A STUDY ON ANOMALY DETECTION USING MIXTURE MODELS

YOGESH PAWAR

A THESIS
IN
CONCORDIA INSTITUTE FOR INFORMATION SYSTEMS ENGINEERING

PRESENTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF APPLIED SCIENCE
(INFORMATION SYSTEMS SECURITY)
CONCORDIA UNIVERSITY
MONTRÉAL, QUÉBEC, CANADA

NOVEMBER 2020

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: **Yogesh Pawar**

Entitled: **A Study on Anomaly Detection Using Mixture Models**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science
(Information Systems Security)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining commitee:

Dr. Yong Zeng _____ Chair

Dr. Nizar Bouguila _____ Supervisor

Dr. Manar Amayri _____ Co-Supervisor

Dr. Farnoosh Naderkhani _____ CIISE Examiner

Dr. Lisa Kakinami _____ External Examiner

Approved _____

Dr. Mohammad Mannan, Graduate Program Director

2020 _____

Dr. Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

A Study on Anomaly Detection Using Mixture Models

Yogesh Pawar

With the increase in networks capacities and number of online users, threats of different cyber attacks on computer networks also increased significantly, causing the loss of a vast amount of money every year to various organizations. This requires the need to identify and group these threats according to different attack types. Many anomaly detection systems have been introduced over the years based on different machine learning algorithms. More precisely, unsupervised learning algorithms have proven to be very effective. In many research studies, to build an effective ADS system, finite mixture models have been widely accepted as an essential clustering method.

In this thesis, we deploy different non-Gaussian mixture models that have been proven to model well bounded and semi-bounded data. These models are based on the Dirichlet family of distributions. The deployed models are tested with Geometric Area Analysis Technique (GAA) and with an adversarial learning framework.

Moreover, we build an effective hybrid anomaly detection system with finite and infinite mixture models. In addition, we propose a feature selection approach based on the highest vote obtained. We evaluated the performance of mixture models with Geometric Area Analysis technique based on Trapezoidal Area Estimation (TAE) and the effect of adversarial learning on ADS performance via extensive experiments based on well-known data sets.

Acknowledgments

I would like to express my very profound gratitude to my supervisor *Prof. Nizar Bouguila*. Since the first day, I joined his team as a Master's student, throughout this time he provided his endless support and encouragement. He is incredibly kind and has given me guidance on several occasions. He motivated me to explore not just machine learning but also image processing and deep learning as a data science student. I had a fantastic experience that I will never forget. I will be forever grateful to him for all the experiences he made possible for me.

I express my profound gratitude to my co-supervisor *Dr. Manar Amayri* for her technical advices. I am deeply grateful for all the things I have learned from you. A great thank you from the heart to all my friends and fellow lab mates. Especially I would like to thank *Jaspreet*, *Kamal*, *Samr*, *Nuha*, and *Fahdah* for their support and encouragement during my studies.

Finally, I am deeply grateful to my family for their immense support and encouragement throughout my academic studies. My dear parents *Prakash Pawar* and *Lalita Pawar* and my siblings: *Priyanka* and *Shivam*, in particular, have shown me vast amounts of patience as I progressed through my studies. I could not achieve this without your support and I am truly thankful for always being there for me.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Anomaly Detection and Mixture Models

Intrusion Detection System (IDS) plays an essential role in ensuring the security of a network environment and achieving a solid line of defense against cyber intrusions. The primary purpose of IDS is to monitor host or network activities and detect possible threats by measuring their violations of confidentiality, integrity, and availability [2, 3]. In the machine learning context, this can be seen as a classification or novelty detection task that involves discovering enthralling and rare patterns in network data. IDS methodologies are grouped into three major categories: Misuse-based (MDS), Stateful Protocol Analysis (SPA), and Anomaly Detection Systems (ADS) [4]. The majority of ADS methodologies have been developed using approaches involving data mining and machine learning, artificial intelligence, knowledge-based, and statistical models. For instance, classification based approaches rely on building the knowledge base from the normal traffic activity profile and considering activities that deviate from the baseline profile as anomalous. Major classification-based ADS techniques used in the literature are developed using support vector machines (SVM) [5], Bayesian networks [6], neural networks [7], and rule-based [8] approaches. Anomaly Detection Systems have also been developed using statistical techniques where a threshold is defined to raise alarms for anomalous requests.

Different types of techniques have been developed based on statistical learning (e.g. mixture models) [9], and signal processing [10]. Unfortunately, many of them fail in terms of detection rate and time trade-off especially in the case of large-scale networks. Over the past few years, various statistical machine learning models have been proposed to mitigate

this limitation. The main goal of these models is to distinguish between normal and abnormal network events represented in terms of vectors of features. The task of distinguishing between an attack profile and a normal one is very challenging [11–13]. A recent study on ADS has shown that statistical-based mixture models are very effective to detect malicious network behaviors with significantly low FPR and high detection rate (DR). Finite mixture models for positive vectors, such as Dirichlet (Dir) [14], inverted Dirichlet (ID) [15], and generalized inverted Dirichlet (GID) [16] mixtures have proven to be more efficient than Gaussian mixture model in many real-world applications [17–19]. Indeed, both Dirichlet and inverted Dirichlet have shown to be more flexible for modeling multivariate data than the Gaussian, as both allow multiple symmetric and asymmetric modes [20].

With the spread of the Internet-of-Things (IoT) devices in different prime fields like smart cities, health care, smart home applications, industrial automation, supply chain, and military applications, a huge amount of data is transmitted everyday [21]. To ensure the security of user's privacy as well as Wireless Sensor Networks (WSNs), many supervisory and data acquisition systems often embed learning algorithms for anomaly detection. Rapid and accurate anomaly detection is one of the essential requirements to ensure the productive work of the network. To address different security concerns related to end-users privacy as well as data integrity, various techniques like fog computing, federated learning, and differential privacy are adopted in many real-world applications. Currently, due to the benefits of agility, flexibility, security, scalability, and cost-effectiveness, researchers and many industrial application service providers are moving towards cloud computing [22] [23]. Although, cloud computing provides many advantages, it has some limitations like privacy concerns, high latency, low mobility support, and geo-distribution [23].

Recently, some techniques such as fog computing, federated learning, and differential privacy have been proposed to overcome issues related to cloud computing. For example, fog computing, often called edge computing, provides applications, storage, and data to end-users with less geo-distribution limitations [24]. On the other hand, federated learning gives the ability to end-users to train different machine learning algorithms on multiple decentralized edge devices [25]. Moreover, privacy-related issues can be effectively handled by differential privacy techniques [26], where any sensitive information is kept secure on edge devices in such a way that it prevents anyone to be fully able to reverse-engineer the data to its original form, thus preserving the privacy of end-users as well as data integrity. Achieving edge computing capabilities from the above techniques give more advantages

over cloud computing. But, due to different common characteristics of IoT devices such as size, weak computation capability, and low memory capacity, edge computing techniques are often found to be vulnerable to various security threats. For example, in anomaly detection systems (ADS), a device that gets compromised by an adversary might affect the performance of the overall system by injecting malicious data during the training phase of the learning algorithm. An important problem that we tackle in this thesis is adversarial learning, wherein training, malicious data are injected as normal data in order to compromise ADS's performance.

1.2 Contributions

The goal of this thesis is to build and evaluate an anomaly detection system based on mixture models along with different learning approaches. We propose a feature selection approach based on the highest vote to achieve high accuracy and detection rate (DR). The contributions are listed as follows:

- ☞ **An Effective Hybrid Anomaly Detection System Based on Mixture Models.**

  We propose a feature selection approach based on the highest vote obtained by different techniques in the literature. We introduce a hybrid ADS system based on a combination of finite and infinite mixture models with variational learning for the created sub-datasets based on the attack types in the widely used UNSW-NB15 dataset. We evaluated the proposed framework using well-known datasets such as UNSW-NB15 and NSL-KDD and achieved high accuracy in detecting all attack types. This contribution has been accepted by the 7^{th} *International Symposium on Networks, Computers, and Communications.* [27]

- ☞ **Performance Evaluation of Geometric Area Analysis Technique for Anomaly Detection Using Trapezoidal Area Estimation.**

  We evaluate different mixture models within the anomaly detection framework proposed by [1]. We propose a feature selection approach based on the highest vote obtained by different techniques in the literature. We evaluated the GAA-TAE technique by deploying different mixture models for the widely used UNSW-NB15 and NSL-KDD data sets. This work has been accepted by the 7^{th} *International Symposium on Networks, Computers, and Communications.* [28]

☞ **Performance Evaluation of Adversarial Learning for Anomaly Detection using Mixture Models.**

We use different statistical mixture models based on variational learning to evaluate the performance of ADS considering the effect of adversarial data while estimating the model parameters in the training phase. A decision-making method proposed in [29] is used to create a baseline profile for normal and abnormal data. We evaluate different variational-based models performances with adversarial learning, wherein training, malicious data were injected as normal data. This research work has been submitted to the $22nd$ *IEEE International Conference on Industrial Technology*. [30]

1.3 Thesis Overview

The rest of this thesis is organized as follows

- ❏ Chapter 2 proposes an effective ADS system based on a hybrid of finite and infinite mixture models with maximum likelihood and variational Bayesian inference. By selecting an appropriate number of clusters using the variational Bayesian inference technique, as well as selecting the relevant features using the proposed voting approach, we achieved a significant improvement in the modeling accuracy.

- ❏ In chapter 3, we evaluate the Geometric Area Analysis technique based on Trapezoidal Area Estimation with different mixture models. We evaluated the proposed hybrid ADS through extensive experiments involving two well-known datasets, namely, NSL-KDD and UNSW-NB15.

- ❏ Chapter 4 describes the performance evaluation of different variational learning mixture models along with adversarial learning cases. We evaluated the ADS through extensive experiments involving two datasets: NSL-KDD and UNSW-NB15.

- ❏ In conclusion, we summarize our work and contributions with some remarks for future works.

Chapter 2

An Effective Hybrid Anomaly Detection System Based on Mixture Models

In this chapter, we present the mixture models that we have considered as well as the variational inference approach deployed for learning. Furthermore, we propose a new hybrid Anomaly Detection System (ADS) framework to detect both normal and abnormal patterns of behavior in high-dimensional network data.

2.1 Mixture Model for Multivariate Data

Machine learning algorithms are predominantly organized into two main families: supervised or unsupervised learning. In the case of supervised learning, data labels are provided to train a given model to tackle for instance classification and regression problems. However, in unsupervised learning, the model is learned without using class labels. Mixture models have been widely used as a formal approach to unsupervised learning. A mixture model is a probabilistic statistical model, that identifies a set of clusters within an overall population. Suppose that we have a set of N D-dimensional vectors $\mathcal{X} = \{\vec{X}_1, \ldots, \vec{X}_N\}$ where $\vec{X}_i = \{\vec{X}_{i1}, \ldots, \vec{X}_{iD}\}$ modeled by a finite mixture with M components, then

$$p(\vec{X}_i|\vec{\pi}, \Theta) = \sum_{j=1}^{M} \pi_j p(\vec{X}_i|\theta_j) \tag{1}$$

where $\Theta = (\theta_1, \ldots, \theta_M)$, θ_j is the parameter of the distribution representing component j, and $\vec{\pi} = (\pi_1, \ldots \pi_M)$ represents the mixing weights which are positive and sum to one.
Finite mixture models for positive vectors, such as Dirichlet (Dir) [14], inverted Dirichlet

(ID) [15], and generalized inverted Dirichlet (GID) [16] have proven to be more efficient than Gaussian mixture models in many real-world applications [12, 18, 19]. Indeed, both Dirichlet and inverted Dirichlet have shown to be more flexible for modeling multivariate data than the Gaussian, as both allow multiple symmetric and asymmetric modes [20]. Many other properties of inverted Dirichlet are described in [31]. However, inverted Dirichlet has a major limitation since it assumes that the features of a given data vector are positively correlated, which does not hold for many real-world applications.

In addition, other models such as Generalized Dirichlet (GD) [32], Generalized Inverted Dirichlet (GID) [33], Beta-Liouville (BL) [34], and inverted Beta-Liouville [20] mixture models have been recently developed to offer more general covariance structure and flexibility than Dirichlet- and inverted Dirichlet-based models. On the other hand, the grouping of network data into a finite number of clusters is not the best option in the case of ADS systems, where new malicious behaviors and new types of attacks may be seen. Thus, we considered also infinite mixture models, where the number of components M is supposed to be infinite and dynamically adjusted as data arrive. The detailed explanations and implementations for variational learning of infinite Dir, ID, and GID are given by [33, 35, 36], where stick-breaking representation [37] is adopted to construct the infinite model using a non-parametric Bayesian framework based on the Dirichlet process (DP) [38]. The considered distributions as well as a variational-based approach to learn their corresponding mixture models will be presented in the next section.

2.1.1 Model learning

Two approaches have been considered in the literature for training finite mixture models. First, the widely used technique is to estimate the related parameters using the maximum likelihood [39] approach through the expectation-maximization (EM) algorithm [40]. Details about the estimation of Dir, ID, and GID finite mixture models using this approach can be found in [14–16]. Another approach to estimate finite mixture model parameters is variational learning to handle implicitly uncertainty, which characterizes the anomaly detection problem. The detailed explanations and derivations for variational learning of finite Dir, ID, and GID are given in [33, 41, 42].

A finite Dirichlet mixture is obtained by considering that $p(\vec{X}_i|\theta_j)$ in Eq. 1 is a Dirichlet

distribution with its own positive parameters $\theta_j = \vec{\alpha}_j$:

$$p(\vec{X}_i|\vec{\alpha_j}) = Dir(\vec{X}_i|\vec{\alpha_j}) = \frac{\Gamma(\sum_{l=1}^{D}\alpha_{jl})}{\prod_{l=1}^{D}\Gamma(\alpha_{jl})}\prod_{l=1}^{D}X_{il}^{\alpha_{jl-1}}l \tag{2}$$

where $\vec{\alpha}_j = (\alpha_{j1}, \ldots, \alpha_{jD})$, D is the dimensionality of $\vec{X}$ and $\sum_{l=1}^{D} X_l = 1,\ 0 \leq X_l \leq 1$. In the case of an Inverted Dirichlet mixture, we have [43]

$$\begin{aligned} p(\vec{X}_i|\vec{\alpha_j}) = IDir(\vec{X}_i|\vec{\alpha_j}) = \frac{\Gamma(\sum_{l=1}^{D+1}\alpha_{jl})}{\prod_{l=1}^{D+1}\Gamma(\alpha_{jl})} \\ \prod_{l=1}^{D}X_l^{\alpha_{jl-1}}\ (1+\sum_{l=1}^{D}X_{il})^{-\sum_{l=1}^{D+1}\alpha_{jl}} \end{aligned} \tag{3}$$

where $0 \leq X_{il}$ and the parameters are now $\theta_j = \vec{\alpha}_j = (\alpha_{j1}, \ldots, \alpha_{jD})$.
In the case of Generalized Dirichlet mixture model, $p(\vec{X}_i|\theta_j)$ is the following distribution with parameter $\theta_j = (\vec{\alpha}_j, \vec{\beta}_j)$:

$$\begin{aligned} p(\vec{X}_i|\Theta_j) = GDir(\vec{X}_i|\vec{\alpha_j}, \vec{\beta_j}) = \prod_{l=1}^{D}\frac{\Gamma(\alpha_{jl}+\beta_{jl})}{\Gamma(\alpha_{jl})\Gamma(\beta_{jl})}X_{il}^{\alpha_{jl-1}} \\ (1-\sum_{K=1}^{l}X_{ik})^{\gamma_{jl}} \end{aligned} \tag{4}$$

where $\vec{\alpha}_j = (\alpha_{j1}, \ldots, \alpha_{jD})$, $\vec{\beta}_j = (\beta_{j1}, \ldots, \beta_{jD})$, $\sum_{l=1}^{D} X_{il} < 1$ and $0 < X_{il} < 1$ for $l = 1, ...D$, $\alpha_{jl} > 0,\ \beta_{jl} > 0$, $\gamma_{jl} = \beta_{jl} - \alpha_{jl+1} = \beta_{jl+1}$ for $l = 1, ..., D-1$, and $\gamma_{jD} = \beta_{jD-1}$.
The next deployed mixture is based on the generalized inverted Dirichlet with positive parameters $\theta_j = (\vec{\alpha}_j, \vec{\beta}_j)$, is defined as

$$\begin{aligned} p(\vec{X}_i|\Theta_j) = GID(\vec{X}_i|\vec{\alpha_j}, \vec{\beta_j}) = \prod_{l=1}^{D}\frac{\Gamma(\alpha_{jl}+\beta_{jl})}{\Gamma(\alpha_{jl})\Gamma(\beta_{jl})} \\ \frac{X_{il}^{\alpha_{jl-1}}}{(1+\sum_{K=1}^{l}X_{ik})^{\gamma_{jl}}} \end{aligned} \tag{5}$$

where $\vec{\alpha}_j = (\alpha_{j1}, \ldots, \alpha_{jD})$, $\vec{\beta}_j = (\beta_{j1}, \ldots, \beta_{jD})$, $0 < X_{il}$ for $l = 1, ...D$, $\alpha_{jl} > 0,\ \beta_{jl} > 0$, $\gamma_{jl} = \beta_{jl} + \alpha_{jl} - \beta_{jl+1}$, $l = 1, ..., D$ with β_{jD+1} for , and $\gamma_{jD} = \beta_{jD-1}$.
Finally, the Beta-Liouville and inverted Beta-Liouville mixtures are obtained using the following two distributions namely the Beta-Liouville distribution with set of parameters

$\theta_j = (\alpha_{j1}, ..., \alpha_{jD}, \alpha_j, \beta_j)$ and inverted Beta-Liouville distribution with set of parameters $\theta_j = (\alpha_{j1}, ..., \alpha_{jD}, \alpha_j, \beta_j, \lambda_j)$

$$\begin{aligned} BL(\vec{X}_i \mid \theta_j) = \frac{\Gamma(\sum_{l=1}^{D} \alpha_{jl})\Gamma(\alpha_j + \beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)} \prod_{l=1}^{D} \frac{X_{il}^{\alpha_{jl}-1}}{\Gamma(\alpha_{jl})} \\ \times (\sum_{l=1}^{D} X_{il})^{\alpha_j - \sum_{l=1}^{D} \alpha_{jl}} (1 - \sum_{l=1}^{D} X_{il})^{\beta_j - 1} \end{aligned} \tag{6}$$

where $\sum_{l=1}^{D} X_{il} < 1$ and $0 < X_{il} < 1$ for $l = 1, ...D$,

$$\begin{aligned} IBL(\vec{X}_i \mid \theta_j) = \frac{\Gamma(\sum_{l=1}^{D} \alpha_{jl})\Gamma(\alpha_j + \beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)} \prod_{l=1}^{D} \frac{X_{il}^{\alpha_{jl}-1}}{\Gamma(\alpha_{jl})} \\ \times \lambda^{\beta} (\sum_{l=1}^{D} X_{il})^{\alpha - \sum_{l=1}^{D} \alpha_{jl}} (\lambda + \sum_{l=1}^{D} X_{il})^{-(\alpha+\beta)} \end{aligned} \tag{7}$$

where $0 < X_{il}$ for $l = 1, ...D$.

2.1.1.1 Variational learning Approach

One of the most important tasks when deploying mixture models is parameters estimation. The approaches to estimate the parameters can be divided mainly into two types: deterministic and Bayesian. In the deterministic learning approach, maximum likelihood (ML) estimation using Expectation-Maximization (EM) [44] [41] algorithm is the most widely used. Deterministic approaches are very sensitive to initialization and may cause over-fitting problems. To overcome these limitations, Bayesian learning has been widely used in real-life applications especially via variatioanl inference. The primary goal of variational learning is to approximate the posterior distribution by minimizing the Kullback–Leibler (KL) divergence between the exact (or true) posterior and an approximating distribution [41] [45]. Using that inference procedure, along with estimating model parameters, the number of components can be automatically determined.

In our case, the goal is to estimate via variational inference the set of parameters Θ and mixing coefficients $\vec{\pi}$ for a mixture model by maximizing $p(\mathcal{X}|\vec{\pi})$ given by eq.8. In most of the cases, conjugate prior for a mixture model is intractable, mainly because of the difficulty to evaluate the normalization coefficient and thus can not be used directly in variational inference. Here, we also consider a binary random vector of M dimensions (latent variable) as $\vec{Z}_i = \{\vec{Z}_{i1}, ...\vec{Z}_{iM}\}$ where $\vec{Z}_{ij} = 1$ if $\vec{X}_i$ belongs to j^{th} component and

0, otherwise.

$$p(\mathcal{X}|\vec{\pi}) = \sum_{Z} \int p(\mathcal{X}, Z, \vec{\theta}|\vec{\pi}) d\vec{\theta} \tag{8}$$

At this point, the marginalization in above equation is intractable, thus the variational approach is used to find a tractable lower bound on $p(\mathcal{X}|\vec{\pi})$.

$$\ln p(\mathcal{X}|\vec{\pi}) = L(Q) - \int Q(\theta) \ \ln \frac{p(\theta|\mathcal{X}, \vec{\pi})}{Q(\theta)} d\theta \tag{9}$$

In the above equation, the lower bound $L(Q)$ is maximized when the KL divergence equals zero. The posterior distribution $Q(\theta)$ can be factorized into disjoint tractable distributions based on mean-field theory [46] [47].
To maximize $L(Q)$, the distribution of each factor is optimized with $L(Q)$ using the first-order and second-order Taylor approximations. The detailed variational inferences and complete learning algorithms for all the deployed mixture models can be found in [42] [33] [34] [20] [41].

2.2 The Proposed ADS Framework

This work is mainly motivated by two successful approaches recently proposed for intrusion detection. Moustafa et al. [48] proposed a scalable framework for building an effective ADS statistical decision engine based on the DMM for recognizing abnormal behaviors in network systems. Furthermore, the author in [49] proposed a hybrid IDS system based on different algorithms used together. The idea is to test different algorithms for each of the commonly known attack types on the widely used NSL-KDD dataset and the most suitable algorithm is determined according to the attack type, which has shown to be more efficient than using a single model. Thus, we propose a novel hybrid ADS based on the combination of finite and infinite flexible mixture models for non-Gaussian data.

The proposed framework consists of three modules of feature selection, training different models with each attack type and a statistical decision engine with a lower-upper interquartile range (IQR) [50]. More precisely, we first select the most relevant features in the dataset by combining the selected attributes using feature selection techniques widely used in the literature, as done in [49]. Next, we compute the density of non-Gaussian distributions for the normal profile as the training phase, then the parameters estimated using variational learning from the training phase are used in the testing phase. The algorithms

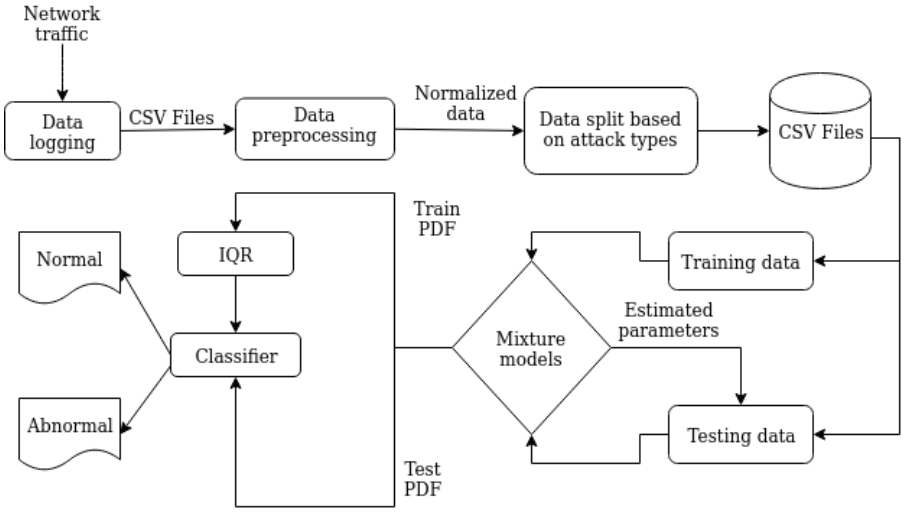


Figure 1: Proposed hybrid anomaly detection system.

with the highest accuracy, detection rate performance, and lowest error rate for each attack type are determined in the proposed system. Finally, the decision-making method for identifying anomalies is designed by specifying a threshold of the lower-upper IQR for the normal profile and considering any deviation from it as an attack as proposed in [48].The block diagram of the proposed system is shown in Fig.1.

2.2.1 Data pre-processing and Framework

The raw data obtained from the network can not be used directly in data analysis. This raw data is often incomplete, irreconcilable, and inadequate in certain behaviors or trends, and is foreseeable that it might contain many errors or null values. Before using the proposed framework, we performed data pre-processing on the considered widely-used datasets, namely NSL-KDD and UNSW-NB15, including feature transformation and normalization.

In feature transformation, categorical data such as $protocol\_type$,$service$, and $flag$ features of NSL-KDD dataset and for UNSW-NB15 dataset $proto$,$state$,and $service$ have been converted into numerical values. The data normalization is then performed in order to make data compatible with any other observation in the dataset. The data normalization can be performed using min-max scalar transformation in a range of 0 to 1. Normalization of data increases the performance of the system by reducing computational time.
The proposed framework involves three main phases, as follows:

1. **Feature selection** The high dimensionality of network data makes it even more challenging to achieve a good performance in identifying normal and abnormal behavior in network systems. The accuracy of an ADS system is given a high priority, but along with accuracy, it is also crucial to consider the time required to detect an attack. Having irrelevant features in the dataset may decrease the performance of the model in terms of both accuracy and time complexity, especially in unsupervised learning.

   In this study, we selected the most relevant features in each considered dataset based on the voting method. That is, we selected the high deterministic properties by combining attributes based on the highest vote of different feature selection techniques such as LightGBM, Random forest, RFE wrapper, data variance, feature correlation, and Chi-2. These techniques consider multiple aspects, including length, variance, dimensionality, correlation, and mean of the dataset. Each method gives a finite set of optimal features, and thus we combine the optimal features selected with the highest votes among the methods. These selected features are to be used in the training of the different mixture models. Using the proposed feature selection approach, the most relevant features obtained are presented in Table 1.

Table 1: Selected features from both datasets

Datasets	Selected features
NSL-KDD	dst_host_srv_count, dst_host_same_srv_rate, dst_host_count, same_srv_rate, protocol_type, logged_in, flag, dst_host_srv_diff_host_rate, dst_host_serror_rate, dst_host_same_src_port_rate, count
UNSW-NB15	synack, sttl, sinpkt, dttl, dload, ct_srv_dst, swin, smean, sload,sbytes, rate, dur, dmean, dbytes, ct_state_ttl, ct_srv_src, ct_dst_src_ltm, ackdat

2. **Training of models**

   After pre-processing and feature selection, we created sub-datasets based on the different attack types, In the NSL-KDD, we grouped different types of attacks into four major categories described in [49] as DoS, $U2R$, $R2L$, and $Probe$. For UNSW-NB15, we categorized the attacks into 8 different types including $Fuzzers$, $Analysis$, $Backdoors$, DoS, $Exploits$, $Generic$, $Reconnaissance$, and $Worms$. According to each attack type, the different considered mixture models are evaluated. In the

training stage, we initialize the model parameters using K-means and method of the moment, and these parameters are to be updated using the learning approaches to fit the datasets. After calculating the responsibility matrix for each data point in a dataset, IQR for a normal profile is calculated to be used in the testing phase.

3. **Tesing and Decision Engine**
   In the testing phase, parameters obtained from the training data set are used to calculate the probability of each data point that belongs to a particular cluster of the testing data set. The models with the highest accuracy, detection rate performance, and lowest error rate are determined according to attack types, and the selected algorithms are used in system design.

   For decision-making, the upper $(Q3)$, lower $(Q1)$, and intermediate (IQR) are calculated using training data of each normal profile to find the outliers/anomalies of any observed instance. Any new observation falling below $(PDF^{testing} < (lower - w * (IQR)))$ or above $(PDF^{testing} > (upper + w * (IQR)))$, is considered as an attack, and as normal otherwise [48]. The interval values [50] w has been chosen to be between 1.5 and 3 for finite mixture with maximum likelihood and for infinite variational mixture models. In addition, we chose interval values for finite variational mixture models between 1.5 and 5. For decision making, we compare $PDF^{testing}$ or responsibility matrix to IQR value obtained in the training phase of each model. The selected model, with the best performance, is then used to build hybrid ADS to increase the accuracy to identify normal and abnormal behaviors in NSL-KDD and NSW-NB15 datasets.

2.3 Experimental Results

In this section, we evaluated the effectiveness of the proposed hybrid ADS based on finite and infinite mixture models using two widely used datasets, namely, UNSW-NB15[1] and NSL-KDD[2]. We considered different models and learning approaches as follows:

- Finite mixture models with maximum likelihood learning approach: Dirichlet (Dir) , Inverted Dirichlet (ID), Generalized Inverted Dirichlet (GID).

[1] https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/

[2] https://www.unb.ca/cic/datasets/nsl.html

- Finite mixture models with variational learning approach: VDir, VIDir, VGID.
- Infinite mixture models with variational learning approach: InfVDir, InfVIDir, InfVGID.

In our experiments, after training the different models using the training data, we use the estimated model parameters to calculate the posterior probabilities of the testing set. The best model, according to each attack type, is selected and used in the model design. Finally, the statistical decision engine is used to detect normal and abnormal behaviors in the network system.

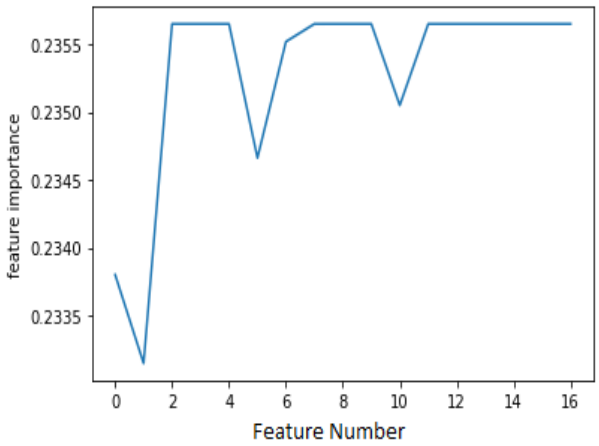


Figure 2: NSL-KDD: Feature importance computed in Infinite-VGID.

2.3.1 Feature selection approach evaluation

The first set of experiments evaluates the proposed feature selection approach using infinite VGID to compute the feature importance of our selected optimal features. In the NSL-KDD data set, we have a total of 41 features. Using the proposed feature selection technique, we were able to identify the most relevant 14 features to train our ADS and detect different attack types with high accuracy. We can show the effectiveness of our selected features for the NSL-KDD data set in Fig.2, where we have added one irrelevant feature at index 1. It is quite evident that the model accurately identified the irrelevant feature by computing low feature importance. Similarly, from a total of 47 features in the UNSW-NB15 data set, we have selected the 16 most relevant features where the adequacy of the selected features is shown in Fig.3 . For evaluation, we have added one irrelevant feature at index 5, and it

can be seen clearly in the figure that we have selected the optimal number of features by assigning low feature importance value stated as feature saliency in [33].

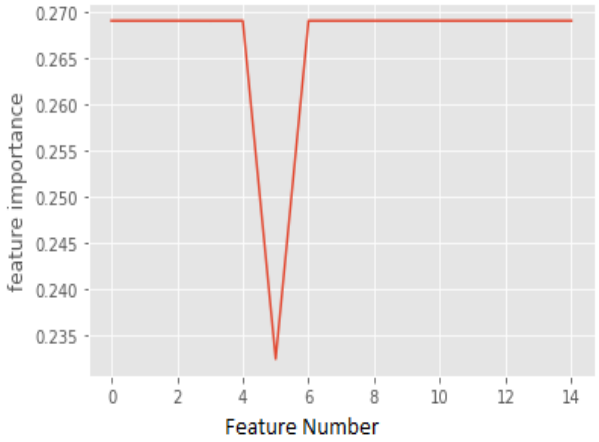


Figure 3: UNSW-NB15: Feature importance computed in Infinite-VGID.

2.3.2 ADS performance evaluation

The performance evaluation of mixture models based on the hybrid ADS framework was conducted on the selected features from the two datasets, measured by the overall accuracy, Detection Rate (DR) and False Positive Rate (FPR). Here, we compare the performance of both datasets for different models used to build hybrid ADS. From Table 3, it can be clearly inferred that the infinite VGID provides high accuracy and detection rate with low FPR as compared to other models for the NSL-KDD dataset, while GID, gives more accuracy but slightly high FPR compared to Infinite VGID. For the dataset UNSW-NB15, the infinite VDir gives high accuracy and with 0 FPR. Using GID, the overall detection rate and accuracy are higher than other models. For the NSL-KDD data set, after using the decision engine with different values of w ranging from 1.5 to 5.0, the accuracy of each model increases as the interval increases. From Table 3, we can conclude that all the models in finite mixtures give excellent results above 90.00%, but GID with variational learning provides the most accurate result with an accuracy around 94.05% with $w = 3.0$ compared to the other mixture models for NSL-KDD dataset.

Similarly, for the UNSW-NB15 data set, the accuracy of each model increases with the increase in the interval for different values of w from 1.5 to 5.0. By comparing the false positive rate from Table 9, we observe the same pattern of FPR that decreased abruptly

Table 2: Overall accuracy for dataset UNSW-NB15.

Model	ACC w/o DE (%)	ACC w DE (%)	DR (%)	FPR (%)}
Dir	91.66	91.66	96.49	9.2
ID	91.66	91.38	94.73	9.2
GID	94.63	92.75	92.36	7.2
VDir	85.44	87.92	96.42	18.5
VIDir	85.44	87.92	96.42	18.5
VGID	84.51	87.62	92.10	15.2
InfVDir	90.71	90.71	78.57	0.0
InfVIDir	85.44	85.44	96.42	22.9
InfVGID	85.45	90.40	85.79	3.5
InfVGID+FS	87.10	86.12	95.38	20.5

Table 3: Overall accuracy for dataset NSL-KDD.

Model	ACC w/o DE (%)	ACC w DE (%)	DR (%)	FPR (%)}
Dir	87.35	93.37	91.13	4.6
ID	87.49	93.39	93.01	6.1
GID	93.45	93.30	80.36	5.5
VDir	93.18	93.45	92.54	5.7
VIDir	88.06	91.41	94.66	11.4
VGID	83.29	94.05	89.10	5.4
InfVDir	79.17	89.49	83.03	4.8
InfVIDir	89.94	90.55	81.09	1.0
InfVGID	92.82	88.04	92.12	2.7
InfVGID+FS*	88.71	88.38	85.38	9.4

* Feature selection.

for the w values above 3.0 for finite variational; it's opposite to infinite, where we observe high FPR. So for this data set, we can consider the values of $(w >= 3.0)$ for our evaluated models. We can conclude that the infinite VDir provides excellent performance with an accuracy of 90.71% at $w = 3.0$, which is higher than the other infinite mixture models. The column values (A,B,C,D,E,F,G, and H) in Tables 4,5,6, and 7, denotes Worm, Reconnaissance, Fuzzers, Generic, Analysis, Exploits, Backdoor, and DoS attack types, respectively.

The next set of experiments evaluates the performance of the proposed model in detecting each type of attack in the UNSW-NB15 dataset. We compare the selected models based on DR, FPR, AND accuracy with and without using the decision engine. For the UNSW-NB15 dataset, we divided the dataset according to the type of attack and trained

Table 4: Accuracy with decision engine for all the attack types in UNSW-NB15 dataset.

Dataset	Attack types (%)							
	A	B	C	D	E	F	G	H
Dir	**97.64**	**97.46**	**94.88**	98.22	**89.33**	**89.33**	90.00	**96.66**
IDir	95.57	95.52	88.00	**99.11**	**88.55**	82.66	90.00	**91.55**
GID	95.30	86.92	**94.49**	**98.22**	86.00	88.88	**89.77**	90.00
VarDir	83.61	70.84	86.40	98.33	74.56	74.60	77.60	80.00
VarIDir	83.33	74.42	89.13	84.13	74.33	76.40	77.46	81.40
VarGID	83.87	82.22	81.11	83.55	78.66	79.11	77.77	80.88
InfVarDir	69.03	75.44	76.00	94.13	75.13	71.86	79.06	85.80
InfiVarIDir	**95.98**	79.47	78.73	**99.20**	74.62	74.81	86.76	76.81
InfiVarGID	78.47	**89.17**	**91.25**	97.90	**98.77**	**86.37**	**98.70**	**90.44**

Table 5: Accuracy without decision engine for all the attack types in UNSW-NB15 dataset

Dataset	Attack types (%)							
	A	B	C	D	E	F	G	H
Dir	94.98	97.01	94.44	93.55	81.00	88.00	78.64	89.55
IDir	94.98	93.59	84.88	93.55	85.22	74.66	78.66	89.77
GID	92.01	80.17	80.26	98.44	85.33	66.44	88.66	88.00
VarDir	76.29	74.55	64.26	98.60	72.73	76.06	77.61	79.20
VarIDir	76.29	74.55	69.40	91.46	71.46	75.26	76.73	79.33
VarGID	81.25	73.49	86.19	94.28	78.77	81.24	79.04	79.61
InfVarDir	68.49	77.21	73.94	93.67	74.06	84.40	76.73	79.60
InfiVarIDir	71.42	82.52	66.80	97.71	71.84	68.98	87.96	77.81
InfiVarGID	63.88	89.50	85.65	81.73	96.90	89.03	96.89	73.36

each sub-dataset with the different considered mixture models. Table 4 and 5, present the comparison of the performance accuracy with and without using the decision engine for each attack type. Table 6 and 7, show the comparison of the performance test results for DR and FPR for each attack category.

Table 6: False Positive Rate for all the attack types in UNSW-NB15 dataset

Dataset	Attack types (%)							
	A	B	C	D	E	F	G	H
Dir	0.6	2.9	4.3	0.0	3.6	14.9	13.2	0.3
IDir	3.7	2.6	17.0	0.0	5.7	20.0	13.5	9.6
GID	4.5	13.7	3.8	0.6	14.2	15.5	9.27	10.5
VarDir	16.2	34.5	16.0	0.0	26.0	27.1	24.8	24.9
VarIDir	16.4	19.8	12.8	23.0	26.2	35.4	24.9	22.8
VarGID	13.3	19.9	15.5	24.1	11.2	19.5	24.2	24.8
InfVarDir	34.0	25.5	22.5	8.4	24.2	20.0	24.8	21.1
InfiVarIDir	1.3	21.2	21.7	0.0	31.6	26.9	9.2	31.6
InfiVarGID	0.2	0.1	8.0	2.6	0.6	12.5	0.5	2.1

Table 7: Detection Rate for all the attack types in UNSW-NB15 dataset

Dataset	Attack types (%)							
	A	B	C	D	E	F	G	H
Dir	85.71	99.11	93.24	94.59	74.91	97.97	96.62	90.54
IDir	90.47	88.00	100.00	97.29	76.94	89.86	97.29	93.91
GID	92.68	89.77	86.98	94.00	86.48	97.97	87.83	91.21
VarDir	81.54	91.12	91.20	95.00	75.40	78.00	82.40	89.80
VarIDir	79.23	52.75	93.00	99.20	74.40	100.00	82.20	89.80
VarGID	69.56	97.05	74.32	99.32	58.10	76.35	81.75	92.56
InfVarDir	90.00	78.87	73.00	99.20	73.80	55.60	86.80	99.60
InfiVarIDir	79.54	81.31	79.85	97.54	87.73	83.79	78.52	94.47
InfiVarGID	86.36	99.70	87.70	99.08	89.95	81.11	86.27	72.29

The overall accuracy of the UNSW-NB15 dataset obtained without using the decision engine is 88.32%, and with decision engine accuracy increased to 90.64%. In our hybrid ADS, we achieved an accuracy of 96.58%, which is a significant increase over previous results obtained. The FPR of hybrid ADS decreased significantly from 0.12% to 0.03%. The best models to detect each type of attack according to accuracy, DR, and FPR are highlighted in Table 4.

Chapter 3

Performance Evaluation of Geometric Area Analysis Technique Using Trapezoidal Area Estimation

Previously, we have successfully applied the mixture models as well as the variational inference approach to estimate model parameters. The performance of models with variational learning approach was proven to be very effective compared to other described models. In this chapter, the GAA-TAE technique proposed in [1] is used and integrated into the ADS framework to detect abnormal data with high accuracy and low FPR.

3.1 Introduction

The main goal of a clustering algorithm is to find patterns and to group similar vectors in the same cluster. In the case of network data, it is very challenging to differentiate between normal and abnormal data vectors when both reflect the same pattern. To overcome this problem, different distance measures have been considered along with model-based clustering algorithms. The GAA technique, presented in [1] is one solution that we will follow and can be divided into following steps:

3.1.0.1 Normal profile creation

For D-dimensional positive vector $\vec{X}$ representing a network observation, the GAA technique is applied to calculate its area based on its TAE computed from the Beta mixture

model (BMM) parameters and distances of records. In this technique normal profile is constructed from legitimate network data. These legitimate data are first used to estimate BMM parameters and then used to calculate distances between the mean of normal records and each record. Finally, for each data vector TAE is calculated using the BMM Probability Density Function (PDF) and distance between records [51–53] for training and testing dataset.

3.1.0.2 TAE estimation

The final PDF of normal record and test record is used to estimate TAE from Eq. 10

$$area(V) = \frac{b-a}{D}[f(x_1) + 2\sum_{i=1}^{D-1} f(x_i) + f(x_D)] \tag{10}$$

GAA technique is mainly based on trapezoidal rule which is one of the numerical integration families called Newton-Cotes formulas [54]. When this rule is used for multivariate data it is called a composite trapezoidal rule. The PDF of each vector is considered as the area under the curve as shown in Fig. 4. The final normal areas are sorted and divided into

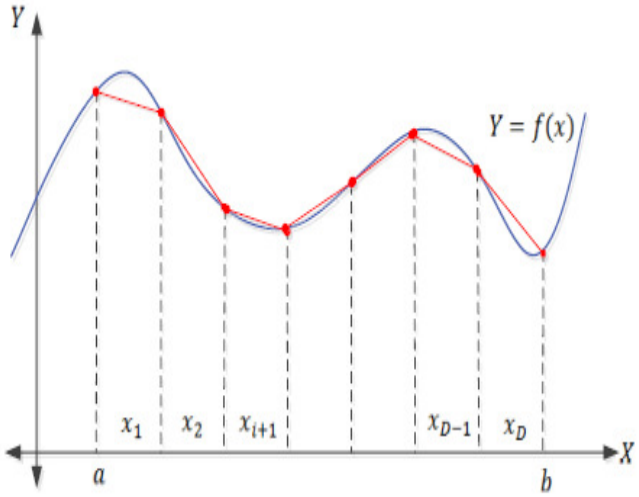


Figure 4: Composite trapezoidal rule. [1]

K_i intervals where each K_i represents a minimum and maximum value ($\min_{K_i}$ and $\max_{K_i}$, respectively) and used in decision-making step.

3.1.0.3 Testing and Decision-making

In this step, PDF and area of test data are calculated by using same estimated parameters from normal profile. Next, the test area is compared with normal area. If the area value for test data vector falls in between normal min and max range, it is considered a normal record, otherwise as an attack one.

3.1.1 Mixture Models

3.1.1.1 Beta Mixture Model

Mixture models have been widely used for data modeling. Although Gaussian mixture model, has proven to be efficient in several applications, it fails to fit the observations accurately especially when the data are clearly non-Gaussian due to its non-convex clustering properties [55]. BMM, on the other hand, has proven to be more efficient in handling many real-world applications involving one-dimensional data [56]. BMM can be more efficient on modeling the distribution of bounded data than the Gaussian mixture [57]. The PDF of a Beta distribution is given by:

$$Beta(x|v,w) = \frac{1}{beta(v,w)} x^{v-w}(1-x)^{w-1}, v, w > 0 \tag{11}$$

where $x \in [0,1]$ is the normalized feature, v and w indicate the shape parameters of the Beta distribution and $beta(v,w)$ is the Beta function given by:

$$\mathrm{beta}(v,w) = \frac{\Gamma(v)\,\Gamma(w)}{\Gamma(v+w)} \tag{12}$$

where $\Gamma()$ is the Gamma function.
Let $\vec{X} = (x_1, \ldots, x_D)$ be a D-dimensional vector of independent normalized features supposed to follow Beta distributions described in following Eq 13.

$$p(\vec{X}|\vec{v},\vec{w}) = \prod_{d=1}^{D} Beta(x_d; v_d, w_d) \tag{13}$$

where $\vec{v} = (v_1, \ldots, v_D)$ and $\vec{w} = (w_1, \ldots, w_D)$. In [1], the authors have considered a finite mixture model based on the distribution in Eq.13, by normalizing semi-bounded positive features, given by:

$$p(\vec{X}|\Theta) = \sum_{j=1}^{M} p(\vec{X}|\vec{v}_j, \vec{w}_j) p_j \tag{14}$$

where $\Theta = \{p_j, \vec{v}_j, \vec{w}_j\}$ refers to the entire set of parameters to be estimated, p_j are positive mixing proportions, with $\sum_{j=1}^{M} p_j = 1$, $p(\vec{X}|\vec{v}_j, \vec{w}_j)$ is the joint density function for a D-dimensional positive vector given by Eq.13. These parameters can be learned using the maximum likelihood approach proposed in [39] or the Bayesian one proposed in [57]. By investigating this mixture model, we can notice a main shortcoming which is related to supposing that the features are independent which may not be the case. The goal of this paper is to consider other mixture models to handle this shortcoming. In order to avoid supposing that the features are independent, we consider the generalized Dirichlet mixture [58]. The generalized Dirichlet distribution with parameters $\vec{\alpha} = (\alpha_1, \ldots, \alpha_D)$ and $\vec{\beta} = (\beta_1, \ldots, \beta_D)$ is explained in section 2.1.

As discussed extensively in a series of papers [58, 59] the consideration of the generalized Dirichlet allows the transformation of the data using a geometric transformation in such a way that the independence between the features becomes a fact and not an assumption. Each original vector $\vec{X}$ is geometrically transformed into a vector $\vec{Y} = (y_1, \ldots, y_D)$ as: $y_d = x_d$ if d = 1 and $y_d = \frac{x_d}{(1-\sum_{l=1}^{D-1} x_l)}$ for $d = 2, 3, \ldots, D$. Hence, each feature y_d has a Beta distribution and the resulting vector $\vec{Y}$ follows the distribution in Eq.13 .
In the following we consider two other techniques. The first one considers a mixture model based on Inverted Beta Distribution [15, 60] as a replacement to the Beta distribution considered in the original approach in [1]. The inverted Beta distribution is given by:

$$iBeta(x|\alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1+x)^{-(\alpha+\beta)} \tag{15}$$

where $x > 0$ and $\Gamma()$ is the Gamma function. The learning of the resulting mixture model could be based on the approaches proposed in [15, 60, 61].

However, it is clear that the previous technique supposes that the features are independent. A better alternative is the Generalized Inverted Dirichlet (GID) mixture as introduced [16]. The GID distribution is given by [16].

$$p(\vec{X}|\vec{\alpha}, \vec{\beta}) = \prod_{d=1}^{D} \frac{\Gamma(\alpha_d+\beta_d)}{\Gamma(\alpha_d)\Gamma(\beta_d)} \frac{x_d^{\alpha_{d-1}}}{(1+\sum_{l=1}^{D} x_l)^{\gamma_d}} \tag{16}$$

where $\gamma_d = \beta_d + \alpha_d - \beta_{d+1}$ for $d = 1, \ldots, D$ with $\beta_{D+1} = 0$.
As described in [16], we can factorize GID distribution as a product of inverted Beta distributions by using the following geometric transformation: $y_1 = x_1$ and $y_d = \frac{x_d}{(1+\sum_{d=1}^{D-1} x_d)}$ for $d = 2, 3, \ldots, D$. Thus, each feature y_d has an inverted beta distribution with parameters α_d and β_d as described in eq.15 . The learning of the parameters of a GID mixture model

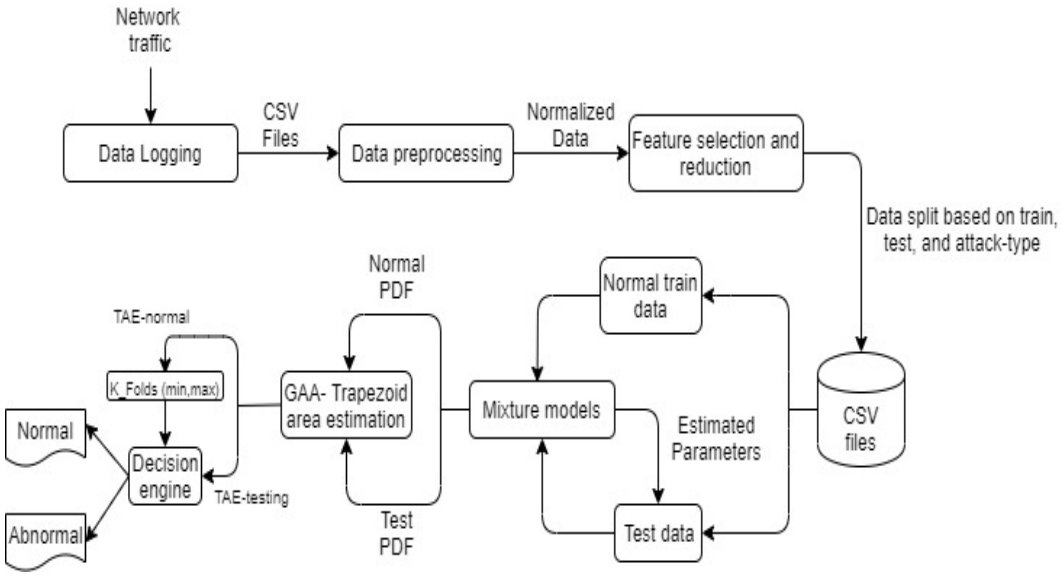


Figure 5: Framework for anomaly detection system.

could be based on the approaches proposed in [15, 33]. The PDFs estimated from above mixture models will then be used to calculate trapezoid area by applying the composite trapezoidal rule and uniform-grid property (features of equal length) as described in [62] and using Eq. 10.

3.2 ADS Framework

In this section, we describe the ADS framework to train the high dimensional vectors to create the normal profile using estimated parameters, distances between the means of the records, and normal area using the GAA-TAE method as described in [1]. This module can be divided into two sub-modules namely- area estimation, and a decision engine to distinguish between normal and abnormal data instances. The data preprocessing including dimensionality reduction techniques described in section 2.2.1. The block diagram of the system is shown in Fig.5.

3.2.1 Area estimation and standard profile creation

To build the ADS, we divided both datasets into training and testing sets. Only normal data vectors were selected to create a standard profile. Besides, we further divided the UNSW-NB15 dataset according to attack types, including Fuzzers, Analysis, Backdoors,

DoS, Exploits, Generic, Reconnaissance, and Worms to evaluate detection of each category using different models. In the training phase, for every given normal data vector we estimated the parameters of the tested mixture models using Maximum Likelihood (ML) and Expectation-Maximization (EM) algorithms. The normal profile includes the estimated parameters, PDFs for normal data vector (PDF^{normal}), absolute distance (calculated using mean of all the normal records (μ) and mean of each normal record (μ_n) using following equations 17 to 19, and mixing weights.

$$\mu = 1/N \sum_{i=1}^{N} v_i/(v_i + w_i) \tag{17}$$

$$\mu_n = 1/D \sum_{d=1}^{D} v_{nd}/(v_{nd} + w_{nd}) \tag{18}$$

$$abs_{distance} = |\mu - \mu_n| \tag{19}$$

In testing phase, we use same estimated parameters from normal profile to calculate PDFs for testing set $(PDF^{testing})$ and use mean of normal profile (μ_n) to calculate distance measure for testing records. After calculating (PDF^{normal}) and $(PDF^{testing})$, we use the PDFs to calculate TAE area using eq. 10 to get $(area^{normal})$ and $(area^{testing})$. Further, we divided $(area^{normal})$ into (K^{normal}) folds, where each fold contains minimum and maximum values of normal area for each normal vector. (K^{normal}) folds can be calculated using following equation [1]:

$$K_{folds} = [N/2], [(N-1)/2], [(N-2)/2], ..., [4/2] \tag{20}$$

Finally, we use this (K^{normal})) fold in the decision engine to classify normal and abnormal data instances proposed in [1]. Any observation falling in the range of $(area^{testing} >= min_{K_i})\ and\ (area^{testing} <= max_{K_i})$ is considered as normal otherwise as abnormal data vector.

3.3 Experimental Results

In this chapter, we evaluate the performance of the GAA technique using TAE. We deployed two widely used datasets, namely, UNSW-NB15 and NSL-KDD for the following

models: Beta Mixture Model (BMM), Inverted Beta Mixture Model (IBeta), Generalized Dirichlet Mixture Model (GDir), and Generalized Inverted Dirichlet (GID) mixture model. The parameters of these models have been estimated using the maximum likelihood approach within expectation-maximization framework as detailed in [39], [15], [58], and [16], respectively.

The first set of experiments evaluates the effectiveness of absolute distance with the TAE method to detect malicious attack vectors in a network with low FPR. For BMM, Fig. 6a and Fig. 6b represent the areas estimated for normal and abnormal data vectors without using any distance measure with minimum and maximum range for the normal profile from 0.2 and 0.89, respectively. TAE estimate for abnormal data instance is 0.87, which is under normal range and produces FPR.

Fig. 6c and Fig. 6d, represent IBeta-TAE for normal and abnormal data vectors. It can be observed that some abnormal data vectors fit the same as the normal vectors and thus generate high FPR. We can observe that by adding distance measure, in Fig. 6c and Fig. 6d, the model can distinguish between normal range (0.383 to 0.841) and abnormal data vectors. In this case, the overall IBeta-TAE value (0.836) is close to the normal range, but still, some abnormal data are considered as normal.

Fig. 6e and Fig. 6f show the effectiveness of the GID model using TAE technique and distance measure to detect the same normal and abnormal data vectors very effectively with a normal profile range from 0.27 to 0.82 and significant change in respective TAE values for abnormal data vector value as 0.895. Thus, the GID-TAE gives good detection capability with low FPR as we prove it also in the next set of experiments.

Fig. 6g and Fig. 6h, represent GDir-TAE for normal and abnormal data vectors. Although it can be observed that some abnormal data vectors look like normal ones and thus reduce the overall accuracy, the model can distinguish between normal range (0.12 to 0.77) and abnormal data vectors. In this case, the overall GDir-TAE value is (0.86) and generates less FPR.

In the second set of experiments, the performance evaluation of mixture models was conducted with selected features, and the principal components using the PCA technique for both datasets. The overall accuracy of the different models is measured by Detection Rate and False Positive Rate. Tables 8 and 9 summarize the obtained results for NSL-KDD and UNSW-NB15 data sets, respectively. The column values (A,B,C,D,E,F,G, and H) in Table 10, denotes Worm, Reconnaissance, Fuzzers, Generic, Analysis, Exploits, Backdoor,

Table 8: Overall accuracy for NSL-KDD.

Model	ACC (%)	DR (%)	FPR (%)
BMM	92.12	99.16	0.29
IBeta	90.50	97.00	0.29
GID	91.12	94.33	0.18
GDir	87.28	90.60	0.21

Table 9: Overall accuracy for UNSW-NB15.

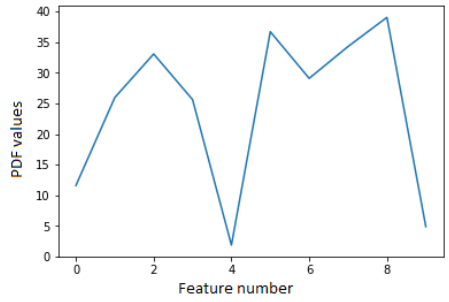
Model	ACC (%)	DR (%)	FPR (%)
BMM	95.86	93.50	0.023
IBeta	96.40	96.00	0.014
GID	96.44	97.50	0.010
GDir	98.85	96.00	0.010

and DoS attack types, respectively.

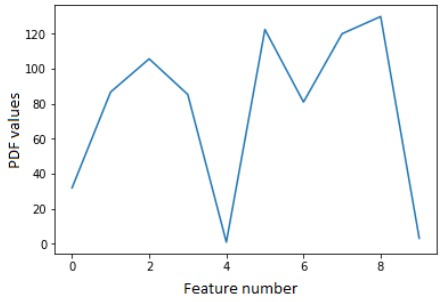
We evaluated the performance of the proposed model with the TAE technique in detecting each type of attack in the UNSWNB15 dataset in the final set of experiments. Table 10 shows the comparison of the performance test results for accuracy for each attack category in the UNSW-NB15 dataset. The performance of GID with TAE techniques gives us higher accuracy in each attack type as compared to other models except the shellcode attack type.

Table 10: Accuracy with TAE technique for all the attack types in UNSW-NB15 dataset

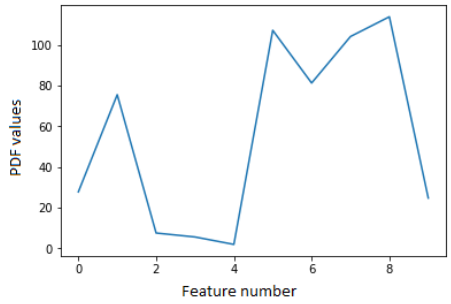
Dataset	Attack types (%)								
	A	B	C	D	E	F	G	H	I
BMM	92.93	95.06	94.53	90.80	90.53	91.33	93.46	87.06	89.83
IBMM	88.53	96.80	94.13	92.93	90.00	96.66	93.73	83.20	92.66
GID	91.28	99.57	96.71	100.00	90.85	99.85	97.14	85.71	94.28
GDir	97.57	93.57	96.71	83.85	90.28	99.85	90.10	88.42	96.82



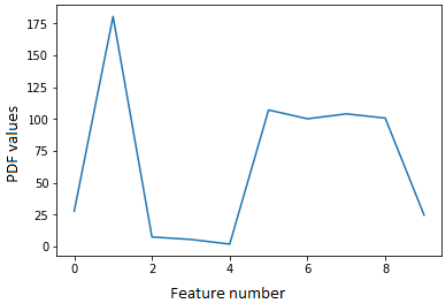
(a) Normal vector Beta PDF without Dist. TAE range (0.2-0.89)



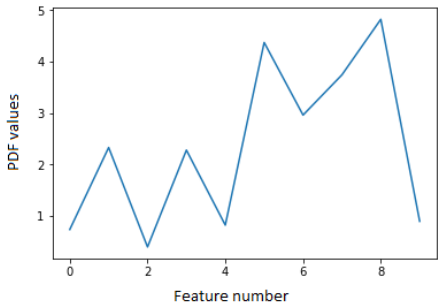
(b) Abnormal vector PDF (Beta with Dist. TAE-0.87)



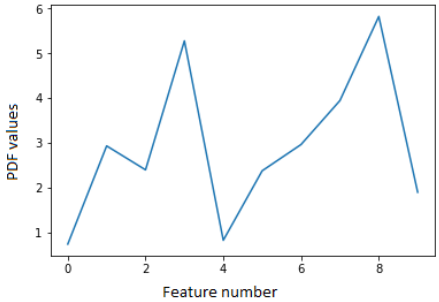
(c) Normal vector IBeta PDF without Dist. TAE (0.38-0.84)



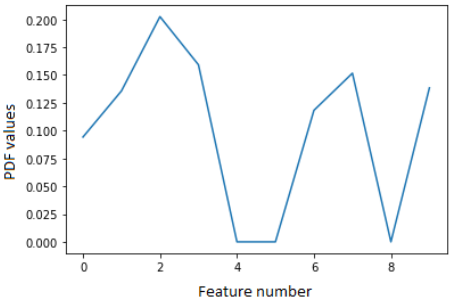
(d) Abnormal vector PDF (IBeta with Dist. TAE-0.83)



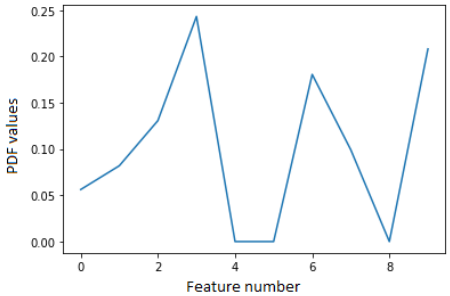
(e) Normal vector GID PDF without Dist. TAE (0.27-0.82)



(f) Abnormal vector PDF (GID with Dist. TAE-0.89)



(g) Normal vector GDir PDF without Dist. TAE (0.12-0.77)



(h) Abnormal vector PDF (GDir with Dist. TAE-0.86)

Figure 6: Normal profile range with TAE values Beta, IBeta, and GID model to detect abnormal data vector

Chapter 4

Performance Evaluation of Adversarial Learning using Mixture Models

In many cases, attackers mainly use two types of attacks, to compromise learning models, namely evasion and poisoning attacks [7] [8]. In evasion attacks, an adversary exploits specific vulnerabilities of the system to use it for future attacks wherein poisoning attacks, attacks aim to inject malicious data into the training phase of the algorithm in such a way that the baseline created by the learned model will generate a high false-positive rate and decrease the performance of the system to detect abnormality in the network. In this chapter, we describe the adversarial learning of ADS framework and the learning of the deployed models using the high dimensional training vectors to create normal profile. This module can be divided into two modules namely data pre-processing including dimensionality reduction, training and testing phases to calculate decision boundary. The block diagram of the ADS is shown in Fig. 7.

4.1 Mixture Models

Due to its simplicity, the Gaussian mixture model [63], where each $p(\vec{X}_i|\theta_j)$ is supposed to be a Gaussian distribution, has been widely used in many research works. However, it fails to discover the true underlying data structure in the case of non-Gaussian data. To handle non-Gaussian data, other mixture models that we shall investigate in this paper have been proposed. For instance, the Dirichlet mixture model (DMM) [64] was proposed to model proportional data while the Inverted Dirichlet mixture model (IDir) [15] was proven

to be more flexible to model positive vectors. In addition, other models such as Generalized Dirichlet (GD) [32], Generalized Inverted Dirichlet (GID) [16], Beta-Liouville (BL) [34], and inverted Beta-Liouville [20] mixture models have been recently developed to offer more general covariance structure and flexibility than Dirichlet- and inverted Dirichlet-based models. These distributions as well as a variational-based approach to learn their corresponding mixture models are presented in the section 2.1.

4.2 Adversarial Learning and ADS Framework

The primary goal of any machine learning algorithm is to identify the hidden pattern and structure of data using for instance different statistical methods. This process of learning can be manipulated by an adversary for different malicious reasons. This is often referred to as adversarial machine learning [65, 66]. For example, in ADS, an adversarial attack might inject malicious data into a machine learning model as it is normal legitimate training data, thus producing inaccurate training results to circumvent attacks in the future as normal. An adversary can inject and disturb the performance of a learning model at different stages of model learning like during data pre-processing, feature extraction, training, and testing. Mostly in poisoning attacks, data labels are manipulated during training, and in evasion attacks, it takes place after a model has already been trained by identifying boundaries that separate normal and abnormal data.

In our analysis, at the training phase, different finite mixture models, learned via variational inference, were deployed along with two different cases. In the first case, model's parameters are estimated with normal observations and in the second case, some malicious observations are considered as normal as an adversarial attack took place. Estimated parameters were then used to calculate probability density for each model in the testing phase and any variation from the estimated baseline is considered as an anomaly.

4.2.0.1 Data pre-processing

To improve model performance, the dataset needs to be pre-processed. At first, categorical features were converted into numerical values by assigning a unique number for each type to a feature. In the second step, we applied the same technique to identify the best features proposed in a previous study that we conducted [28] along with this we applied Principal Component Analysis (PCA) technique as done for instance in [1]. The data pre-processing

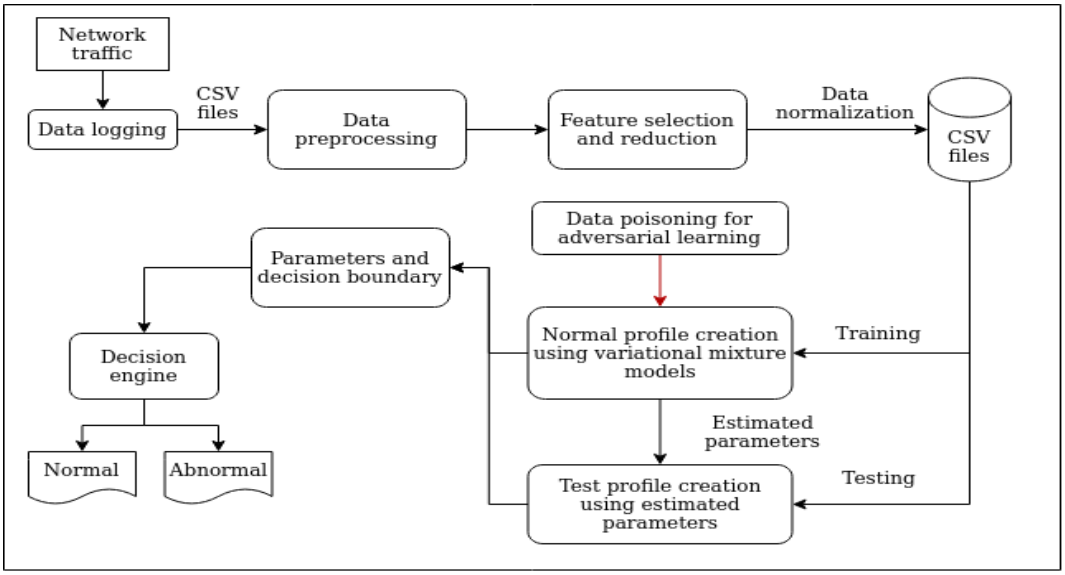


Figure 7: Framework for anomaly detection system.

and feature selection are explained in section 2.2.1.

4.2.0.2 Training and testing phase

After estimating the parameters of the different mixtures, we determine the normal profile consisting of upper and lower bound of training data from equations 21 and 22 developed in [29]:

$$lower^{normal} = \mu(pdf^{normal}) - (w * \sigma(pdf^{normal})) \tag{21}$$

$$lower^{normal} = \mu(pdf^{normal}) + (w * \sigma(pdf^{normal})) \tag{22}$$

This normal profile is then used in the classifier defined in [29] to distinguish normal and abnormal data as ($pdf^{testing} \geq lower^{normal} \,||\, pdf^{testing} \leq upper^{normal}$). Any data observation that deviates from the classifier range is considered as an attack, otherwise normal. We applied the above same procedure with an adversarial learning case where we inject some malicious data as normal data to determine the best variational mixture model to handle the adversarial attack.

4.3 Experimental Results

We have performed different experimental evaluation scenarios to assess the performance of the various variational mixture models with and without adversarial attacks. Each data instance in NSL-KDD consists of 41 features with one normal and four attacks labels. In the UNSW-NB15 dataset, each data instance consists of 47 features with one normal and nine different attacks categories labels. In our analysis, we have selected data of sample sizes between 120,000 and 160,000 for NSL-KDD and UNSW-NB15, respectively. As described in [29], for the first case we have selected legitimate data observation 60-75% of total data and for the second case, attack data samples are about 10-15% of the entire dataset. The ADS framework with different variation mixture models was evaluated using accuracy and false positive rate (FPR) metrics defined as:

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{23}$$

$$FPR = \frac{(FP)}{(FP + TN)} \tag{24}$$

where, the accuracy is the percentage of all normal and attack records correctly classified, TP represents true positive, TN represents true negative, FN represents false negative, and FP represents false positive. Based on our experiments, we initialize the number of mixture component to 10.

Table 11: FPR of the NSL-KDD dataset for both cases.

Model	Case-1 (%)	Case-2 (%)
Dir	0.050	0.060
IDir	0.059	0.060
GDir	0.090	0.091
GID	0.070	0.072
BL	0.036	0.039
IBL	0.035	0.038

Table 12: FPR of the UNSW-NB15 dataset for both cases.

Model	Case-1 (%)	Case-2 (%)
Dir	0.070	0.072
IDir	0.060	0.063
GDir	0.050	0.055
GID	0.040	0.047
BL	0.066	0.061
IBL	0.042	0.051

Table 13: Accuracy of the NSL-KDD dataset for both cases.

Model	Case-1 (%)	Case-2 (%)
Dir	92.98	92.14
IDir	92.91	92.15
GDir	89.99	89.71
GID	91.45	91.16
BL	94.55	94.37
IBL	94.67	94.59

Table 14: Accuracy of the UNSW-NB15 dataset for both cases.

Model	Case-1 (%)	Case-2 (%)
Dir	94.70	94.10
IDir	93.42	93.09
GDir	94.60	94.38
GID	94.75	94.52
BL	93.21	93.17
IBL	93.65	93.59

We evaluated the performance of the different variational mixture models while considering both cases (Case 1: without attacks, Case 2: with attacks). From Tables 13 and 11, it can be inferred that the accuracies of the Beta-Liouville and Inverted Beta-Liouville models were higher compared to other learning models. Furthermore, the FPRs of both BL and IBL were lower than the other models in the NSL-KDD dataset. For the UNSW-NB15 dataset, the accuracy of variational GID was highest with lowest FPR as shown in Tables 14 and 12.

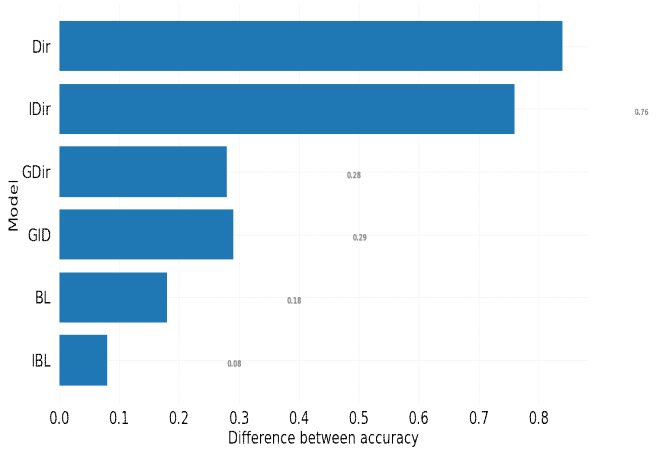


Figure 8: Accuracy difference between two cases for NSL-KDD dataset.

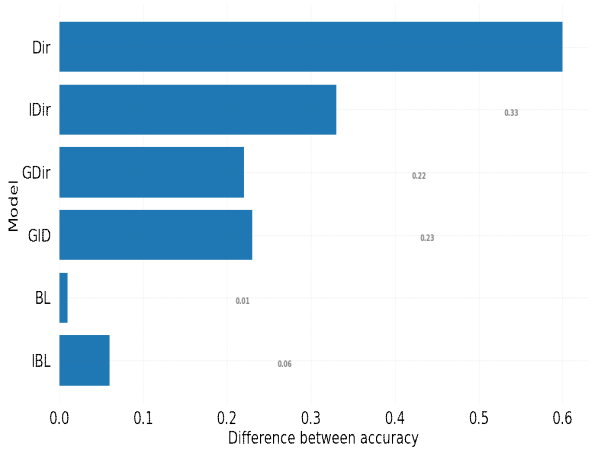


Figure 9: Accuracy difference between two cases for UNSW-NB15 dataset.

The absolute differences between accuracies for both cases for the NSL-KDD dataset

is displayed in Fig. 8. We can notice that IBL, and BL have the lowest difference. Thus, both models are more robust to handle adversarial attacks than others. Similarly, for the UNSW-NB15 dataset, according to the absolute differences in Fig. 9, both IBL and BL have the lowest differences of accuracies for both cases.

Chapter 5

Conclusion

Computer security concerns have been greatly exacerbated due to the evolution of computer networks in everyday life, especially internet security. Thus, the detection of malicious network behaviors has become the highest priority today.

In chapter 2, we have proposed a hybrid ADS and we have shown that it provides promising results by accurately detecting abnormal network behavior. Moreover, results indicate that the FPR in the proposed ADS is much less as compared to using a single mixture model. Thus, it can be considered as a powerful tool for analyzing non-Gaussian data. By selecting an appropriate number of clusters using the variational Bayesian inference technique, as well as selecting the optimal features using the proposed voting approach, we achieved a significant improvement in the modeling accuracy.

Then, in chapter 3, we implemented the Geometric Area Analysis technique based on Trapezoidal Area Estimation with different mixture models. We evaluated the proposed hybrid ADS through extensive experiments involving two datasets NSL-KDD and UNSW-NB15. We have shown that the GID and GDir using the TAE technique provide promising results by accurately detecting abnormal network behavior. Moreover, results indicate that the FPR in the GID is much less as compared to other mixture models. Thus, it can be considered to build ADS for a high-speed network to detect malicious activity. By selecting an appropriate number of folds and selecting the optimal number of features using the voting approach with the PCA technique for dimensionality reduction, we achieved a significant improvement in the modeling accuracy.

Finally, we evaluated the performance of different variational learning mixture models along with adversarial learning cases. We evaluated ADS through two datasets, namely,

NSL-KDD and UNSW-NB15. We have shown that the Beta-Liouville and the Inverted Beta-Liouville with adversarial learning provide promising results by accurately detecting abnormal network behavior. Moreover, results indicate that the FPR in the BL and IBL is much less as compared to other mixture models.

Future works could be devoted to extending the adversarial learning technique using infinite mixture models for a high-speed network on edge computing architecture to detect malicious activity.

Bibliography

[1] Nour Moustafa, Jill Slay, and Gideon Creech. Novel geometric area analysis technique for anomaly detection using trapezoidal area estimation on large-scale networks. *IEEE Transactions on Big Data*, PP, 07 2017.

[2] Robin Berthier, William H Sanders, and Himanshu Khurana. Intrusion detection for advanced metering infrastructures: Requirements and architectural directions. In *2010 First IEEE International Conference on Smart Grid Communications*, pages 350–355. IEEE, 2010.

[3] Vidar Evenrud Seeberg and Slobodan Petrovic. A new classification scheme for anonymization of real data used in ids benchmarking. In *The Second International Conference on Availability, Reliability and Security (ARES'07)*, pages 385–390. IEEE, 2007.

[4] Salvatore Pontarelli, Giuseppe Bianchi, and Simone Teofili. Traffic-aware design of a high-speed fpga network intrusion detection system. *IEEE Transactions on Computers*, 62(11):2322–2334, 2012.

[5] Eleazar Eskin, Andrew Arnold, Michael Prerau, Leonid Portnoy, and Sal Stolfo. A geometric framework for unsupervised anomaly detection. In *Applications of data mining in computer security*, pages 77–101. Springer, 2002.

[6] Christopher Kruegel, Darren Mutz, William Robertson, and Fredrik Valeur. Bayesian event classification for intrusion detection. In *19th Annual Computer Security Applications Conference, 2003. Proceedings.*, pages 14–23. IEEE, 2003.

[7] Simon Hawkins, Hongxing He, Graham Williams, and Rohan Baxter. Outlier detection using replicator neural networks. In *International Conference on Data Warehousing and Knowledge Discovery*, pages 170–180. Springer, 2002.

[8] Wenke Lee, Salvatore J Stolfo, and Kui W Mok. A data mining framework for building intrusion detection models. In *Proceedings of the 1999 IEEE Symposium on Security and Privacy (Cat. No. 99CB36344)*, pages 120–132. IEEE, 1999.

[9] Eleazar Eskin. Anomaly detection over noisy data using learned probability distributions. In *Proceedings of the seventeenth international conference on machine learning, ICML'00*, pages 255–262.

[10] Marina Thottan and Chuanyi Ji. Anomaly detection in ip networks. *IEEE Transactions on signal processing*, 51(8):2191–2204, 2003.

[11] Shuai Fu and Nizar Bouguila. A bayesian intrusion detection framework. In *2018 International Conference on Cyber Security and Protection of Digital Services, Cyber Security*, pages 1–8, 2018.

[12] Wentao Fan, Nizar Bouguila, and Djemel Ziou. Unsupervised anomaly intrusion detection via localized bayesian feature selection. In *11th IEEE International Conference on Data Mining, ICDM 2011, Vancouver, BC, Canada, December 11-14, 2011*, pages 1032–1037, 2011.

[13] Nizar Bouguila and Tarek Elguebaly. A fully bayesian model based on reversible jump MCMC and finite beta mixtures for clustering. *Expert Syst. Appl.*, 39(5):5946–5959, 2012.

[14] N. Bouguila and D. Ziou. Unsupervised selection of a finite dirichlet mixture model: an mml-based approach. *IEEE Transactions on Knowledge and Data Engineering*, 18(8):993–1009, Aug 2006.

[15] Taoufik Bdiri and Nizar Bouguila. Positive vectors clustering using inverted dirichlet finite mixture models. *Expert Syst. Appl.*, 39:1869–1882, 02 2012.

[16] Sami Bourouis, Mohamed Al Mashrgy, and Nizar Bouguila. Bayesian learning of finite generalized inverted dirichlet mixtures: Application to object classification and forgery detection. *Expert Systems with Applications*, 41(5):2329 – 2336, 2014.

[17] Sangwoo Park, Erchin Serpedin, and Khalid Qaraqe. Gaussian assumption: The least favorable but the most useful [lecture notes]. *IEEE Signal Processing Magazine*, 30, 11 2012.

[18] Mohiuddin Ahmed, Abdun Mahmood, and Jiankun Hu. A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, 60:19–31, 11 2015.

[19] Stephen Ranshous, Shitian Shen, Danai Koutra, Steve Harenberg, Christos Faloutsos, and Nagiza F. Samatova. Anomaly detection in dynamic networks: a survey. *Wiley Interdisciplinary Reviews: Computational Statistics*, 7(3):223–247, 2015.

[20] Can Hu, Wentao Fan, Ji-Xiang Du, and Nizar Bouguila. A novel statistical approach for clustering positive data based on finite inverted beta-liouville mixture models. *Neurocomputing*, 333:110–123, 2019.

[21] Mohit Taneja and Alan Davy. Resource aware placement of iot application modules in fog-cloud computing paradigm. pages 1222–1228, 05 2017.

[22] Ms Parag K Shelke, Ms Sneha Sontakke, and AD Gawande. Intrusion detection system for cloud computing. *International Journal of Scientific & Technology Research*, 1(4):67–71, 2012.

[23] Mark D Ryan. Cloud computing security: The scientific challenge, and a survey of solutions. *Journal of Systems and Software*, 86(9):2263–2268, 2013.

[24] Amandeep Singh Sohal, Rajinder Sandhu, Sandeep K Sood, and Victor Chang. A cybersecurity framework to identify malicious edge device in fog computing and cloud-of-things environments. *Computers & Security*, 74:340–354, 2018.

[25] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konečnỳ, Stefano Mazzocchi, H Brendan McMahan, et al. Towards federated learning at scale: System design. *arXiv preprint arXiv:1902.01046*, 2019.

[26] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

[27] Yogesh Pawar, Nuha Zamzami, and Nizar Bouguila. An effective hybrid anomaly detection system based on mixture models. In *IEEE International Symposium on Networks, Computers and Communications*, 2020.

[28] Yogesh Pawar, Manar Amayri, and Nizar Bouguila. Performance evaluation and analysis of geometric area analysis technique for anomaly detection using trapezoidal area estimation based on mixture models. In *IEEE International Symposium on Networks, Computers and Communications*, 2020.

[29] Nour Moustafa, Kim-Kwang Raymond Choo, Ibrahim Radwan, and Seyit Camtepe. Outlier dirichlet mixture mechanism: Adversarial statistical learning for anomaly detection in the fog. *IEEE Transactions on Information Forensics and Security*, 14(8):1975–1987, 2019.

[30] Yogesh Pawar, Manar Amayri, and Nizar Bouguila. Performance evaluation of adversarial learning for anomaly detection using mixture models. In *IEEE International Conference on Industrial Technology*, 2021.

[31] M. Ghorbel. On the inverted dirichlet distribution. *Communications in Statistics - Theory and Methods*, 39(1):21–37, 2009.

[32] Nizar Bouguila and Djemel Ziou. A powreful finite mixture model based on the generalized dirichlet distribution: Unsupervised learning and applications. volume 1, pages 280 – 283 Vol.1, 09 2004.

[33] Taoufik Bdiri, Nizar Bouguila, and Djemel Ziou. Variational bayesian inference for infinite generalized inverted dirichlet mixtures with feature selection and its application to clustering. *Applied Intelligence*, 44(3):507–525, Apr 2016.

[34] Wentao Fan and Nizar Bouguila. Learning finite beta-liouville mixture models via variational bayes for proportional data clustering. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

[35] Wentao Fan and Nizar Bouguila. Infinite dirichlet mixture model and its application via variational bayes. In Xue-wen Chen, Tharam S. Dillon, Hisao Ishibuchi, Jian Pei, Haixun Wang, and M. Arif Wani, editors, *10th International Conference on Machine Learning and Applications and Workshops, ICMLA 2011, Honolulu, Hawaii, USA, December 18-21, 2011. Volume 1: Main Conference*, pages 129–132. IEEE Computer Society, 2011.

[36] Wentao Fan and Nizar Bouguila. Topic novelty detection using infinite variational inverted dirichlet mixture models. pages 70–75, 12 2015.

[37] David Blei and Michael Jordan. Variational inference for dirichlet process mixtures. *Bayesian Analysis*, 1, 03 2006.

[38] Wentao Fan and Nizar Bouguila. Variational learning of dirichlet process mixtures of generalized dirichlet distributions and its applications. In Shuigeng Zhou, Songmao Zhang, and George Karypis, editors, *Advanced Data Mining and Applications*, pages 199–213. Springer Berlin Heidelberg, 2012.

[39] Nizar Bouguila and Djemel Ziou. Using unsupervised learning of a finite dirichlet mixture model to improve pattern recognition applications. *Pattern Recognit. Lett.*, 26(12):1916–1925, 2005.

[40] S. Ganesaligman. Classification and mixture approaches to clustering via maximum likelihood. *Bayesian Anal.*, page 38(3).

[41] Wentao Fan, Nizar Bouguila, and Djemel Ziou. Variational learning for finite dirichlet mixture models and applications. *IEEE Transactions on Neural Networks*, 23:762–774, 05 2012.

[42] Parisa Tirdad, Nizar Bouguila, and Djemel Ziou. *Variational Learning of Finite Inverted Dirichlet Mixture Models and Applications*, pages 119–145. Springer International Publishing, Cham, 2015.

[43] George G. Tiao and Irwin Cuttman. The inverted dirichlet distribution with applications. *Journal of the American Statistical Association*, 60(311):793–805, 1965.

[44] Geoffrey J McLachlan and Thriyambakam Krishnan. *The EM algorithm and extensions*, volume 382. John Wiley & Sons, 2007.

[45] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[46] Christopher Bishop, Neil Lawrence, Tommi Jaakkola, and Michael Jordan. Approximating posterior distributions in belief networks using mixtures. *Advances in neural information processing systems*, 10:416–422, 1997.

[47] Edouard Brézin. *Introduction to statistical field theory*. Cambridge University Press, 2010.

[48] Nour Moustafa, Gideon Creech, and Jill Slay. *Big Data Analytics for Intrusion Detection System: Statistical Decision-Making Using Finite Dirichlet Mixture Models*. 05 2017.

[49] Ünal Çavuşoğlu. A new hybrid approach for intrusion detection using machine learning methods. *Applied Intelligence*, 49(7):2735–2761, 2019.

[50] Peter J. Rousseeuw and Mia Hubert. Robust statistics for outlier detection. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1):73–79, 2011.

[51] Charu C Aggarwal. Outlier analysis. In *Data mining*, pages 237–263. Springer, 2015.

[52] Oded Maimon and Lior Rokach. *Data mining and knowledge discovery handbook*. Springer, 2005.

[53] Sung-Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions. *Int. J. Math. Model. Meth. Appl. Sci.*, 1:300–307, 2007.

[54] Josef Stoer and Roland Bulirsch. *Introduction to numerical analysis*, volume 12. Springer Science & Business Media, 2013.

[55] Wei Pan, Xiaotong Shen, and Binghui Liu. Cluster analysis: Unsupervised learning via supervised learning with a non-convex penalty. *Journal of machine learning research : JMLR*, 14:1865, 07 2013.

[56] Yuan Ji, Chunlei Wu, Ping Liu, Jing Wang, and Kevin R. Coombes. Applications of beta-mixture models in bioinformatics. *Bioinformatics*, 21(9):2118–2122, 02 2005.

[57] Nizar Bouguila, Djemel Ziou, and Ernest Monga. Practical bayesian estimation of a finite beta mixture through gibbs sampling and its applications. *Statistics and Computing*, 16:215–225, 06 2006.

[58] Nizar Bouguila and Djemel Ziou. A powreful finite mixture model based on the generalized dirichlet distribution: Unsupervised learning and applications. In *17th ICPR 2004, Cambridge, UK, August 23-26, 2004*, pages 280–283, 2004.

[59] Nizar Bouguila and Djemel Ziou. A countably infinite mixture model for clustering and feature selection. *Knowl. Inf. Syst.*, 33(2), 2012.

[60] Taoufik Bdiri and Nizar Bouguila. Learning inverted dirichlet mixtures for positive data clustering. In *Proc. of the 13th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, RSFDGrC'11, pages 265–272. Springer-Verlag, 2011.

[61] Taoufik Bdiri and Nizar Bouguila. Bayesian learning of inverted dirichlet mixtures for SVM kernels generation. *Neural Computing and Applications*, 23(5):1443–1458, 2013.

[62] Takao Asano, Tetsuo Asano, and Hiroshi Imai. Partitioning a polygonal region into trapezoids. *Journal of the ACM*, 33:290–312, 1986.

[63] Douglas A Reynolds. Gaussian mixture models. *Encyclopedia of biometrics*, 741, 2009.

[64] Nizar Bouguila, Djemel Ziou, and Jean Vaillancourt. Unsupervised learning of a finite mixture model based on the dirichlet distribution and its application. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 13:1533–43, 11 2004.

[65] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016.

[66] Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84:317 – 331, 2018.