International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Fuzzy Integrated Latent Dirichlet Allocation Algorithm for Intrusion Detection in Cloud Environments

*[1]Mr. Vinayak Kishan Nirmale, [2]Dr. C. Madhusudhana Rao, [3]Mr. Mylapalli Ramesh, [4]Dr. M. Nirmala, [5]Mr. S. Girinath, [6]Dr. Nellore Manoj Kumar

Submitted: 23/12/2023 **Revised**: 29/01/2024 **Accepted**: 07/02/2024

Abstract: This research presents an in-depth exploration of the FI-LDA model, showcasing its efficacy in anticipating and preventing intrusions, thereby bolstering security measures within cloud environments. The study introduces a novel approach to intrusion prevention, fostering a robust predictive model that significantly enhances the system's capability to discern evolving attack patterns. Leveraging fuzzy modeling, the research demonstrates the utilization of vast amounts of unlabeled data, resulting in heightened accuracy and reliability of the system. The evaluation of diverse elements crucial for cybersecurity underscores the comprehensive approach adopted to achieve the research objectives. While the FI-LDA model exhibited a favorable trade-off, addressing a pervasive flaw, there remains a call for further refinement to detect assault patterns more effectively. The research concludes by highlighting the commendable effectiveness of the FI-LDA model in identifying and detecting malicious activities within the cloud environment, affirming its strong overall performance and contribution to advancing intrusion detection systems.

Keywords: *Intrusion, Cloud, Fuzzy, Latent Dirichlet, NIDS, DoS, R2L, FI-LDA*

1. Introduction

In recent years, the widespread adoption of cloud computing has ushered in a new era of technological advancement, redefining the way data is stored, processed, and accessed. Cloud services offer unparalleled convenience, scalability, and cost-effectiveness, making them integral to the operations of organizations across various industries. However, the increasing reliance on cloud platforms has given rise to a parallel surge in cyber threats, targeting the vulnerabilities inherent in this distributed and interconnected ecosystem.

The inherent nature of cloud computing, characterized by shared resources, virtualization, and on-demand service provisioning, introduces unique challenges for traditional security measures. Conventional intrusion detection systems, designed for traditional network architectures, struggle to adapt to the dynamic and elastic nature of cloud environments. As a result, there is a pressing need to develop advanced intrusion detection systems that are tailored specifically for the intricacies of cloud computing.

This research aims to explore the evolving landscape of intrusion detection systems within the context of cloud computing. The study focuses on the distinctive challenges posed by the cloud environment and proposes innovative approaches to enhance the detection and mitigation of cyber threats. By delving into the complexities of cloud infrastructure and the characteristics of modern cyber threats, this research seeks to contribute valuable insights into the design, implementation, and optimization of intrusion detection systems for cloud computing.

The inherent advantages of cloud technology, such as on-demand provisioning, resource sharing, and remote accessibility, have positioned it as a linchpin in modern information technology infrastructures. However, this paradigm shift has brought forth a concomitant surge in cybersecurity threats, with attackers leveraging the unique features of cloud environments to exploit vulnerabilities and perpetrate sophisticated attacks.

The shared nature of resources, virtualization, and the decentralization of data in cloud architectures challenge the traditional security paradigms built for on-premises networks. Intrusion detection, a crucial component of any cybersecurity strategy, must evolve to address the intricacies of the cloud environment. Unlike traditional

[1]*Lecturer in Mathematics, Department of Polytechnic, MIT World Peace University, Paud Road, Kothrud, Pune, Maharashtra, India, Pincode: 411038,*
Email id: vinayak.nirmale@mitwpu.edu.in
[2] *Professor and Head, Department of CSE, Institute of Aeronautical Engineering, Hyderabad. Telangana, India, Pincode: 500043, Email: npr4567@gmail.com*
[3]*Assistant Professor, Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur Dist., Andhra Pradesh, India, Pincode: 522302. Email id: mylapalli2@gmail.com*
[4]*Professor, Department of Mathematics, Sathyabama Institute of Science and Technology, Chennai, Tamilnadu, India, Pincode: 600119. Email: nirmala.maths@sathyabama.ac.in*
[5]*Assistant Professor, Department of Computer Applications, Mohan Babu University, A. Rangampeta, Tirupati, Andhra Pradesh, India, Pincode: 517 102. Email: girisuddala@gmail.com*
[6]*Department of Mathematics, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences (SIMATS), Thandalam, Chennai, Tamilnadu, India, Pincode: 602 105. Email: nelloremk@gmail.com*

networks with well-defined perimeters, cloud ecosystems are characterized by fluidity, elasticity, and a distributed infrastructure, necessitating an adaptive and responsive approach to threat detection.

The migration of critical services and sensitive data to cloud environments introduces a multitude of security concerns, including unauthorized access, data breaches, and service disruptions. Traditional intrusion detection systems, often signature-based and reliant on predefined rules, struggle to adapt to the dynamic and scalable nature of cloud architectures. The conventional approach of perimeter defense becomes less effective as the boundaries of the traditional network dissolve in the cloud.

Traditional intrusion detection systems, designed for static and monolithic networks, often struggle to keep pace with the dynamic nature of cloud environments. Signature-based detection mechanisms, reliant on known patterns of attacks, may fall short in identifying novel threats that exploit the unique vulnerabilities inherent in virtualized and shared resources. The shift to microservices, containers, and serverless computing further complicates the security landscape, requiring a reevaluation of intrusion detection strategies to ensure comprehensive coverage.

The stakes are high in cloud security, as a breach can led to not only data compromise but also service disruptions and financial losses. The need for a robust and adaptive Intrusion Detection System tailored for the cloud environment has never been more critical. This research seeks to address this gap by examining the challenges faced by traditional intrusion detection systems in cloud settings and proposing innovative approaches to fortify the security posture of cloud-based infrastructures.

2. Literature Review

For identifying harmful capabilities from network activity, the NIDS paradigm was widely used. Data source, pre-processing, ruling processes, and protection measures are all part of the main SVM classifier. Initially, a data source is made up of a collection of internet connectivity, each of which was used to distinguish between worrisome and genuine observation. Also, before the predication NIDS then organises data by removing duplicate characteristics, resulting in a pattern collection including worrisome and lawful property-based actions. Finally, a detection model that detects anomalous data is included in the detection procedure. Finally, defence reaction is defined as a choice taken by cyber or application administrators to prevent an attack from occurring.

In [1] suggested the Intrusion, usage, and hybrids are the three types of NIDS techniques. To begin, the anomaly-based technique creates a regular pattern and identifies how it differs as from assault configuration. Because it can detect both zero-day and known threats, It requires no opportunity to establish rules and is far more effective for network avoidance than a misuse-based IDS when the choice system is effectively modelled. In [2], proposed Misuse oriented NIDS, on the other hand, monitors data for instances of known threats vectors that should be black- listed. It may not identify zero-day exploits, despite having reduced false alarm rates and better spam filtering. In addition, updating security settings with fresh attack criteria based on detected assaults necessitates a significant amount of effort. Several researchers have employed an aggregation model to optimize NIDS performances. The goal is to combine notifications for fewer alarms, making it easier for defense managers to manage alerts effectively. Using some of the traits, [3] suggested to build an ensemble strategy for determining varied assault kinds. Based on a voting rule-based technique, the author came to a final choice for forecasting assault. In [4] developed an ensemble strategy to anticipate attacking data using Classification Trees (CART) and Bayesian networks (BN). World at large, the factual outputs of these techniques show that jazz band techniques outperform individual models in terms of sheer performance. However, it adds to the computation power.

All NIDS productivity research was based on a supply that has a functionality which is categorised into many sorts of classifications objectives such as payload feature representation, reference port, flow, and behavioural feature models. In [5], research was based on a supply that has a functionality that was categorised into many sorts of classifications objectives such as payload feature representation, reference port, flow, and behavioural feature models.

In [6] proposed Netflow and Coral-Reef programmes that were used to extract some source/destination port number characteristics. Some aspects are ineffective, such as requiring valuable data from network packets, which is particularly inefficient in today's network to detect assaults with quicker and dynamically fluctuation toward the current network architecture. In [7], proposed the payload-based featured system collects a large number of signatures from varied uses. Some aspects aid in the detection of harmful functionality with greater precision. These are also certain autonomously terminals that are used and help with the providing of prediction while using non-standard connections. Furthermore, certain functionalities need a significant level of effort to upgrade identities on a regular basis in order to foresee assaults, as well as the complexity of collecting larger network traffic patterns.

Based on ports, targets, and guest behaviour patterns, behavioural aspects catch the attention of hosts. Finally, the baseline flow identities (protocols, ports, source/destination IP) and features are extracted such as data packets and inter-arrival periods are added to the stream space given. When utilised exactly as advised without expecting aggregates

more than a feature in extracting function to recognize malware methods happening by inundation of massive flows targeting infected machines, the two types achieve superior accuracy.

For analyzing the quality of NIDS, many scientists use data model sources from TCP/IP services. It focuses on detecting various types of exploits tactics and malware attacks, such as DDoS, DoS, and spamming models. For example, the author of [8] designed a scalable and flexible IDS using a set of HTTP, DNS, phishing scam, and Flow. This can cause components. It was devoid of characteristics that were required for execution. Furthermore, replicated data were obtained from a wide range of systems without a setup setting for estimating modern IDS performance.

In [9], suggests a text detection strategy based on DNS turbulence model and characteristics derived from DNS servers. This work was built using statistics extracted from DNS queries such as query web domain and originating Address. Also, with no further empirical routing aggregates analysis which involves possible elements of consolidated flooded dangers like DDoS assaults, they may be adjusted or masked via secure VPN use. In [10-14], proposed a domain name creation methodology for detecting malicious nodes by analysing individual DNS requests. This research made use of existing flow practices and methodologies, and the results were analysed using machine learning techniques. For the ruling process, the author was unable to employ consolidated flow that effectively detects botnet risks.

3. Proposed Methodology

Preliminaries

With today's modern technical advancements, a Network Intrusion Detection System (NIDS) is a technology that analyses and monitors an infrastructure to forecast aberrant functioning [15]. In the 1980s, Denning invented NIDS and predicted a prediction model for intrusion detection. It was often used in computer security breaches [16]. Unidentified operations have been on the demand, and traditional measures such as authorization and gateways have failed to protect networks against unknown assaults [17]. As a result, some researchers focused on the creation of CNN models or complex systems, such as computer crimes systems, value service telecommunication networks, number of co semantic web networking site structure [13], NIDS, and compassion cognitive theories. NIDS is gaining traction as a key source of SaaS systems on the basis of these safety applications. This is due to cloud needs for large-scale data transfer and engagement. Data is sent in a variety of formats. It was unavoidable that information privacy protection was continually compromised by incursion. NIDS was required throughout the building of cloud environment to provide security of information. As a result, our effort expects a new cloud-based NIDS secure environment.

Despite the importance of effective NIDS operation, establishing NIDS is thought to be even more difficult [18]. This is due to a number of crises, such as intrusion prevention and data gathering. These need to be taken into account. Following that, reference traffic statistics such as NSL-KDD and KDDCup 99 are created, and NIDS was modelled to improve intrusion detection performance [19] Various artificial intelligence (AI) and Machine Learning (ML) methodologies were employed in NIDS since intrusions identification was considered a major aspect of categorization.ML-based techniques were usually classified as either unsupervised (ULA) or supervised (SSA) (SLA).

The purpose of SLA is to align aspect data toward certain subcategories using labeled training data. To recognize and identify intrusions, several SLA such as Decision Tree (DT), Deep Neural Networks (DNN), and Support Vector Machine (SVM) have been utilized effectively. On a variety of standard datasets, the SLA for NIDS had been achieved with greater precision. Furthermore, several drawbacks were obvious. Processing labelled data initially necessitates expensive knowledge, and hence detection updation is costly. As a result, labelled data can completely distinguish newer types of threats when the training system relies on a different classifier. Similarly, to SLA, the ULA train recognition technique detects underlying unlabeled training organization in the absence of any labelled instances. Various types of network events were distinguished in ULA by evaluating unlabeled data distribution. The samples with comparable features were placed in the same class. Despite the fact that ULA does not require labelled data, it often led to an estimation method with lower power and a larger false positive rate (FPR).

In [20], used Semi-supervised learning method (SSLA) is also used for NIDS to eliminate different flaws. It determines the detector system by combining unlabeled and labelled data. As a result, SSLA is regarded to be healthier than SLA since it reduces the reliance on large dataset. SSLA often outperforms unsupervised learning in terms of consistency and FPA since it uses a less quantity of labelled data. Furthermore, SSLA suffers from the same set of drawbacks as the SLA and ULA models. SSLA for NIDS, on the other hand, requires a more comprehensive approach to reduce the detrimental impact of both techniques.

This research work proposed a unique SSLA for intrusion prevention in cloud systems by combining a Fuzzy based semi-supervised technique with Latent Dirichlet Allocation (F-LDA). The variation in classifying responses is reduced with this method. In a more suitable setting, the generalization ability surpassed the functioning of a singular model organism. Because there are a variety of attack methods that go undetected in learning algorithm,

composite SSLA is a better option. A simple learner is created as possibilities for labelled data and builds a hybrid method after class evaluation. However, a Fuzzy based model is employed to fully assess unlabeled distribution of the data. Then, using the outcomes of ULA, a new hybridized learned system is built and Latent Dirichlet Assortment is added. Finally, the proposed model was evaluated using the NSL-KDD dataset. The following are the key contribution only with path of economic development:

- This research introduced a new SSLA for categorization. This work employs classifier as a fundamental operator with hybridized system to analyze threat intelligence for non-linear classification method. To establish scores, the results of the Fuzzy model are combined and LDA is used.
- For analyzing hidden information of unlabeled data, this work uses a Fuzzy based approach. From large datasets, this approach retrieves important information and removes superfluous statistics. This can help the projected model function better.
- Through a fusion method, this research work incorporated both SLA and ULA. The labelled data must rectify the unidentified data in the process of this. This is due to a lack of labelled data; different resources is used to develop a classification technique, which then executes the detection method for consistency and quality.

The statistics Probabilistic terms of cognitive of linear Discriminant allocation is used to deduce the underlying semantics of a group of texts. The LDA model was built on the premise that the witnessed texts are generated from a list of topics which are random probabilities spanning words.

Table 1. Notations

S. No.	Notions	Description
1.	D	Total Documents
2.	K	Total Topics
3.	Nd	Total Words in the d-th Document
4.	V	List of Vocabulary
5.	w	Corpus Documents
6.	wd	D-th Documents
7.	wd, n	n-th word of d-th Document
8.	θ	Matrix for Topic Distribution of the Corpus
9.	θi	Vector of Topic Distribution of the Each Document
10.	Φ	Matrix Column denotes the Topic-Word Distribution
11.	ϕ i	Vector of the Word Distribution of the Each Topic

Fuzzy Integrated Latent Dirichlet Allocation Algorithm (FI-LDA)

This section discusses in full the uniqueness based on LDA and semi-supervised technique. This method is more effective, and it may be used to anticipate incursion. The duplicate data is deleted and the supervised learning is delivered. Fuzzy modelling is used to quantify chronotropy amongst attributes for unsupervised learning.

LDA provides insight into statistical modelling of corpus context identification. This is really a widely used statistical framework for detecting internet traffic and predicting malicious applications that has yet to be investigated. It's a probabilistic model since it's based on the assumption that the document was created using a weighted combination of unknown models. The end purpose of LDA is to recognise a group of classified data. An interpretive issue must be solved utilising corpus created by the LDA generation process in order to do this. For the creation of a collection of documents, this issue presents probabilities. The main goal is to assess the posterior probability of additional unknown parameters provided by the dataset as in Eq. (1):

$$p(\theta, \emptyset | D, \alpha, \beta) = \frac{p(\theta, \emptyset, D | \alpha, \beta)}{p(D | \alpha, \beta)}$$

For the purpose of testing LDA, Gibbs samples was utilised. Consider w and z to be vectors representing all received packets that is assigned to traffic "T." In Eq. (2), the inter dispersion of the predicted model is provided as follows:

$$p(z_t = k|z \rightarrow t, w) = \frac{n_{k,\rightarrow t}^{(w)} + \beta}{[\sum_{v=1}^{V} n_k^v + \beta] - 1} \frac{n_{k,\rightarrow t}^{(k)} + \alpha}{[\sum_{j=1}^{k} n_i^j + \alpha] - 1}$$

Here, $'t'$ specifies iteration argues against, $n_{k,\rightarrow t}^{(k)}$ is number of incoming statistics containers to complex except current traffic justification. $\sum_{v=1}^{V} n_k^v + \beta] - 1$ is total number of traffic owed to complex except in attendance set of connections analysis, $n_i^j + \alpha$ is number of inward bound packets to the complex. After succeeding giving out, the matrices are subtracted with Eq. (3) & (4):

$$\theta_{i,k} = \frac{n_i^{(k)} + \alpha}{\sum_{j=1}^{K} n_i^{(j)} + \alpha}$$

$$\emptyset_{k,w} = \frac{n_k^{(w)} + \beta}{\sum_{v=1}^{V} n_k^{(v)} + \beta]}$$

Correntropy is used to forecast the similarities and differences across attacker and typical occurrences by calculating the relationship between selected features. When the samples are different, statistic technique is used to evaluate the variable importance for forecasting dangerous functions. For predicting the interconnections of multiple feature measurements, it uses non- linear homology and 1st statistic. This is recognized as having a lower sensitivity to anomalies. Consider these two statistical properties, r_1 and r_2 with correntropy as Eq. (5):

$$V_\sigma(r_1, r_2) = E[K_\sigma(r_1 - r_2)]$$

Here, $E[.]$ is mathematical expectation, $K_\sigma(.)$ is Gaussian kernel function and σ is kernel size. It is depicted as in Eq. (6):

$$K_\sigma(.) = \frac{1}{\sqrt{2\pi\sigma}} \exp(-\frac{(.)}{2_{\sigma^2}})$$

The joint probability density function is generally unidentified when finite number of observations $\{r_i, r_j\}_{(i,j)=1}^{2}$ is attained. The correntropy is measured as in Eq. (7):

$$\hat{V}_{M,\sigma}(A, B) = \frac{1}{M} \sum_{i,j=1}^{M} K_\sigma\,(r_i - r_j)$$

When applying the pantries metric to variable selection data transmitted, as shown in Eq. (7), it is calculated for both aberrant and standard vector, as shown in Eq. (8):

$$I_{1:N} = \begin{bmatrix} f_{11} & f_{12} & \cdots \\ f_{21} & f_{22} & f_{ij} \end{bmatrix}; Y_{1,N} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$$

'I' represents an observer towards communication network, 'Y' represents a classification model for any and all observations towards different classifiers, 'N' means the number of observation, and 'F' represents the feature set. When the difference between legitimate and attack matrix values is seen, it may be concluded that the characteristics are significant. For both invasion and common occurrences, the correntropy of each variable is calculated. The distinctions between the situations are exposed.

Feature selection is very important in forecasting NIDS because it allows for the selection of key aspects and the elimination of extraneous values, which aid in the differentiation of dangerous and non-malicious occurrences and improves NIDS efficacy. The end goal of feature selection is to reduce computing costs, decrease redundant information, increase NIDS accuracy, and aid in dataset normalcy analysis. The coefficient value evaluates the degree of strength among particular attributes and is calculated using a basic feature selection technique. For anticipating aberrant functions of examples, the least rated attributes are considered as the most important portion of fuzzy modelling. The feature association coefficient is calculated.

$$CC(r_1, r_2) = \frac{cov(r_1, r_2)}{\delta r_1 . \delta r_2}$$

From the equation mentioned above, δ is standard deviation of features, cov () is feature covariance. The mean value of $r1$ and $r2$ are $Mr_1 = \frac{\sum_i^N a_i}{N}$ and $Mr_2 = \frac{\sum_i^N b_i}{N}$ respectively. The CC results have been altered to [+1,-1]. It defines correlation between two characteristics r1 and r2 when values are closer to +1 and -1. When results are significant to 0, there is no known association between characteristics. Data points specify characteristics in the same manner, whereas negative values specify qualities in the direction opposite.

LDA deals with unlabeled data that enters and exits the network. Fuzzy modelling is used to cope with this disorganized or unlabeled data. It's classified as an uncertainty kind, with a value ranging from 0 to 1. It's used for a variety of things, including categorization. To determine the importance of each observation, fuzzy modelling is used. As a result, it eliminates the network's produced uninteresting phrases or corpus. It is also used to improve generalization ability. As a result, the detection capability for newly announced harmful events increased.

The unlabeled statistics is defined as $S^a = \{x_1^a, x_2^a, \ldots, x_n^a\}\}$ with n_a. Here, correntropy representation is used for extracting features and to educate the fuzzy representation towards intrusion. With this, samples Sa is provide with group brand. The unlabeled samples are re-written with prediction label as which $S^{asl} =$

$\{x_1^a, \bar{x}(x_1^a), \ldots, (x_{n_a}^a, \bar{x}(x_{n_a}^a))\}$ is a self-labeled sample. Here, in turn entropy is used to work out fuzzy representation of classifier productivity. It is given as in Eq. (10):

$$F(x) = -\frac{1}{k}(\bar{x}(x) log_2 \bar{x}_i(x) + (1 - \bar{x}_i(x)) log_2(1 - \bar{x}_i(x)))$$

Self-labeled data are categorized based on ranking values to compute the fuzzy model. This concept is divided into three sections: the lower fuzzy set, the average fuzzy set, and the higher fuzzy set. The mean fuzzy set performs better when it comes to improving NIDS. As a result, these approaches only examine the overall fuzzy set, ignoring the upper and lower fuzzy sets. This model, like the learning phase, employs a ramping strategy to deal with data and build a fuzzy predictor. The median fuzzy set containing predictive values is used to train the classification model. To achieve homogeneous using fuzzy, the sampled rate is fixed and the quantity of entries is comparable to ❜.

For intrusion detection, the expected F-LDA is combined with a fuzzy system. This creates an SSL model using data that has been tagged. The fuzzy value was calculated using the projection of large datasets, and the sample is divided into three groups: higher, median, and less fuzzy sets. Fitting the average fuzzy set is used to achieve this. It is not possible to expand the classification algorithm when the inputs to the networks are limited. As a result, the fuzzy model is built correctly. As ahead and implement, the labelled samples are combined with an averaging fuzzy set. Finally, learning outcomes are defined as (.) for both unsupervised and supervised. To achieve improved generalization, the ability to combine the LDA-based network signatures with a fuzzy process is defined. This indicates a higher level of ability to recognize assault sequences. In the instance of unlabeled data, a fuzzy model may be used to investigate the inner workings of unsupervised learning. As a result, the fuzzy model offers entire dataset for categorization, allowing more data to be used. To improve system performance of a model, superfluous inbound data is removed. The complete detecting strategy becomes more robust and performs better with the merging of unsupervised and supervised models.

Fuzzy Integrated Latent Dirichlet Allocation Algorithm

Input: Labeled Samples

1. Set up the internet data flow
2. Use LDA to choose elements and delete those that aren't needed.
3. For i = 1,2, … , n
4. Create bootstraps based on the sample rate.
5. Develop a classifiers based on a neural model.
6. Divide the fuzzy model into three stages: lesser, median, and greater.
7. Compute ❜using Eq. (10)
8. End for
9. Choose the ❜model for generating prediction accuracy
10. Run the labelled data through the network flow to train it.
11. Create a semi-supervised system using a fuzzy model and an LDA.

Output: Evaluate the averaged fuzzy set method by predicting network traffic for vulnerability scanning.

4. Results and Discussion

In this research work, an internet traffic collection known as the NSL-KDD dataset is presented in conjunction with the expected model. Various testing assessments were carried out to assess the technique's usefulness. Both the testing and training sets were included in this database. The collection characterization was determined by the characteristics provided, together with preliminary statistical and content metadata for wired connection. The size of the feature is 41. Probe, routine, denial of service (DoS), user to admin, and faraway to regional are among the five offer tremendous opportunities labelled in the dataset (R2L). Several researchers regard the NSL-KDD dataset to be an accurate benchmark in malware detection. As a result, the NSL-KDD dataset is used in this research work to evaluate the semi-supervised technique. It consists of a number of attack methods that are more suited to testing generalization capacity. Random samples are picked at this point, and the other samples are used as unlabeled data.

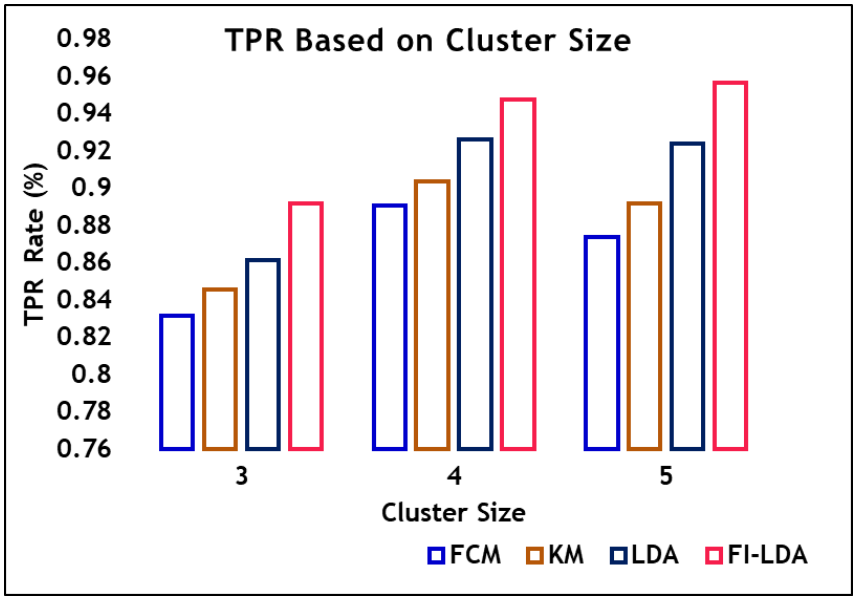
Intrusion detection is a multi-class issue in this work. The experimental investigation was carried out on a computer. Intel i5 CPU, Windows 7 OS, 8 GB RAM @3.00 GHz are the system specifications. U2R, and R2L are the two distinct qualities available. While anticipating higher size classes, the expected model operates finer and more consistently. The findings were compared to existing intrusion detection techniques to confirm the effectiveness of the expected model.

Various attacks define the values that represent the action where unauthorized users start accessing from a remote machine so as to exploit the host system's vulnerabilities. Attacks are less frequent attacks with 0.23% of the total attack space in training data. Fig 1 shows the results obtained on the basis of cluster size.

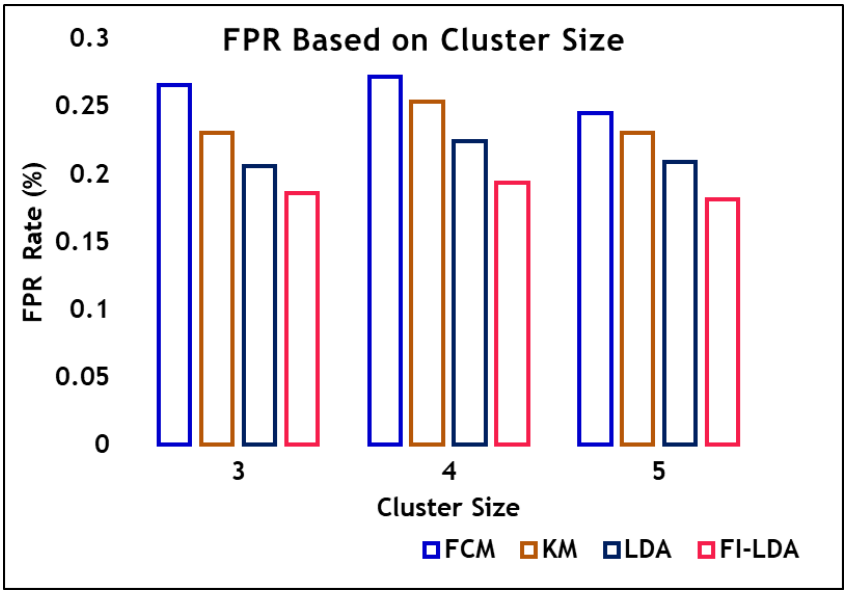
Table 2. Average Performance-based on Cluster Size for Attacks

A. TPR based on Cluster Size				
Cluster Size	FCM	KM	WLI	Proposed FI-LDA
3	0.83176638	0.84593438	0.86185317	0.89223647
4	0.89088319	0.90374644	0.92626515	0.94766049
5	0.87417974	0.89193732	0.92438746	0.95665716
B. FPR based on Cluster Size				
Cluster Size	FCM	KM	WLI	Proposed FI-LDA
3	0.26600476	0.2308612	0.20588785	0.18634731
4	0.27261905	0.25375494	0.22453782	0.19372093
5	0.24545455	0.2311054	0.20894737	0.18168831
C. Accuracy based on Cluster Size				
Cluster Size	FCM	KM	WLI	Proposed FI-LDA
3	0.88259629	0.89634259	0.90697293	0.92066999
4	0.86182336	0.89593728	0.90299572	0.94871795
5	0.89807692	0.90609687	0.9260913	0.96200998
D. Precision based on Cluster Size				
Cluster Size	FCM	KM	WLI	Proposed FI-LDA
3	0.84399524	0.85138804	0.88411215	0.93652695
4	0.87380952	0.89624506	0.90546218	0.94627907
5	0.85454545	0.88688946	0.90105263	0.95883117
E. F-Score based on Cluster Size				

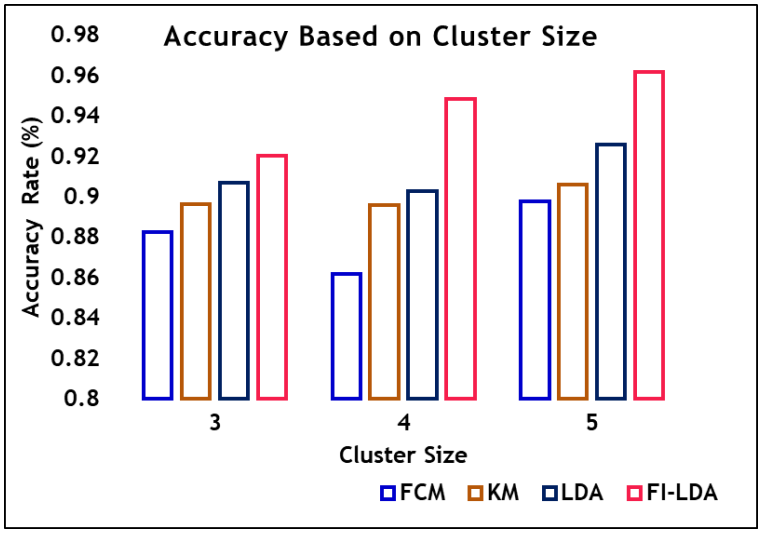
Cluster Size	FCM	KM	WLI	Proposed FI-LDA
3	0.83783619	0.848652	0.872841	0.913845
4	0.88226376	0.89998	0.915746	0.946969
5	0.8642511	0.889406	0.912571	0.957743



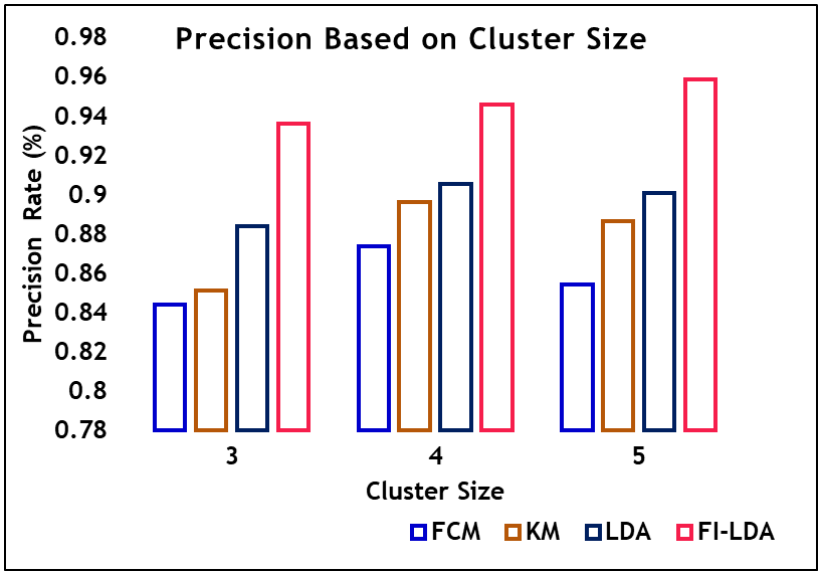
(A)



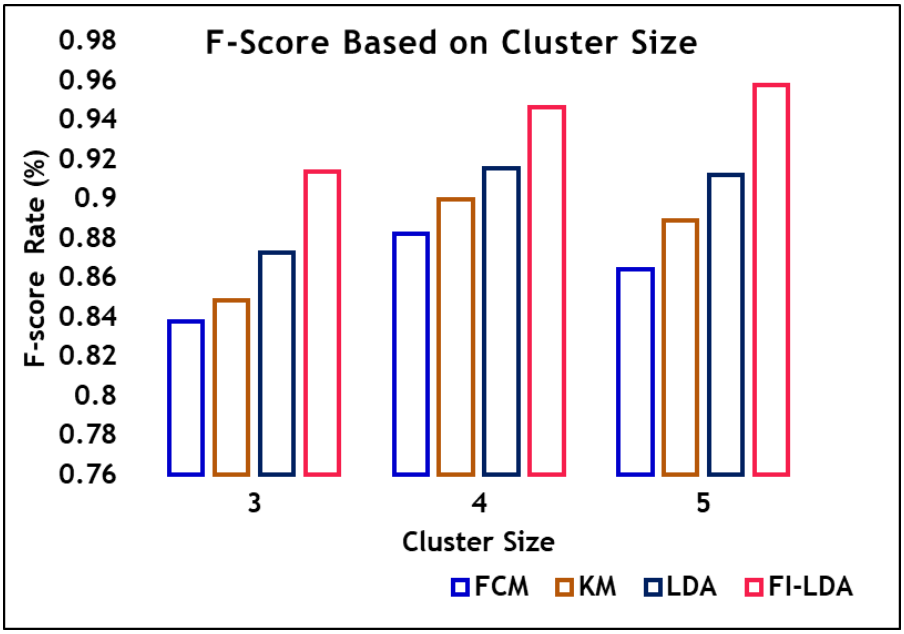
(B)



(C)



(D)



(E)

Fig 1. Average Performance-based on Cluster Size for Attacks

From the Fig. 1 and Table 2 and it is indicated that proposed FI-LDA is more capable to detect attacks that are less frequent than FCM, KMeans, and FDA. FI-LDA detects attacks with 95.55% true positive rate, 18.1 % false-positive rate, 96.20 % accuracy, 95.88% precision and .95 F-score value. And it is also evident that the attacks has been efficiently detected when the number of clusters is 5. For cluster size 5, Root to local attacks have been detected with the highest true positive rate and lowest false positive rate.

Fig 1(a) shows the True Positive Rate where FCM attains 87.76%, Kmeans attains 89.53% , FDA attains 90.63% and FI-LDA attains 96.38% True Positive Rate. From the Fig 1(a) it is evident that proposed FI-LDA achieves highest True Positive Rate. The Fig 1(b) shows the False Positive Rate where FCM records 24.23%, Kmeans records 23.97% , FDA records 23.96% and FI-LDA records 18.28% False Positive Rate. From the Fig 1(b) it is evident that proposed FI-LDA achieves lowest False Positive Rate. Fig 1(c) shows the detection Accuracy where FCM attains 89.61%, Kmeans attains 91.37% , FDA attains 93.65% and FI-LDA attains 95.42% detection Accuracy. From the Fig 1(c) it is evident that proposed FI-LDA achieves highest detection Accuracy. Fig 1(d) shows the Precision where FCM attains 88.01%, Kmeans attains 89.24%, FDA attains 92.76% and FI-LDA attains 96.59 % Precision. From the Fig 1(d) it is evident that proposed FI-LDA achieves highest Precision. Fig 1(e) shows the F-Score values where FCM attains 87.71%, Kmeans attains 89.27% , FDA attains 91.68% and FI-LDA attains 95.54 % F-Score value. From the Fig 1(e) it is evident that proposed FI-LDA achieves highest F-Score value.

5. Conclusion

In this research work, the FI-LDA model had been proved to be more productive in anticipating incursion and enhancing the security in cloud models via a continuing testing procedure. Three successive elements end the important addition world at large. Originally, a unique technique to incursion prevention provides a strong prediction model. It enhances the ability to recognize emerging travel patterns. Next, fuzzy modelling makes use of a large amount of unlabeled data to increase the system's accuracy rate and reliability. Varied elements were evaluated for ensuring cybersecurity in order to achieve the aim. Though it provided better trade off, there was a need for enhancement as there was a widespread flaw that makes it impossible to detect assault patterns. The FI-LDA demonstrates commendable effectiveness in identifying and detecting malicious activities within the cloud environment, showcasing strong overall performance.

References

[1] Ajinkya Wankhade & Chandrasekaran, K 2016, 'Distributed-intrusion detection system using combination of ant colony optimization (ACO) and support vector machine (SVM)', International Conference on Micro- Electronics and Telecommunication Engineering, pp. 646-651,

[2] Alampallam Ramaswamy Vasudevan & Subramanian Selvakumar 2016, 'Local outlier factor and stronger one class classifier based hierarchical model for detection of attacks in network intrusion detection dataset', Frontiers of Computer Science, vol. 10, no. 4, pp. 755-766.

[3] Aman Bakshi & Yogesh, B 2010, 'Securing cloud from DDOS attacks using intrusion detection system in virtual machine', International Conference on Communication Software and Networks, pp. 260-264.

[4] Amin Nezarat & Yaser Shams march 2017, 'A game theoretic-based distributed detection method for VM-to-hypervisor attacks in cloud environment', Journal on Super Computing, Springer, vol. 73, no. 10, pp. 4407-4427.

[5] Amir Vahid Dastjerdi, Kamalrulnizam Abu Bakar & Sayed Gholam Hassan Tabatabaei 2009, 'Distributed intrusion detection in clouds using mobile agents', International Conference on Advanced Engineering Computing and Applications in Sciences, pp. 175-180.

[6] Anandapriya, M & Lakshmanan, B 2015, 'Anomaly based host intrusion detection system using semantic based system call patterns', IEEE International Conference on Intelligent Systems and Control, pp. 1-4.

[7] Andrey Rukavitsyn, Konstantin Borisenko & Andrey Shorov 2017, 'Self- learning method for DDoS detection model in cloud computing', IEEE Conference on Russian Young Researchers in Electrical and Electronic Engineering, pp. 544-547.

[8] Aqeel Sahi, David Lai, Yan L & Mohammed Diykh 2017, 'An efficient DDoS TCP flood attack detection and prevention system in a cloud environment', IEEE Journal and Magazines, vol. 5, pp. 6036-6048.

[9] Ashish Kumbhare & Manoj Chaudhari 2014, 'IDS: Survey on intrusion detection system in cloud computing', International Journal of Computer Science and Mobile Computing, vol. 3 no. 4, pp. 497-502.

[10] Ashwini Khadke & Mangala Madankar 2016, 'Review on mitigation of distributed denial of service (DDoS) attacks in cloud computing', International Conference on Intelligent Systems and Control, pp. 1-5.

[11] Berkah Santoso, I, Rien Suryatama Idrus, M & Irwan Prasetya Gunawan 2016, 'Designing network intrusion and detection system using signature- based method for protecting openstack private cloud', International Annual Engineering Seminar, pp. 61-66.

[12] Chandrashekhar Azad & Vijay Kumar Jha 2017, 'Fuzzy min–max neural network and particle swarm

optimization based intrusion detection system', Technical Paper on Microsystem Technologies, vol. 23, no. 4, pp. 907-918.

[13] Chi-Chun Lo, Chun-Chieh Huang & Joy Ku 2010, 'A cooperative intrusion detection system framework for cloud computing networks', International Conference on Parallel Processing Workshops, pp. 280-284.

[14] Chirag Modi, N, Dhiren Patell, R, Avi Patd & Rajarajan Muttukrishnan 2012, 'Bayesian classifier and snort based network intrusion detection system in cloud computing', IEEE Third International Conference on Computing, Communication and Networking.

[15] Chithik Raja, M & Munir Ahmed Rabbani, M 2016, 'Combined analysis of support vector machine and principle component analysis for IDS', International Conference on Communication and Electronics System, pp. 1-5.

[16] Chunjie Zhou, Shuang Huang, Naixue Xiong, Shuang-HuaYang, Huiyun Li, Yuanqing Qin & Xuan Li 2015, 'Design and analysis of multimodel- based anomalyintrusion detection systems in industrial process automation', IEEE Transaction on Systems, Man and Cybernetics Systems, vol. 45, no. 10, pp. 1345-1360.

[17] Claudio Mazzariello, Roberto Bifulco & Roberto Canonico 2010, 'Integrating a network IDS into an open source cloud computing environment', International Conference on Information Assurance and Security, pp. 265-270.

[18] Dataset: Available from: <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>.

[19] Fadwa Abdul Aziz Alseiari & Zeyar Aung 2015, 'Real-time anomaly- based distributed intrusion detection systems for advanced metering infrastructure utilizing stream data mining', International Conference on Smart Grid and Clean Energy Technologies, pp. 148-153.

[20] FarzanehIzak Shiri, Bharanidharan Shanmugam & Norbik BashahIdris 2011, 'A parallel technique for improving the performance of signature- based network intrusion detection system', International Conference on Communication Software and Networks, pp. 692-696.