}}}\psi\left(\frac{\mathrm{x}-\mathrm{x_0}}{\mathrm{a}}\right) \tag{22}$$

In order to compute the continuous wavelet transform (CWT), the signal is correlated with the wavelet atom, depicting the similarity strength of the signal to the selected wavelet at a particular scale. Formally, in mathematical terms it is defined as,

$$\mathrm{WT}[f(x_0,a)] = \frac{1}{\sqrt{a}}\int_{-\infty}^{\infty} f(x)\,\psi\left(\frac{x-x_0}{a}\right)dx \tag{23}$$

However, in real world the discretized version of the CWT is utilized because of the high computational requirement. The discrete version is used to express the signal in terms of mutually orthogonal set of wavelets which are derived from a scaling function which is orthogonal to discrete translations. The scaling functions are mathematically defined as,

$$\phi_{j,k}\,(t) = 2^{j/2}\phi(2^j t - k) \tag{24}$$

$$\psi_{j,k}\,(t) = 2^{j/2}\psi(2^j t - k) \tag{25}$$

The subscripts *j* and *k* indicate the dilation and translation parameters respectively. The corresponding wavelet coefficients can be obtained using the following equations [176].

$$W_\phi[j_0,k] = \frac{1}{\sqrt{M}}\sum_n x[n]\phi_{j_0,k}[n] \tag{26}$$

$$W_\psi[j,k] = \frac{1}{\sqrt{M}}\sum_n x[n]\psi_{j,k}[n] \tag{27}$$

where $j \geq j_0$. The equations (25) and (26) represent the Approximation and Detail coefficients respectively. Basically, detail coefficient depicts the information difference in estimated function at a scale of 2^{j_0} and 2^j. Whereas, the orthogonal projection of the function on a space V_{2_j} is called the approximation coefficient [120].

The proposed algorithm exploits the information difference that exists between different scales and becomes evident when the function is decomposed in terms of their wavelet basis. Therefore, the first and foremost step is to determine the decomposition levels for the given feature vector which should be a maximum possible value. Using this information, the dataset is then divided into different levels or scales. Then, the corresponding approximation and detail coefficients are extracted for the feature set. In this experiment, dyadic sampling is utilized so the data is down-sampled and up-sampled by a factor of 2. Moreover, it is imperative to obtain the samples for the target data corresponding to a particular scale. The approximation and detail coefficients of the feature set are successively fed to the Hebb rule based learning algorithm and classification performance parameters are computed based on the estimated output. The two metric are then compared to determine which metric gives better result. The estimated output of the coefficient that leads to better classification performance is selected as the output on that particular scale. This process is repeated on multiple scales until finest scale is reached.

3.2.1 Pseudo Code [119]

MATLAB has been used to program the standard and proposed multiscale Hebbian rule based artificial neural network. The corresponding weight and bias adjustment approaches are defined in the pseudo code as follows. The results have been compared with the standard gradient descent based neural network for which MATLAB's neural network toolbox has been employed. Also, 5 x 2 cross validation has been used to address the bias and variance balance problem.

The algorithmic details of standard and proposed Hebbian learning based neural network is as follows:

```
$function$ $\boldsymbol{main}()$

FOR LOOP (5 loops)

// During each of the 5 supervised training and testing iterations, data sets $S_1$and $S_2$ are used alternatively as training and testing data sets. (Comment)

4. Load the input data set $S_{in}$ consisting of four features i.e. duration, total bytes exchanged during a TCP connection, mean inter-arrival time and the total round trip time.
5. Divide the dataset $S_{in}$ into two parts with the ratio of first to second part as 70% and 30%. Label the sets as $S_1$ and $S_2$ respectively.
6. Load the corresponding target data labels $T$.
7. Perform anomaly classification using single scale Hebbian weight update rule based neural network for the training and testing data respectively.
        $\boldsymbol{SingleScaleHebbNN}(S_1, T)$
        $\boldsymbol{SingleScaleHebbNN}(S_2, T)$
8. Perform anomaly classification using multiscale Hebbian weight update rule based neural network for the training and testing data respectively.
        $\boldsymbol{MultiScaleHebbNN}(S_1, T)$
        $\boldsymbol{MultiScaleHebbNN}(S_2, T)$
end FOR loop
$\boldsymbol{end\ main}()$
```

```
$\boldsymbol{function\ SingleScaleHebbNN}\ (Input\ Data\ \boldsymbol{F_{in}}\ , Target\ Data\ \boldsymbol{T})$

a. Initialize a neural network with the input layer of size $n$ and a hidden layer of size $m$. The output layer has a single neuron to classify the samples as normal or anomalous.
b. Initialize the learning rate $\eta$ and the iteration count $iter\_count$.
c. Initialize the input layer weight matrix $W_{in}$ of size $(nxm)$ by computing cross-correlation of $F_{in}$ and a uniform random vector of size $(nx1)$ that carries input bias values $b_{in}$.
d. Using uniform random number generator, initialize hidden layer weight vector $W_l$ of size $(mx1)$ and a scalar output bias value $b_l$.
e. Set the activation function for the neurons in hidden layer as hyperbolic tangent function which takes weighted features $z_1 = W_{in}.F_{in} + b_{in}$ as input and is expressed as follows:
    i. $a_1 = \tanh(z_1)$
f. The output layer neurons classify the weighted hidden layer output samples using following equations:
    i. $z_2 = W_l\,.a_1 + b_l$
    ii. $a_2(z_2) = \text{hardlim}(z_2)$
```

```
        iii. $y = a_2(z_2)$
g. The weight values are updated using the Hebbian rule which are expressed as follows:
        i. $\Delta w_{in} = a_1(F_{in} - w_{in}.a_1)$
        ii. $\Delta w_l = y\ (a_1 - w_{in}.y)$
        iii. $W_l = W_l + \eta\ .\Delta w_l$
        iv. $W_{in} = W_{in} + \eta\ .\Delta w_{in}$
h. Similarly, the bias values are updated using the Hebbian rule which are expressed as follows:
        i. $\Delta b_{in} = a_1(F_{in} - b_{in}.a_1)$
        ii. $\Delta b_l = y\ (a_1 - b_{in}.y)$
        iii. $b_l = b_l + \eta\ .\Delta b_l$
        iv. $b_{in} = b_{in} + \eta\ .\Delta b_{in}$
i. Compute the classification performance metric between the estimated output $y$ and target dataset $T$.
j. Return the performance metric and the estimated output.
$\boldsymbol{end\ function}$
```

```
$\boldsymbol{function\ MultiScaleHebbNN}$ $(Input\ Data\ \boldsymbol{F_{in}}\ , Target\ Data\ \boldsymbol{T})$

1. Determine the levels $level$ to which wavelet decomposition can be performed. This should be selected as the maximum possible decomposition levels.
2. Divide the input dataset $F_{in}$ into multiple scales $level$ by using wavelet decomposition on individual features. Extract the corresponding **approximation** and **detail** coefficients i.e. $appCoef_{in}$ and $detCoef_{in}$.
3. Wavelet decomposition down-samples the data dydacially by 2 and therefore the scale corresponding samples of Target dataset $T_{scale}$ are as well extracted.
4. FOR LOOP ($i$ = 1 → $level$)
      if $i$ == 1
      i. Call the single scale Hebbian learning function with input features $F_{in}$ and target dataset $T$.
                    $[\boldsymbol{perf}, \boldsymbol{output}] = \boldsymbol{SingleScaleHebbNN}(\boldsymbol{F_{in}}, \boldsymbol{T})$
      else
      i. Call the single scale Hebbian learning function with approximation coefficient $appCoef_{in}$ as input features and target dataset $T_{scale}$. $[\boldsymbol{perf_{app}}, \boldsymbol{output_{app}}] = \boldsymbol{SingleScaleHebbNN}(\boldsymbol{appCoef_{in}}, \boldsymbol{T_{scale}})$
      ii. Call the single scale function Hebbian learning with detail coefficient $detCoef_{in}$ as input features and target dataset $T_{scale}$. $[\boldsymbol{perf_{det}}, \boldsymbol{output_{det}}] = \boldsymbol{SingleScaleHebbNN}(\boldsymbol{detCoef_{in}}, \boldsymbol{T_{scale}})$
      iii. If $perf_{app} < perf_{det}$ update the output corresponding to the current scale with $output_{det}$. Otherwise, update it with $output_{app}$.
      end if
   end FOR loop
$\boldsymbol{end\ function}$
```

Chapter VII: Results and discussion

The multiscale analysis extracts hidden complexities which is akin to human cognition aspects of simplifying complex objects to capture hidden patterns. For example, high I.Q. humans can read hidden patterns from a very abstract object. Based on applying three proposed algorithms over highly complex datasets, promising results are obtained in terms of detection performance improvement. The improved classification performance is attributed to the hidden information which is revealed when the analysis is carried out across various different scales. Using the standard Platonic measures, the single scale evaluation of the feature space masks the complexity of the data which often have localized and singular behavior and therefore, is not visible on the one global scale. The results of the standard and proposed learning strategies in the context of cyber threat detection is presented in this chapter. A detailed discussion on the results and the corresponding comparison with the traditional learning algorithms follows as well.

1. Results for traditional and proposed cognitive k-Nearest Neighbour algorithm [103]

The number of neighbours, *k,* for the k-nearest neighbour algorithm was chosen as 3 after testing different values. The rule of thumb is to select it as the square root of the number of features. The number of features in the presented research is 4 and 3 is a fairly suitable choice for the number of neighbours. The classification performance of the algorithms is numerated in Table 5. As evident, it shows obvious performance improvement in the standard k-NN with the fractal based modification. Although, Table 5 depicts that the accuracy, sensitivity and specificity of the two algorithms is comparable and there is a slight enhancement which may not be as noticeable, the precision and F-measure values improved more than 10%. F-measure or the F1-score as known alternately, indicates how accurate an algorithm is in learning the positive class. This is critical for imbalance datasets where the ratio of positive samples to the negative is very low which indicates that learning the pattern of positive samples or specifically, attack instances in the context of cyber-security data is difficult. As explained in the previous chapters this phenomenon is attributed to the advanced morphing and evasion techniques employed by the threat actors to remain undetected while fulfilling their illicit objectives like stealing important strategic information from a local organization or government.

Classification metric	Single Scale Euclidean k-NN	Multiscale Correlation Fractal Dimension k-NN	% Improvement
TPR	92.83 %	93.58 %	0.81%
FPR	6.34 %	5.57 %	12.1%
TNR	93.66 %	94.43 %	0.82%
FNR	7.16 %	6.41 %	10.47%
Accuracy	93.65%	94.42%	0.82%
Sensitivity	92.83%	93.58%	0.81%
Specificity	93.66%	94.43%	0.82%
Precision	13.35%	15.02%	12.51%
F-measure	26.66%	29.99%	12.49%

Table 5: Classification performance k-NN - single scale vs. multiscale - Contagio + IMPACT dataset [103].

It can be observed from the results that k-NN itself performs significantly well for the dataset because of its ability to learn the local behavior instead of generalizing a global optimum. The fractal based cognition has further enabled the algorithm to estimate the class of the instance by re-comparing the correlation based complexity of the cluster before and after the addition of the instance. The fundamental rationale behind this concept is that a sample which belongs to the fractal structure, the synthetically generated dataset, will tend to avoid altering the dimension of the cluster as it belongs to it. However, if the sample is not part of the fractal structure and rather belongs to another fractal, then its addition will modify the dimension greatly. For example, if there are two different fractal structures say, Koch curve and Minkowski triangle and a random sample is picked from any one of them. If the sample belongs to the Koch curve then, addition of the sample in the curve will not change its dimension because of its inherent correlation with the other samples self-similarly. But, if it belongs to Minkowski triangle and is added to the Koch curve it will affect its dimensionality as it is not correlated with other samples in the fractal object. Same concept is applied in this experiment in the context of cluster of anomalous and normal instances, both of whom have different dimensions. Therefore, the notion of scale based evaluation using correlation metric is able to determine the intricate attack patterns which are concealed under the normal packet flows.

Moreover, as indicated by the results, the cyber-security dataset is multifractal in nature as both the anomalous and normal data have different dimension. Further, it is validated by the heavy tailed distribution. This implies that it is a combination of two monofractals and signifies the underlying non-uniform, non-homogenous distribution of data. Basically, this experiment has ascertained the presence of different local measures which form a spectrum of fractal dimensions when expressed as a power-law relationship. Had there not been such a phenomenon then an improvement in the classification performance metric was not possible. However, in order to improve the results, some addition of attributes in the feature vector can be looked into as a future task.

2. Results for traditional (GD based) and proposed (fractal based) Artificial Neural Network [17]

The results for the standard gradient descent based neural network and the proposed fractal based model is shown in Table 6. It is clearly evident that the standard gradient descent based ANN does not perform well as it is able to classify only 50% of the attacks and normal samples correctly giving it an accuracy of about 50%. This implies that the system is essentially making a guess like a tossing coin which has equal probability of heads and tails and is therefore, not suitable to be used with this dataset. On the other, the multiscale analysis of the error curve and the corresponding changes in the backpropagation algorithm has led to a neural network depicted by a reduction in false positive rate and false negative rates.

Moreover, the low F1-score of the traditional ANN clearly signifies that the performance of the algorithm is very weak over the given complex dataset. This can be attributed to the overlapping or inseparable attack and normal samples boundary on a single scale Euclidean space which causes the algorithm to get stuck in a local minima. Contrarily, the introduction of Fractal based ANN drastically improve the F-measure to an average of 60% over the same data. Further, the improvement in accuracy measure is showing promising results indicating that the algorithm is performing well in term of handling bias. Moreover, the precision measure has also increased but, it is still relatively low which is due to the high variance characteristic of dataset. Introduction of higher order moments like skewness and kurtosis may improve the results further and can be taken as an extension of this work. An analysis involving more generalized fractal nature [111] of the traffic pattern is required here.

Classification metric (mean)	**Single Scale Euclidean Gradient Descent Artificial Neural Network**			**Multiscale Information Fractal Dimension Artificial Neural Network**		
	5 x 2 Cross Validation			**5 x 2 Cross Validation**		
	70% Train and 30% Test	**30% Train and 70% Test**	**Mean %**	**70% Train and 30% Test**	**30% Train and 70% Test**	**Mean %**
TPR (%)	50.01	50.03	50.02	97.55	74.43	85.9
FPR (%)	49.99	50.20	50.01	7.64	13.2	10.42
TNR (%)	50	49.97	49.98	92.35	86.79	89.57
FNR (%)	49.98	49.96	49.97	2.44	25.56	14
Precision (%)	54.61	11.06	32.84	56.35	36.11	46.23
Accuracy (%)	50.15	49.91	50.03	92.83	85.67	89.25
F-Measure	0.522	0.181	0.396	0.714	0.486	0.601

Table 6: Classification performance ANN - single scale vs. multiscale - UNSW-NB15 dataset [17]

One may question the reason for the better performance of proposed algorithm and the answer lies in the multiscale analysis. Basically, the normal and attack samples belong to different fractal structures, both having distinct information fractal dimension. This difference is carried over to the computation of error side as well where both the threat and benign samples had different error curves when viewed on multiple scales and thus, have separate dimensions. Exploiting this concept is especially necessary in the context of cyber security data where the data samples literally overlap on a single scale and hence, are difficult to be separated leading to the misclassification. Moreover, the fractal dimension provides a convenient tool to observe the fractional changes in term of the structure and information of the data samples which otherwise is not evident when considering the topological dimension. For example, a Koch curve has a topological dimension of 2 but a fractal dimension of 1.2619 which implies that its complexity lies in between that of a line (1-dimensional) and a surface (2-dimensional). Also, the information content of the attack and benign samples is completely different as both have different Shannon entropy and the power law relationship. In other words, the feature space of the attack and normal samples have different information fractal dimension because of the distinct probability distributions associated with each of them. Therefore, a single scale analysis masks this important aspect and is not able to accurately classify the samples.

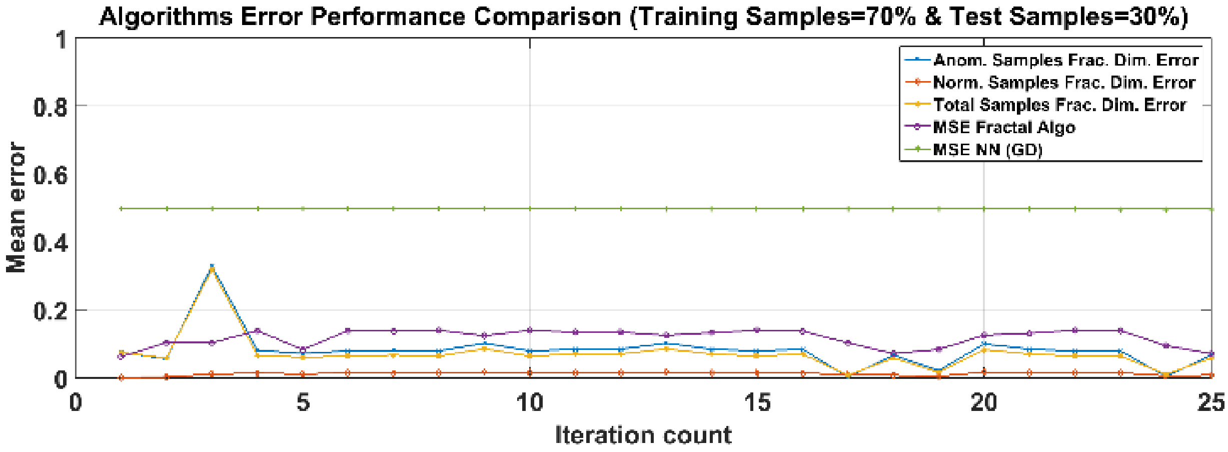


Figure 14: Mean error performance curve (70% Training and 30% Test samples) [17].

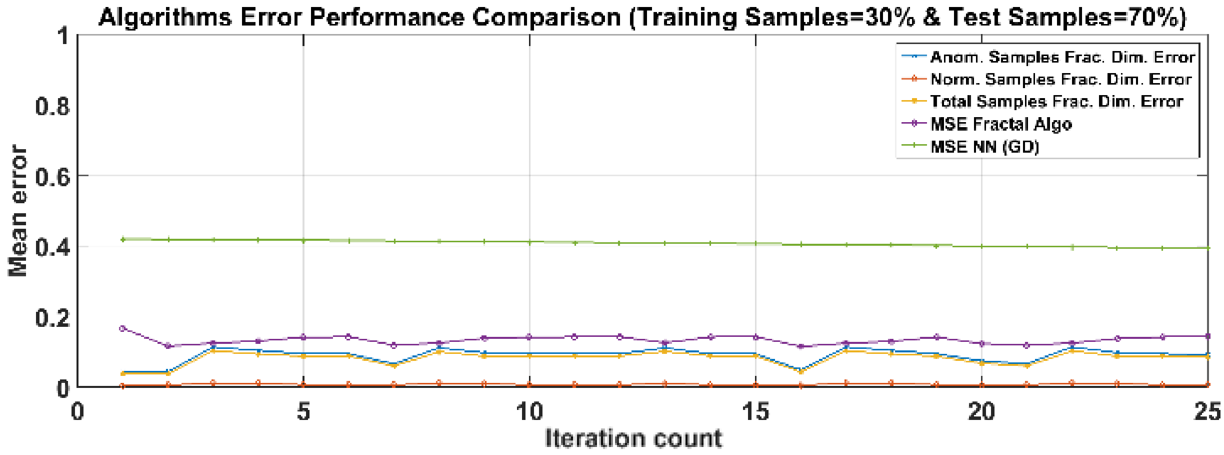


Figure 15: Mean error performance curve (30% Training and 70% Test samples) [17].

In Figure 14 and Figure 15, the error curves also further ascertain the obtained numerical results which indicate that the fractal based modification has resulted in superior performance. This algorithm can work as a baseline for improving existing machine learning algorithms which employ single scale Euclidean measures. Although, multiple improved variants of the backpropagation algorithm are available, the proposed algorithm is compared with a very fundamental gradient descent based ANN because the objective was to determine the performance difference while keeping the learning model simple and fundamental. Addition of any other complexity may blur the analysis of the source of improvement. Nevertheless, the objective was to determine a strategy for the classification of overlapping samples in cyber threat detection and to simultaneously reduce both the false alarms.

The toughest part of applying the machine learning is to obtain a real time internet dataset which is not readily available due to the privacy regulations and the associated storage and assembling difficulties [177]. Therefore, the generalization of the proposed algorithm over any data set cannot be established. However, it can be argued that the technique will work reliably for binary classification of datasets that can be characterized by the inseparability problem on single scale platonic measurements.

Is it possible to project the over-lapping dataset into a higher dimensional space to linearly classify the samples? It can be argued that this approach may not reveal the desired results since it has three major limitations: (i) the curse of dimensionality will make the analysis difficult as the number of features grow, (ii) advanced cyber threats continuously morph their states therefore, if a suitable higher dimension space is found where the samples are linearly separable then it is a matter of few seconds that another higher dimensional space is required for the analysis of the variant of the same attack; and (iii) samples often overlap in a higher dimensional space as was observed during the experiment. Hence, the proposed information fractal dimension based cost function for error estimation provides a convenient and reliable method to extract hidden complex attack patterns without altering the dimensionality of the feature space.

Another important aspect is vulnerability of the proposed method for the cyber threats itself. It refers to the reliability of the proposed model if the threat model, decision boundary, or tampering strategies are altered by the attacker [103]. This risk is generally greater when the system is deployed in a real time environment for the classification of threats like internet gateways. However, the proposed methodology utilizes fractal based analysis which requires large collection of data samples over longer duration and therefore, is not only offline but has ability to detect long range dependence using the concept of probabilistic self-similarity. Hence, the proposed technique has an edge over the traditional based on the experimental results and associated arguments.

3. Results for traditional Hebbian rule and proposed wavelet based multiscale modification [119]

A comparison of classification results between the standard Gradient Descent based neural network, traditional single scale Hebbian learning rule and wavelet based multiscale Hebbian learning rule is enumerated in Table 8. It is clearly evident that GD based ANN and standard Hebbian rule are comparable in terms of their detection accuracy. Basically, both the algorithm have low precision and accuracy depicting the fact that they are unable to handle high bias and high variance in the UNSW-NB 15 dataset. This is further translated into lower F1-score indicating the incapability of the algorithm to fulfill the objective of binary classification. However, GD based ANN is able to provide approximately 50% correct results. Contrarily, Hebbian rule which inherently proposes linear classification boundary has a much lower F-measure of about 10%.

In order to introduce the notion of non-linearity, wavelet based analysis of the input features is performed which is able to determine the complexity of the dataset using scale-space evaluation. Basically, the approximation and detail coefficients are fed to the system successively and the results are compared to choose the estimate. This methodology was proven successful as shown from the results which are improved to a rate of 95% and 73 % for true negative and true positive respectively. Nevertheless, the improvement in the specificity is much more elaborate than that of sensitivity. This can be attributed to the imbalance in the dataset which has a positive to negative samples ratio of 1:9.

Statistical Distributions		**Classification Performance**			
		Mean TPR (%)	**Mean TNR (%)**	**Mean FPR (%)**	**Mean FNR (%)**
Chi^2 Distribution	Single Scale	83.98	16.77	83.22	16.01
	Multi Scale	81.32	72.15	27.85	18.68
Std. Normal Distribution	Single Scale	99.98	1.57	98.43	0.013
	Multi Scale	86.59	55.23	44.77	13.41
Uniform Distribution	Single Scale	70.94	28.50	71.49	29.05
	Multi Scale	75.64	80.03	19.97	24.36
Exponential Distribution	Single Scale	60.53	40.18	59.81	39.46
	Multi Scale	70.21	60.82	39.18	29.79

Table 7: Classification performance Hebbian - single scale vs. multiscale - Statistical distribtuions [119].

Due to the improvement in the specificity and sensitivity values, the precision and accuracy measures of the proposed methodology have increased compare to the GD based ANN and traditional Hebbian rule. Nonetheless, the precision value is lesser than that of accuracy establishing the fact that the algorithm has the better capacity to handle high bias than the high variance problem. A combination of Hebbian rule and an error minimization strategy can be exploited as a future work to further improve the performance.

Moreover, it should also be noted that as the algorithm takes both the coefficients i.e. approximation and detail in consideration while estimating the relevant class, this algorithm can generalize well. This is because no prior assumptions have been made and therefore, the data samples can be resolved depending upon the underlying behavior of the system. Also, this lack of presumption aids the algorithm in determining the local behaviors which vary greatly with the scales or resolution. This is further ascertained by testing the algorithm on various different distributions as mentioned in Table 7. These distributions were generated through MATLAB simulation and were normalized before being fed to the single and multiscale Hebbian learning rule. The results clearly validate the better generalization capacity of the algorithm.

Classification metric (mean)	**Algorithms**		
	Gradient Descent Neural Network	**Single Scale Hebbian Neural Network**	**Wavelet based Multiscale Hebbian Neural Network**
TPR (%)	50.02	44.41	73.55
FPR (%)	50.01	48.47	4.46
TNR (%)	49.98	51.52	95.33
FNR (%)	49.97	55.59	26.44
Precision (%)	32.84	19.53	61.93
Accuracy (%)	50.03	48.10	93.56
F-Measure	0.396	0.103	0.675

Table 8: Classification performance ANN - GD ANN vs. single scale Hebbian vs. multiscale - UNSW-NB15 dataset [119].

Further, it can be observed that the true positive rate has decreased slightly for Chi^2 and normally distributed data samples with the usage of proposed multiscale Hebbian rule. However, the corresponding huge improvement in the true negative rate is promising and signifies that over all the algorithm works well and does not tend to learn one particular pattern rather tries to learn the generalized behavior of both positive and negative samples to achieve better classification performance. In addition, the proposed strategy simultaneously decreases both the false negative and false positive rate which is not achievable through a single scale, linear Hebbian rule. It can be concluded from Table 7 that the detection rate of proposed wavelet based solution lies in between 60%-70% approximately, over a general dataset and hence, is a promising empirical result.

Chapter VIII: Conclusion and future work

A summary of the key contributions of this thesis are as follows:

1. Proposed a multi scale correlation fractal dimension measure as a similarity metric in standard k-Nearest Neighbours supervised algorithm.
2. Proposed a multi scale information fractal dimension based cost function for error estimation in standard Gradient Descent Artificial Neural Network.
3. Proposed a wavelet based multi scale Hebbian ANN.

The threat landscape with all of its diversity and complexity is expanding steadily and swiftly. The attack techniques are not only innovative but, are also simple and straight forward at their core though complex apparently. The latest trend in ensuring the success of an advanced attack is to exploit the combination of a multitude of highly technical and readily accessible services, protocols, software, tools and people. Often several parallel chains of threats are launched, each of which is distinct in terms of attack methodology and objective. The biggest challenge posed by these attacks is that they tend to camouflage their behavior and do not leave a distinct pattern when observed and analyzed on a global scale using a Platonic measurement. Alternatively, the feature space is not only non-linear but consists of overlapping samples of positive and negative classes implying the non-existence of an accurate and precise classification boundary in terms of mono scale. Consequently, the detection of illicit activities using traditional learning algorithms is proving to be futile because of the strong presumption of these algorithms regarding the unique topological dimension of estimated classes on a single scale features space.

While plenty of new and innovative cyber-defence strategies have surfaced in the past few years, threat actors however, seem to stay a step ahead of the cyber realm defenders by evading even state-of-the-art detection systems. A comprehensive analysis of each dataset in this thesis indicated that the internet data follows the heavy tailed distribution and is self-similar in nature. This was due to the presence of many outliers with high variance and high mean. This characteristic is also attributed to the attack behavior which mimic the trajectory of normal internet flows to hide its malevolent activity. The primary step in detecting such threats is careful extraction of feature vector through pre-processing of data based on TCP/IP session information. It involves selecting attributes which help create a meaningful picture when fed to the machine learning system. Therefore, a detailed guideline regarding this critical step is also included in this thesis.

Further, it is required to introduce intelligence which resembles human perception in the existing learning methods. One such method is the multiscale analysis of features, algorithm and error space based on their multiscale probabilistic or morphological complexity. The key idea was to investigate the binary classification performance of machine learning algorithms by instilling cognitive complexity in their decision making engine. This is conceptually analogous to the functioning of human brain which utilizes multiple connected neuronal assemblies to perform complex functions.

In the presented thesis, three different traditional machine learning algorithms are modified by proposing complexity based multiscale analysis in each of them using the mathematical tools of fractals and wavelets. The empirical results are then compared with their traditional counter parts of k-Nearest Neighbours (k-NN), Gradient Descent based Artificial Neural Network (ANN), and Hebbian learning rule and improved detection accuracy is achieved. The algorithms are also tested on imbalance dataset, frequently encountered in cyber realm. The scale based analysis approach revealed the subtle and intricate boundary that exists between the samples of the two overlapping classes and hence, was able to reduce false positives and false negatives simultaneously. Also, all of these were able to handle the high bias problem relatively well which is evident from the improved accuracy measure. It is also expected that the performance of algorithms over a generalized set of data will be comparable to the demonstrated results as long as data follows the trend discussed in the presented research.

The substantial improvement of precision performance which is an indicator of algorithm's capacity to address high variance issue has not been fully achieved and therefore, can be considered as a future work. Moreover, the algorithms proposed in this thesis utilize the concept of Shannon information fractal dimension and correlation fractal dimension. However, other more generalized fractal dimensions like Mandelbrot and Renyi dimension can be modeled to further improve the performance and validate the current findings. Also, the current work utilizes dyadic sampling using mother wavelet which can be further extended to use different forms of sampling and basis function. Another aspect to expand the research is to compare this technique with other machine learning algorithms such as support vector machines, decision tree, and naive Bayes. Similar form of modification can be done in these algorithms to assess their performance and further strengthen the idea of multiscale analysis. The mathematical proofs of the proposed algorithm are another dimension to continue this work in future.

This thesis is expected to function as one of the foundational stone in the context of multiscale analysis in cyber security. A conscious effort has been put in to present a detailed overview of the current cyber-security challenges and the advanced mitigation techniques being employed to detect and further prevent the cyber-attacks. A thorough discussion on the need of cognitive analysis is also included which signifies the two important mathematical tools used for this purpose. Another important aspect of data pre-processing and feature selection is emphasized through examples of synthetic and real world datasets while including a guideline as a reference. The three algorithms along with their detailed pseudo code and motivation is as also included. The results section highlights the classification performance of these results by comparing each with their standard counterpart. Building incrementally, this work introduces basic concept of security in cyber realm and connects it to the intricate concepts of multiscale analysis and its edge over the current state-of-the-art cyber threat detection mechanisms.

References

[1] Marco Barreno, Blaine Nelson, Russell Sears, Anthony D. Joseph and J. D. Tygar, "Can Machine Learning Be Secure?," in *proc. of the ACM Symposium on Information, Computer, and Communication Security*, 2006.

[2] IBM-Research, "AI and Cognitive Computing," IBM Corporation, 2015. [Online]. Available: http://www.research.ibm.com.

[3] Kalyan Veeramachaneni, Ignacio Arnaldo, Vamsi Korrapati, Constantinos Bassias and Ke Li, "AI2 : Training a big data machine to defend," in *proc. of the IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS)*, New York, NY, USA, 2016.

[4] Imperva Inc., "Web Attacks: The Biggest Threat to Your Network," Imperva Inc., 2014. [Online]. Available: https://www.imperva.com.

[5] Hung-Jen Liao, Chun-Hung Richard Lin, Ying-Chih Lin and Kuang-Yuan Tung, "Intrusion detection system: A comprehensive review," *Journal of Network and Computer Applications,* vol. 36, no. 1, pp. 16-24, Jan. 2013.

[6] Muhammad Salman Khan, Sana Siddiqui, Robert D. McLeod, Ken Ferens and Witold Kinsner, "Fractal based adaptive boosting algorithm for cognitive detection of computer malware," in *proc. of IEEE Intl. Conference on Cognitive Informatics and Cognitive Computing (ICCI*CC)*, Stanford University, CA, USA, 2017.

[7] Chirag N. Modi, Dhiren R. Patel, Avi Patel and Muttukrishnan Rajarajan, "Integrating signature apriori based Network Intrusion Detection System (NIDS) in Cloud computing," in *proc. of the 2nd Intl. Conference on Communication, Computing & Security*, 2012.

[8] Chirag Modi, Dhiren Patel, Bhavesh Borisaniya, Hiren Patel, Avi Patel and Muttukrishnan Rajarajan, "A survey of intrusion detection techniques in Cloud," *Journal of Network and Computer Applications,* vol. 36, no. 1, p. 42–57, 2013.

[9] Jing Liu, Yang Xiao, Kaveh Ghaboosi, Hongmei Deng and Jingyuan Zhang, "Botnet: classification, attacks, detection, tracing, and preventive measures," in *proc. of the 4th. Intl. Conference on Innovative Computing, Information and Control*, Dec. 2009.

[10] Pat Belcher, "Hash Factory: New Cerber Ransomware Morphs Every 15 Seconds," Jun. 2016. [Online]. Available: https://www.invincea.com.

[11] Leonid Kalinichenko, Ivan Shanin and Ilia Taraban, "Methods for anomaly detection: A survey," in *proc. of the 16th All-Russian Conference "Digital Libraries: Advanced Methods and Technologies, Digital Collections-RCDL-2014*, Dubna, Russia, Oct. 2014.

[12] Monowar H. Bhuyan, D. K. Bhattacharyya and J. K. Kalita, "Survey on incremental approaches for network anomaly detection," in *International Journal of Communication Networks and Information Security (IJCNIS)*, 2011.

[13] Jamil Farshchi, "Statistical-Based Intrusion Detection," Symantec Inc. USA, Apr. 2003.

[14] K. Parvathi Devi and Y. A. Siva Prasad, "Study of Anomaly Identification Techniques in Large Scale Systems," *International Journal of Computer Trends and Technology,* vol. 3, no. 1, 2012.

[15] James Cannady and Jay Harrell, "A comparative analysis of current intrusion detection technologies," in *proc. of the 4th. Technology for Information Security Conference*, 1996.

[16] A. Pasupulati, J. Coit, K. Levitt, S. F. Wu, S. H. Li, J. C. Kuo and K. P. Fan, "Buttercup: On Network-based Detection of Polymorphic Buffer Overflow Vulnerabilities"," in *proc. of IEEE/IFIP Network Operations and Management Symposium*, Seoul, South Korea, Apr. 2004.

[17] Sana Siddiqui, Muhammad Salman Khan, Ken Ferens and Witold Kinsner, "Fractal based cognitive neural network to detect obfuscated and indistinguishable Internet threats," in *proc. of IEEE Intl. Conference on Cognitive Informatics and Cognitive Computing (ICCI*CC)*, Jul. 2017.

[18] Sana Siddiqui, Muhammad Salman Khan and Ken Ferens, "Cognitive computing and multiscale analysis for cyber security," in *Computer and Network Security Essentials Book*, Springer International Publishing AG, 2017.

[19] Anna Sperotto, Ramin Sadre and Aiko Pras, "Anomaly characterization in flow-based traffic time series," in *Lecture Notes in Computer Science, IP Operations and Management*, 2008.

[20] Marius Kloft, Ulf Brefeld, Patrick Düssel, Christian Gehl and Pavel Laskov, "Automatic feature selection for anomaly detection," in *proc. of the 1st ACM workshop on Workshop on AISec*, Alexandria, Virginia, USA, Oct. 2008.

[21] Ignasi Paredes-Oliva, Ismael Castell-Uroz, Pere Barlet-Ros, Xenofontas A. Dimitropoulos and Josep Sole-Pareta , "Practical anomaly detection based on classifying frequent traffic patterns," in *proc. of IEEE Conference on Computer Communications Workshops*, Orlando, FL, USA, Mar. 2012.

[22] Taeshik Shon, Yongdae Kim, Cheolwon Lee and Jongsub Moon, "A machine learning framework for network anomaly detection using SVM and GA," in *proc. of the 6th Annual IEEE SMC Information Assurance Workshop*, West Point, NY, USA, Jun. 2005.

[23] Brad Miller, Ling Huang, A. D. Joseph and J. D. Tygar, "I know why you went to the clinic: Risks and realization of HTTPS traffic analysis," in *proc. of Intl. Symposium on Privacy Enhancing Technologies Symposium*, 2014.

[24] Przemysław Berezinski, Jozef Pawelec, Marek Małowidzki and Rafał Piotrowski, "Entropy-based Internet traffic anomaly detection: A case study," in *proc. of 9th Intl. Conference on Dependability and Complex Systems, Advances in Intelligent Systems and Computing*, Brunow, Poland, Jul. 2014.

[25] George Nychis, Vyas Sekar, David G. Andersen, Hyong Kim and Hui Zhang, "An empirical evaluation of entropy-based traffic anomaly detection," in *proc. of the 8th ACM SIGCOMM conference on Internet measurement*, Greece, Oct. 2008.

[26] Thuy T.T. Nguyen and Grenville Armitage, "A survey of techniques for internet traffic classification using machine learning," *IEEE Communications Surveys and Tutorials,* vol. 10, no. 4, pp. 56-76, 2008.

[27] Jake Ryan, Meng-Jang Lin and Risto Miikkulainen, "Intrusion detection with neural networks," in *proc. of the 1997 Conference on Advances in Neural Information Processing Systems*, Denver, Colorado, USA , 1998.

[28] Jimmy Shun and Heidar A. Malki, "Network intrusion detection system using neural networks," in *proc. of IEEE 4th. Intl. Conference on Natural Computation*, Jinan, China, Oct. 2008.

[29] Richard P. Lippmann and Robert K. Cunningham, "Improving intrusion detection performance using keyword selection and neural networks," *The International Journal of Computer and Telecommunications Networking - Special Issue on Recent Advances in Intrusion Detection Systems,* vol. 34, no. 4, pp. 597 - 603, Oct. 2000.

[30] James Cannady, "Artificial neural networks for misuse detection," in *proc. of the National Information Systems Security Conference (NISSC'98)*, Oct. 1998.

[31] Vu N. P. Dao and Rao Vemuri, "A performance comparison of different back propagation neural networks methods in computer network intrusion detection," *Differential Equations and Dynamical Systems,* vol. 10, no. 1&2, pp. 201-214, 2002.

[32] Ashish Chandra, Mohammad Suaib and Rizwan Beg, "Web spam classification using supervised artifiical neural network algorithms," *Advanced Computational Intelligence: An International Journal (ACII),* vol. 2, no. 1, 2015.

[33] Vicente Alarcon-Aquino, J. A. Mejia-Sanchez, Roberto Rosas-Romero and J. F. Ramirez-Cruz, "Detecting and classifying attacks in computer networks using feed-forward and Elman neural networks," in *proc. of the 1st. European Conference on Computer Network Defence*, Wales, UK, 2006.

[34] Justin Ma, Lawrence K. Saul, Stefan Savage and Geoffrey M. Voelker , "Learning to detect malicious URLs," *ACM Transactions on Intelligent Systems and Technology (TIST),* vol. 2, no. 3, Apr. 2011.

[35] Yanheng Liu, Daxin Tian and Bin Li, "A wireless Intrusion Detection Method based on dynamic growing neural network," in *proc. of IEEE 1st. Intl. Multi-Symposiums on Computer and Computational Sciences*, Hanzhou, Zhejiang, China, Jun. 2006.

[36] Daxin Tian and Yang Xiang, "A multi-core supported intrusion detection system," in *proc. of Intl. Conference on Network and Parallel Computing*, Shanghai, China, Oct. 2008.

[37] Daxin Tian, Yanheng Liu and Bin Li, "A distributed Hebb neural network for network anomaly detection," in *proc. of Intl. Symposium on Parallel and Distributed Processing and Applications (ISPA 2007)*, 2007.

[38] Daniel Barbara and Ping Chen, "Using the fractal dimension to cluster datasets," in *proc. of the 6th. ACM SIGKDD Intl. Conference on Knowledge Discovery and Data Mining*, Boston, MA, USA, Aug. 2000.

[39] Yulios Zavala, Jeferson Wilian de Godoy Stenico and Lee Luan Ling, "Internet traffic classification using multifractal analysis approach," in *proc. of the 15th Communications and Networking Simulation Symposium*, San Diego, CA, USA, Mar. 2012.

[40] Ruoyu Yan and Yingfeng Wang, "Hurst parameter for security evaluation of LAN traffic," *Information Technology Journal,* vol. 11, no. 2, pp. 269-275, 2012.

[41] Seyed Mahmoud Anisheh and Hamid Hassanpour, "Designing an approach for network traffic anomaly detection," *International Journal of Computer Applications,* vol. 37, no. 3, 2012.

[42] Roderick Murray-Smith, "A fractal radial basis function neural net for modelling," in *proc. of Intl. Conference on Automation, Robotics and Computer Vision*, Singapore, 1992.

[43] Li Zhao, Weidong Li, Liqing Geng and Yanzhen Ma, "Artificial neural networks based on fractal growth," in *Advances in Automation and Robotics - Lecture Notes in Electrical Engineering*, Springer, Berlin, Heidelberg, 2011, pp. 323-330.

[44] Erhard Bieberich, "Recurrent fractal neural networks: a strategy for the exchange of local and global information processing in the brain," *Biosystems,* vol. 66, no. 3, pp. 145-164, Aug. 2002.

[45] Eung-Soo Kim, Masaki San and Yasuji Sawada, "Fractal Neural Network: Computational performance as an associative memory," *Progress of Theoretical Physics,* vol. 89, no. 5, pp. 965-972, May 1993.

[46] Kevin Haley and Paul Wood, "Internet Security Threat Report," Symantec Corporation, Apr. 2016.

[47] Louis Marinos, Adrian Belmonte and Evangelos Rekleitis, "ENISA Threat landscape 2015," Greece, Jan. 2016.

[48] Douglas Adams, "Checkpoint Security Report," Check Point Software Technologies Ltd., 2016.

[49] Scott Hilton, "Dyn Analysis Summary of Friday October 21 Attack," Dyn - Oracle Corporation, Oct. 2016.

[50] Kevin Townsend, "Reports Outline Current Threat Landscape," Security Week - Wired Business Media, 21 Sep. 2016. [Online]. Available: http://www.securityweek.com.

[51] Lee Neely, "Exploits at the Endpoint: SANS 2016 Threat Landscape Survey," SANS Institute, Aug. 3, 2016.

[52] Andra Zaharia, "10 Alarming Cyber Security Facts that Threaten Your Data," Heimdal Security, 12 May 2016. [Online]. Available: https://heimdalsecurity.com.

[53] IBM Security, "Reviewing a year of serious data breaches, major attacks and new vulnerabilities-Analysis of cyber attack and incident data from IBM's worldwide security services operations," IBM X-Force Research, 2016.

[54] Jessica Davis, "7 largest breaches of 2015," Healthcare IT News, 11 Dec. 2015. [Online]. Available: http://www.healthcareitnews.com.

[55] Dan Lohrmann, "The Top 17 Security Predictions for 2017," Fireye, Jan. 8, 2017.

[56] Chris Richter, "Safeguarding the Internet - Level 3 Botnet Research Report," Level 3 Communications, LLC., Jun. 10, 2015.

[57] Imperva Inc., "2015 Web Application Attack Report (WAAR)," Imperva Inc., Nov. 2015.

[58] Darya Gudkova, Maria Vergelis, Nadezhda Demidova and Tatyana Shcherbakova, "Spam and Phishing in Q2 2016," Kaspersky Lab, 2016.

[59] IBM Security, "IBM X-Force Threat Intelligence Quarterly, 2Q 2015," IBM Security Systems, 2015.

[60] Eric M. Hutchins, Michael J. Clopperty and Rohan M. Amin, "Intelligence-Driven Computer Network Defence Informed by Analysis of Adversary Campaigns and Intrusion

Kill Chains," in *proc. of the 6th Intl. Conference on Information Warfare and Security*, Washington, DC, USA, Mar. 2011.

[61] J. Vijaya Chandra, Narasimham Challa and Sai Kiran Pasupuleti, "Advanced persistent threat defense system using self-destructive mechanism for cloud security," in *proc. of IEEE Intl. Conference on Engineering and Technology (ICETECH)*, Mar. 2016.

[62] NTT Group Security, "2016 NTT Group - Global Threat Intelligence Report," 2016.

[63] Stefan Achleitner, Thomas La Porta, Patrick McDaniel, Shridatt Sugrim, Srikanth V. Krishnamurthy and Ritu Chadha, "Cyber Deception: Virtual networks to defend insider reconnaissance," in *proc. of the 8th ACM CCS Intl. Workshop on Managing Insider Security Threats*, Oct. 2016.

[64] Stephen Cobb and Andrew Lee, "Malware is called malicious for a reason: The risks of weaponizing code," in *proc. of the 6th IEEE Intl. Conference on Cyber Conflict (CyCon 2014)*, Oct. 2014.

[65] Alexandr Rivlin, Divyesh Mehra, Henry Uyeno and Vinay Pidathala, "System and method of detecting delivery of malware using cross-customer data". USA Patent US9363280 B1, 7 Jun. 2016.

[66] Alina Oprea, Zhou Li and Ting-Fang Yen, "Detection of early-stage enterprise infection by mining large-scale log data," in *proc. of the 45th Annual IEEE/IFIP Intl. Conference on Dependable Systems and Networks (DSN)*, Jun. 2015.

[67] Aditya K Sood and Richard J Enbody, "Malvertising – Exploiting web advertising," *Computer Fraud & Security,* vol. 2011, no. 4, p. 11–16, 2011.

[68] Steven Van Acker, Nick Nikiforakis, Lieven Desmet, Frank Piessens and Wouter Joosen, "Monkey-in-the-browser: malware and vulnerabilities in augmented browsing script market," in *proc. of the 9th ACM Symposium on Information, Computer and Communications Security*, Jun. 2014.

[69] Ameya Sanzgiri and Dipankar Dasgupta, "Classification of insider threat detection techniques," in *proc. of the 11th Annual Cyber and Information Security Research Conference*, 2016.

[70] Hon Lau, "Trojan.Dropper," Symantec Corporation, 2012 . [Online]. Available: https://www.symantec.com/.

[71] Bum Jun Kwon, Jayanta Mondal, Jiyong Jang, Leyla Bilge and Tudor Dumitras, "The Dropper Effect: Insights into malware distribution with downloader graph analytics," in *proc. of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, Oct. 2015.

[72] Websense, "The seven stages of advanced threats - Understanding the cyber-attack kill chain," Websense, Inc., 2013.

[73] Zhou Li, Sumayah Alrwais, Yinglian Xie, Fang Yu and XiaoFeng Wang, "Finding the linchpins of the Dark Web: A study on topologically dedicated hosts on malicious web infrastructures," in *proc. of the 2013 IEEE Symposium on Security and Privacy (S&P)*, May 2013.

[74] NIST - US Department of Commerce, "National Vulnerability Database," DHS/NCCIC/US-CERT, [Online]. Available: https://nvd.nist.gov/. [Accessed 29 12 2016].

[75] Yiyu Yao, Ron Sun, Tomaso Poggio, Jiming Liu, Ning Zhong and Jimmy Huang, Eds., Lecture Notes in Artificial Intelligence - Brain Informatics, vol. 6334, 2010.

[76] Yingxu Wang , George Baciu, Yiyu Yao, Witold Kinsner, Keith Chan, Bo Zhang, Stuart Hameroff, Ning Zhong, Chu-Ren Hunag, Ben Goertzel, Duoqian Miao, Kenji Sugawara, Guoyin Wang, Jane You, Du Zhang and Haibin Zhu, "Perspectives on cognitive informatics and cognitive computing," *International Journal of Cognitive Informatics and Natural Intelligence,* vol. 4, no. 1, pp. 1-29, 2010.

[77] Yingxu Wang, Ying Wang, Shushma Patel and Dilip Patel, "A Layered Reference Model of the Brain (LRMB)," *IEEE Transactions on Systems, Man, and Cybernetics—PART C: Applications and Reviews,* vol. 36, no. 2, pp. 124-133, Mar. 2006.

[78] Seong-Whan Lee, Heinrich H. Bulthoff and Klaus-Robert Muller , Eds., Trends in Augmentation of Human Performance - Recent Progress in Brain and Cognitive Engineering, Springer, 2015.

[79] Yingxu Wang and Ying Wang, "Cognitive Informatics Models of the Brain," vol. 36, no. 2, pp. 203-207, 2006.

[80] Henri Cohen and Claire Lefebvre, Eds., Handbook of categorization in cognitive science, Elsevier Science, 2005, pp. 141-163.

[81] Matthias Rauterberg, "A method of a quantitative measurement of cognitive complexity," in *Human-Computer Interaction: Tasks and Organisation*, 1992.

[82] Charles H. Bennet, "How to define complexity in physics, and why," *Journal of complexity, entropy, and the physics of information,* vol. 8, pp. 137-148, 1990.

[83] Lourdes Mattos Brasil, Fernando Mendes de Azevedo, Jorge Muniz Barreto and Monique Noirhomme-Fraiture, "Complexity and cognitive computing," in *proc. of 11th Intl. Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, Jul. 1998.

[84] Witold Kinsner, "Complexity and its measures in cognitive and other complex systems," in *proc. of IEEE Intl. Conference on Cognitive Informatics and Cognitive Computing (ICCI*CC)*, Aug. 2008.

[85] Bruce Edmonds, "Syntactic measures of complexity," University of Manchester,, Manchester, UK, 1999.

[86] Witold Kinsner, "System complexity and its measures: How complex is complex.," *Advances in Cognitive Informatics and Cognitive Computing Studies in Computational Intelligence,* vol. 323, pp. 265-295, 2010.

[87] Sam Thielman, "Yahoo hack: 1bn accounts compromised by biggest data breach in history," The Guardian, 15 Dec. 2016. [Online]. Available: https://www.theguardian.com.

[88] Vindu Goel and Nicole Perlroth, "Yahoo says 1 billion user accounts were hacked," Dec. 2016. [Online]. Available: http://www.nytimes.com.

[89] Martin Ussath, David Jaeger, Feng Cheng and Christoph Meinel, "Advanced persistent threats: Behind the scenes," in *proc. of IEEE 2016 Annual Conference on Information Science and Systems (CISS)*, Mar. 2016.

[90] Dell Secureworks, "Understand the Threat," 2014. [Online]. Available: http://www.secureworks.com/.

[91] Tim Greene , "Why the 'cyber kill chain' needs an upgrade," Aug. 2016. [Online]. Available: http://www.networkworld.com.

[92] Marc Laliberte, "A New Take on The Cyber Kill Chain," Sep. 2016. [Online].

[93] Jassim Happa and Graham Fairclough, "A model to facilitate discussions about cyber attacks," in *Ethics and Policies for Cyber Operations*, vol. 124, Mariarosaria Taddeo and Ludovica Glorioso, Eds., Springer International Publishing, Dec. 2016, pp. 169-185.

[94] Bhavani Thuraisingham, Murat Kantarcioglu, Kevin Hamlen, Latifur Khan, Tim Finin, Anupam Joshi, Tim Oates and Elisa Bertino, "A data driven approach for the science of cyber security: Challenges and directions," in *proc. of 17th IEEE Intl. Conference on Information Reuse and Integration*, Jul. 2016.

[95] Muhammad Salman Khan, Sana Siddiqui and Ken Ferens, "A cognitive and concurrent cyber kill chain model," in *Computer and Network Security Essentials Book*, Springer International Publishing AG, 2017.

[96] Ron Kohavi and George H. John, "Wrappers for feature subset selection," *Artifical Intelligence Journal,* vol. 97, no. 1-2, pp. 273-324, 1997.

[97] Ron Kohavi, "Feature subset selection as search with probabilistic estimates," *AAAI Fall Symposium on Relevance,* vol. 224, 1994.

[98] Moses Charikar, Venkatesan Guruswami, Ravi Kumar, Sridhar Rajagopalan and Amit Sahai, "Combinatorial feature selection problems," in *proc. of 41st Annual Symposium on Foundations of Computer Science,* Redondo Beach, CA, USA, 2000.

[99] Jose Bins and Bruce A. Draper, "Feature selection from huge feature sets," in *proc. of 8th IEEE Intl. Conference on Computer Vision*, Vancouver, BC, 2001.

[100] Edoardo Amaldi and Viggo Kann, "On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems," *Theoretical Computer Science,* vol. 209, no. 1-2, pp. 237-260, 6 12 1998.

[101] Avrim Blum and Ronald L. Rivest, "Training a 3-node neural network is NP-complete," *Neural Networks,* vol. 5, pp. 117-127, 1992.

[102] Muhammad Salman Khan, Sana Siddiqui and Ken Ferens, "Cognitive modeling of polymorphic malware using fractal based semantic characterization," in *proc. of IEEE Intl. Conference on Technologies for Homeland Security (HST)*, Waltham, MA, USA, Apr. 2017.

[103] Sana Siddiqui, Muhammad Salman Khan, Ken Ferens and Witold Kinsner, "Detecting Advanced Persistent Threats using fractal dimension based machine learning classification," in *proc. of the 2016 ACM Intl. Workshop on Security And Privacy Analytics (IWSPA)*, New Orleans, Louisiana, USA, Mar. 2016.

[104] Witold Kinsner, "A unified approach to fractal dimensions," *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI),* vol. 1, no. 4, pp. 26-47, 2007.

[105] Benoit B. Mandelbrot, "How long is the coast of Britain?," *Science,* vol. 156, pp. 636-638, 1967.

[106] Benoit B. Mandelbrot, Fractals, Form, Chance and Dimension, Freeman, 1977.

[107] Robert L. Devaney, Chaos, Fractals, and Dynamics: Computer Experiments in Mathematics, Addison-Wesley, 1990.

[108] Saipraneeth Gouravaraju and Ranjan Ganguli, "Estimating the Hausdorff–Besicovitch dimension of boundary of basin of attraction in helicopter trim," *Applied Mathematics and Computation,* vol. 218, no. 21, p. 10435–10442, 2012.

[109] Muhammad Salman Khan, Ken Ferens and Witold Kinsner, "A polyscale based autonomous sliding window algorithm for cognitive machine classification of malicious

Internet traffic," in *proc. of Intl. Conference on Security and Management (SAM'15), WordComp`15*, Nevada, USA, 2015.

[110] Zhenghua Shu, Guodong Liu and Ying Xiong, "Signal processing on heart rate signals using the wavelet-based contourlet transform and multifractal," in *proc. of 8th IEEE International Congress on Image and Signal Processing (CISP)*, Shenyangm, China, 2015.

[111] Muhammad Salman Khan, Ken Ferens and Witold Kinsner, "Multifractal singularity spectrum for cognitive cyber defence in internet time series," *International Journal of Software Science and Computational Intelligence (IJSSCI),* vol. 7, no. 3, pp. 17-45, 2015.

[112] Muhammad Salman Khan, Sana Siddiqui, Ken Ferens and Witold Kinsner, "Spectral Fractal Dimension Trajectory (SFDT) to measure complexity of malicious attacks," in *proc. of the Intl. Conference on Security and Management (SAM'16), WorldComp'16*, Nevada, USA, 2016.

[113] Jan H. Houtveen and Peter C. M. Molenaar, "Comparison between the Fourier and Wavelet methods of spectral analysis applied to stationary and nonstationary heart period data," *Psychophysiology,* vol. 38, no. 5, pp. 729-735, 2001.

[114] Piotr Fryzlewicz, Sebastien Van Bellegem and Rainer von Sachs, "Forecasting non-stationary time series by wavelet process modelling," *Annals of the Institute of Statistical Mathematics,* vol. 55, no. 4, pp. 737-764, 2003.

[115] Alain Damlamian and Stephane Jaffard, Wavelet Methods in Mathematical Analysis and Engineering, World Scientific, 2010.

[116] Brij Gupta, Dharma P. Agrawal and Shingo Yamaguchi , Handbook of Research on Modern Cryptographic Solutions for Computer and Cyber Security, IGI Global, May 2016.

[117] Amine Boukhtouta, Serguei A. Mokhov, Nour-Eddine Lakhdari, Mourad Debbabi and Joey Paquet, "Network malware classification comparison using DPI and flow packet headers," *Journal of Computer Virology and Hacking Techniques,* vol. 12, no. 2, p. 69–100, May 2016.

[118] Soo-Yeon Ji, Bong-Keun Jeong, Seonho Choi and Dong Hyun Jeong, "A multi-level intrusion detection method for abnormal network behaviors," *Journal of Network and Computer Applications,* vol. 62, p. 9–17, Feb 2016.

[119] Sana Siddiqui, Muhammad Salman Khan and Ken Ferens, "Multiscale Hebbian neural network for cyber threat detection," in *proc. of 2017 IEEE Intl. Joint Conference on Neural Networks (IJCNN)*, May 2017.

[120] Stephane Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 11, no. 7, pp. 674-693, Jul. 1989.

[121] Stephane Mallat, in *A wavelet tour of signal processing: The sparse way*, 3 ed., Academic Press, 2009.

[122] IMPACT Cyber Trust, "DARPA Scalable Network Monitoring (SNM) Program Traffic," 2009.

[123] Mila Parkour, "Contagio malware database," 2017. [Online]. Available: http://contagiodump.blogspot.ca/.

[124] Deana Shick and Angela Horneman , "Investigating Advanced Persistent Threat 1 (APT1)," Research Showcase, Carnegie Mellon University, May 2014.

[125] Mike Auty, "Anatomy of an advanced persistent threat," *Journal of Network Security,* vol. 2015, no. 4, pp. 13-16, Apr. 2015.

[126] McAfee Inc., "Combating Advanced Persistent Threats- How to prevent, detect, and remediate APTs," McAfee Inc., 2011.

[127] Beth E. Binde, Russ McRee and Terrence J. O'Connor, "Assessing Outbound Traffic to Uncover Advanced Persistent Threat - Joint Written Project," 2011.

[128] Wireshark, "https://www.wireshark.org/docs/man-pages/tshark.html," Wireshark, 2015. [Online].

[129] Nirwan Ansari and Amey Bhaskar Shevtekar, "Proactive test-based differentiation method and system to mitigate low rate DoS attacks". USA Patent US8392991 B2, 5 March 2013.

[130] Nour Moustafa and Jill Slay, "ADFA-NB15-Datasets - UNSW-NB15 Network Packets and Flows Captures," Cyber Range Lab of the Australian Centre for Cyber Security, University of New South Wales, New South Wales, Australia, 2014.

[131] Nour Moustafa and Jill Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *proc. of IEEE Military Communications and Information Systems Conference (MilCIS)*, Canberra, 2015.

[132] Nour Moustafa and Jill Slay, "The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set," *Information Security Journal: A Global Perspective,* pp. 1-14, 2016.

[133] Jessica DeCianno, "Indicators of Attack vs. Indicators of Compromise," CrowdStrike, 09 Dec 2014. [Online]. Available: https://www.crowdstrike.com.

[134] Antanas Verikas and Marija Bacauskiene, "Feature selection with neural networks," *Pattern Recognition Letters,* vol. 23, no. 11, p. 1323–1335, 2002.

[135] Maysam Toghraee, Mohammad Esmaeili and Hamid Parvin, "Evaluation Neural Networks on Selected Feature by Meta Heuristic Algorithms," *Artifical Intelligent Systems and Machine Learning,* vol. 8, no. 3, 2016.

[136] Karim Mohammed Rezaul and Vic Grout, "An approach for characterising heavy-tailed internet traffic based on EDF statistics," *Intelligent Engineering Systems and Computational Cybernetics,* pp. 173-184, 2009.

[137] Reginald D. Smith, "The dynamics of internet traffic: self-similarity, self-organization, and complex phenomena," *Journal of Advances in Complex Systems,* vol. 14, no. 905, 2011.

[138] Jayakrishnan Nair, Adam Wierman and Bert Zwart, "The fundamentals of heavy-tails: properties, emergence, and identification," in *proc. of the ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '13)*, Jun 2013.

[139] David Heath, Sidney Resnick and Gennady Samorodnitsky, "Heavy tails and long range dependence in on/off processes and associated fluid models," *Mathematics of Operations Research,* vol. 23, no. 1, pp. 145-165, 1998.

[140] Tom Mitchell, Machine Learning, McGraw Hill, 1997.

[141] Olga Veksler, "Machine learning in computer vision (Fall 2015 Lecture 2)," Dept. of Computer Science, University of Western Ontario, 2015. [Online]. Available: www.csd.uwo.ca.

[142] David W. Aha, Dennis Kibler and Marc K. Albert, "Instance-based learning algorithms," *Machine Learning,* vol. 6, no. 1, pp. 37-66, 1991.

[143] J. Zico Kolter and Marcus A. Maloof, "Learning to detect and classify malicious executables in the wild," *Journal of Machine Learning Research,* vol. 7, pp. 2721-2744, 2006.

[144] Nenad Tomasev and Krisztian Buza, "Hubness-aware kNN classification of high-dimensional data in presence of label noise," *Neurocomputing,* vol. 160, p. 157–172, Feb. 2015.

[145] Ugur Demiryurek, Farnoush Banaei-Kashani and Cyrus Shahabi, "Efficient k-nearest neighbor search in time-dependent spatial networks," in *proc. of 21st Intl. Conference on Database and Expert Systems Applications: Part I*, Bilbao, Spain, 2010.

[146] Youngki Park, Sungchan Park, Sang-goo Lee and Woosung Jung, "Greedy filtering: A scalable algorithm for K-Nearest Neighbor graph construction," in *proc. of 19th Intl. Conference Database Systems for Advanced Applications-Part I*, Bali, Indonesia, 2014.

[147] Witold Kinsner , "It's time for multiscale analysis and synthesis in cognitive systems," in *proc. of IEEE Intl. Conference on Cognitive Informatics and Cognitive Computing (ICCI*CC)*, Banff, AB, 2011.

[148] Maureen Caudill, "Neural Network Primer: Part I," *AI Expert,* 1989.

[149] Nitin Patel, "Graduate Level Course on Data Mining (MIT Course Number 15.062) - Lecture 6 Artificial Neural Networks," MIT Open Coursewre, Spring 2013.

[150] Raul Rojas, Neural Networks: A Systematic Introduction, Berlin, New-York: Springer-Verlag, 1996 .

[151] Daniel Shiffman, The Nature of Code: Simulating Natural Systems with Processing, 1 ed., 2012.

[152] Hartmut Bohnacker, Benedikt Gross, Julia Laub and Claudius Lazzeroni, Generative Design: Visualize, Program, and Create with Processing, Princeton Architectural Press, 2012.

[153] The MathWorks, Inc., "Transfer Function Graphs in Matlab".

[154] Yann Le Cun, Ido Kanter and Sara A. Sona , "Second-order properties of error surfaces: learning time and generalization," *Advances in Neural Information Processing Systems,* vol. 3, pp. 918-924, 1991.

[155] Yann LeCun, Leon Bottou, Genevieve B. Orr and Klaus-Robert Muller, "Efficient BackProp," *Neural Networks: Tricks of the Trade,* pp. 9-50, 1998.

[156] Daniel Graupe, Principles of Artificial Neural Networks, 3 ed., vol. 7, World Scientific Publishing Co. Pte. Ltd..

[157] A. T. Kalkisim, A. S. Hasiloglu and K. Bilen , "The comparison of performance by using alternative refrigerant R152a in automobile climate system with different artificial neural network models," *Journal of Physics,* vol. 707, no. 1.

[158] Yves Chauvin and David E. Rumelhart, Eds., Backpropagation: Theory, Architectures, and Applications (Developments in Connectionist Theory Series), Psychology Press, 2013.

[159] Igor Aizenberg, Leonid Sheremetov, Luis Villa-Vargas and Jorge Martinez-Munoz, "Multilayer neural network with multi-valued neurons in time series forecasting of oil production," *Neurocomputing,* vol. 175 (B), p. 980–989, 2016.

[160] Paugam-Moisy and Andre Elisseeff Helene, "Size of multilayer networks for exact learning: analytic approach," *Advances in Neural Information Processing Systems,* vol. 9, no. 162, 1997.

[161] Jeff Heaton, Introduction to Neural Networks for Java, Heaton Research Inc., 2008.

[162] Zhi-Hua Zhou and Xu-Ying Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *IEEE Transactions on Knowledge and Data Engineering,* vol. 18, no. 1, pp. 63-77, 2006.

[163] Salvador Garcia and Francisco Herrera, "Evolutionary undersampling for classification with imbalanced datasets: Proposals and taxonomy," *Evolutionary Computation,* vol. 17, no. 3, p. 275–306, 2009.

[164] Lizhi Peng, Haibo Zhang, Bo Yang, Yuehui Chen and Xiaoqing Zhou, "Early Stage Internet Traffic Identification Using Data Gravitation Based Classification," in *proc. of IEEE 14th Intl. Conference on Dependable, Autonomic and Secure Computing*, 2016.

[165] Martin T. Hagan, Howard B. Demuth, Mark H. Beale and Orlando De Jesus, Neural Network Design, 2 ed., Martin T. Hagan, 2014.

[166] Tobias Wuchner, Martin Ochoa and Alexander Pretschner, "Robust and Effective Malware Detection Through Quantitative Data Flow Graph Metrics," in *proc. of Intl. Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, 2015.

[167] Siegrid Lowel and Wolf Singer, "Selection of intrinsic horizontal connections in the visual cortex by correlated neuronal activity," *Science,* vol. 255, no. 5041, pp. 209-212, Jan. 1992.

[168] Brian S. Blais, N. Intrator, H. Shouval and Leon N. Cooper, "Receptive field formation in natural scene environments: Comparison of single-cell learning rule," *Neural Computation,* vol. 10, no. 7, pp. 1797-1813, Oct. 1998.

[169] Randall C. O'Reilly and Mark H. Johnson, "Object recognition and sensitive periods: A computational analysis of visual imprinting," *Neural Computation,* vol. 6, no. 3, pp. 357-389, May 1994.

[170] Donald O. Hebb, The Organization of Behavior: A Neuropsychological Theory, New York: Wiley & Sons, 1949.

[171] Simon Haykin, Neural Networks - A Comprehensive Foundation, 3 ed., Pearson Education, Jul. 1998.

[172] Pietro Mazzoni, Richard A. Andersen and Michael I. Jordan, "A more biologically plausible learning rule for neural networks," in *proc. of the National Academy of Sciences of the United States of America*, May 1991.

[173] Randall C. O’Reilly, "Biologically plausible error-driven learning using local activation differences:The generalized recirculation algorithm," *Neural Computation,* vol. 8, no. 5, pp. 895-938, 1996.

[174] Erzsebet Merenyi, "Chapter 3 - Hebbian Learning (ECE502 course)," Electrical and Computer Engineering, Rice University, USA, 2006. [Online].

[175] Erkki Oja, "A simplified neuron model as a principal component analyzer," *Mathematical Biology,* vol. 15, no. 3, pp. 267-273, 1982.

[176] Liu Chun-Lin, "A tutorial of the wavelet transform," DISP Lab, Graduate Institute of Communication Engineering, NTU, Taiwan, 23 Feb. 2010. [Online].

[177] Robin Sommer and Vern Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in *proc. of IEEE Symposium on Security and Privacy*, 2010.