JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 38, XXX-XXX (2022)
DOI: 10.6688/JISE.20220X_38(X).00XX

Anomaly Detection in Dynamic Cloud Environment Using Density Estimated LSTM and Stochastic Krill Herd*

SHEELA S[1*], N. SUBBULAKSHMI[2]
[1*]*Kalasalingam Acadamy of Research and Education,*
Department of Computer Application,
Krishnankoil, Tamil Nadu, India
[1*]*scholar.sheelas@gmail.com*
[2]*Kalasalingam Acadamy of Research and Education,*
Department of Computer Science and Engineering,
Krishnankoil, Tamil Nadu, India
[2]*scholar.nsubbulakshmi@gmail.com*

Anomaly detection in cloud computing is a critical component of guaranteeing the security and dependability of cloud-based services and infrastructures. It entails the detection of anomalous or unexpected behavior that deviates from the established patterns of typical operation inside a cloud environment. The analysis of massive volumes of data produced by multiple cloud components (such as virtual computers, networks, and storage resources) uses complex algorithms and machine-learning approaches. Even though, these works have shortfalls in establishing accurate baselines for normal behavior in a landscape where resources and workloads are constantly changing based on demand, and further the adaptability and learning capabilities necessary to recognize emerging threats. To overcome these challenges, a novel Density Estimated LSTM and Stochastic Krill Herd algorithm are implemented, in which Density Estimation LSTM is developed to evaluate the density distribution of cloud structures by analyzing the probability distribution from the output of LSTM, thereby capturing the changes and variations in cloud patterns. Further, for detecting the anomaly from the emerging threats, the Fractional Stochastic Krill Herd Algorithm (FSKHA) is implemented, which incorporates the behavior of krill swarms and incorporates stochastic elements, thereby recognizing both known and unknown anomalies that are emerging with the new data and trends. The experimental outcomes gained from the proposed model have effective performance in terms of low false alarm rate, higher detection rate, and accuracy.

Keywords: Cloud computing, Anomaly detection, Network Security, Dynamic Cloud Environment, Krill Herd Algorithm, Long Short-Term Memory

1. INTRODUCTION

The paradigm of cloud computing [1-2] has completely changed how organizations and people manage and access digital resources. Fundamentally, cloud computing entails the internet-based delivery of a range of computer services, including servers, storage, databases, networking, software, analytics, and intelligence. In contrast to conventional on-premises computing, which relies on locally installed hardware and software, cloud computing makes use of a massive network of remote servers housed in data centers across the world. Cloud service providers like Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform are in charge of connecting and running these computers. Scalability is one of the main benefits of cloud computing. Depending on their current demands, users can simply modify their computing resources, such as processing power and storage. With this flexibility, organizations can effectively manage changing

workloads and make sure they only pay for the resources they utilize. Additionally, cloud computing [3-4] encourages cost-effectiveness because it does away with the requirement for sizable upfront investments in infrastructure and technology. Instead, consumers pay for a pay-as-you-go or subscription-based approach, which may be more cost-effective in the long term.

Cloud computing's accessibility and capacity [5-6] for remote collaboration are important additional features. Users can access their data and applications from any location with an internet connection, making it possible for workers to be more mobile and connected on a global scale. This has become increasingly important recently as remote work has gained popularity as a method of operation. Cloud computing also provides improved data protection and security safeguards. To protect sensitive data, reputable cloud service providers utilize strong security mechanisms, such as encryption, firewalls, and multi-factor authentication. To make sure businesses follow industry norms and laws, they also make significant investments in compliance certifications and submit to regular audits.

The flexibility and responsiveness of cloud computing [7-8] to shifting needs and circumstances in real-time are referred to as its dynamic nature. Cloud infrastructures are made to be more flexible and scalable, unlike conventional on-premises infrastructure, which necessitates manual modifications and actual hardware updates. To fulfill the particular requirements of an application or workload, resources like computing power, storage capacity, and networking capabilities can be readily added or decreased. This ability to adapt is especially important in the fast-paced and constantly changing digital environment of today, when organizations must act quickly in response to changes in consumer demand, seasonal spikes, or unanticipated traffic surges. Additionally, cloud service providers [9] regularly update and improve their offerings, incorporating new features and technologies, ensuring that consumers have access to the most recent advancements without the need for significant hardware upgrades. Because cloud computing is dynamic, it not only improves operational effectiveness but also gives businesses the tools they need to stay adaptable and competitive in a quickly evolving technology landscape.

The dynamic nature of cloud formation [10-11] presents several key obstacles for prediction. The intrinsic complexity of cloud settings, which include a wide range of interrelated elements, from virtual machines to networking configurations, is one of the main problems. It is challenging to precisely predict how changes in one region can affect the system as a whole due to this complexity. Additionally, user needs and traffic patterns are prone to change in cloud systems. These elements can be influenced by a wide range of external factors, including seasonality, marketing initiatives, and unplanned events. It is difficult to predict resource requirements correctly because of this fluctuation. Additionally, new services or features offered by cloud providers as well as technical improvements may create additional uncertainty.

Detecting anomalies [12-13] in dynamic cloud systems entails finding out-of-the-ordinary or abnormal activity inside the complicated, constantly shifting cloud computing environment. The sheer size and complexity of cloud infrastructures, which can make it challenging to define baseline behavioral patterns, is one of the major issues in this context. It becomes difficult to define what constitutes a typical state because cloud resources are continually changing. Furthermore, advanced algorithms and procedures are needed to

DOI: 10.6688/JISE.20220X_38(X).00XX

discriminate between benign alterations and possibly hazardous anomalies. Additionally, creating a universal anomaly detection system can be difficult due to the wide variety of services and components found in a cloud context. The whole extent of a dynamic cloud may not always be covered by detection approaches that are labor-intensively tailored to certain services or configurations. Real-time detection [14-15] is also essential for swiftly mitigating any damage, but it might be hampered by the enormous amount of data that needs to be processed and examined. A significant problem in guaranteeing the security and stability of cloud-based systems is finding a balance among accuracy and performance in anomaly detection in dynamic cloud topologies. Hence, there is a need for improved techniques in this advanced world for detecting the anomalies in dynamic cloud structure is crucial. The main contribution of this work is as follows:

- To capture and analyze the dynamic change in cloud structure, a novel Density Estimation LSTM is used, which in turn captures the normal behavior of cloud structure within the dynamic changing condition based on the relevant parameters.
- To identify the anomalies in the dynamic cloud structure, an innovative FSKHA is introduced, which enhances the traditional SKH by introducing fractional decision variables and stochastic elements to improve the detection of both known and emerging anomalies.

The paper content is developed as follows: section 2 offers the current works on literature, section 3 evaluates the process flow and unique characteristics of the proposed Density Estimated LSTM and Stochastic Krill Herd Algorithm, section 4 deliberates the result and comparison and finally, section 5 discusses the conclusion.

2. LITERATURE SURVEY

Qureshi et al. [16] suggested an edge computing-based system architecture for IoT networks called Software-Defined Network-based Anomaly Detection System (SDN-ADS). Then, for SDN and edge computing networks, an anomaly detection system has been suggested to identify the behavior of the device. To guarantee the confidence of edge devices for data forwarding, a Trusted Authority for Edge Computing (TA-Edge) has been proposed. As a certificate authority for the designated trusted domain, the edge device was in operation. In the TA-Edge paradigm that has been proposed, the edge node only needs to verify the certificate once to establish trust, after which all communication can be done using local certificates. The proposed method will be tested in the future using more attacks, and hence make sure to cover more intricate networks and keep an eye out for criminal activity.

El-Shamy et al. [17] introduced a support vector machine-based monitoring approach to find the distributed application's bottleneck in the cloud data center and discover performance anomalies. To train the SVM algorithm and create a baseline model of the typical behavior of the distributed application, it gathers data from network devices and produces performance metrics for the distributed application components. The SVM model uses the one-class support vector machine (OCSVM) and multi-class support vector machine (MCSVM) algorithms to detect performance abnormal behavior and pinpoint the source of bottlenecks. The suggested approach does not rely on static threshold settings for performance assessments or call for any prior information about the currently running

apps. Future research seeks to improve data center infrastructure metrics with a focus on end-host factors by including new machine-learning models.

Xu et al. [18] presented a benchmark called StreamAD to help Site Reliability Engineers (SREs) choose appropriate anomaly detection techniques based on particular use cases. This benchmark provides three benefits: It includes eleven unsupervised algorithms with open-source code, abstracts several common operators for online anomaly detection to increase algorithm creation speed, and offers thorough comparisons of various algorithms using various evaluation techniques. Researchers efficiently evaluate novel algorithms in-depth with StreamAD, which helps to advance this field of study. The most recent research will be incorporated into StreamAD in the future, and benchmarks will also be assessed from more angles in order to assess how well different algorithms perform at detection and how easily they can be understood.

Garg et al. [19] proposed an Ensemble Artificial Bee Colony Anomaly Detection Scheme (En-ABC) for multi-class datasets in a cloud context. To identify malicious node behavior, En-ABC has the following components: feature selection and optimization, data clustering, and identification of abnormal node behavior. Restricted Boltzmann Machine and Unscented Kalman Filter, respectively, were used to build the feature selection and optimization models in En-ABC. Additionally, the Mean Square Deviation and Dunn Index were employed to obtain an appropriate clustering based on the ABC-based Fuzzy C-means clustering technique. The results of the clustering have then been used to create a profile of normal and aberrant behavior for the detection of anomalies. Future evaluations of alternative evolutionary algorithms will compare the performance with the suggested method in the same hybridization situation.

Wang et al. [20] implemented an adaptable architecture for stream processing so that IIoT applications detect anomalies online. The framework used a Docker-based distributed computing architecture to increase flexibility, adaptability, and customization. To coordinate data stream processing jobs operating on several docker nodes, the framework additionally made use of a central mediator. Additionally, a batch model training and data stream anomaly detection process-based prediction-based online anomaly detection model has been created. The model employs long short-term memory (LSTM) neural networks to forecast values in the data stream and a dynamic sliding window approach to simulate and identify prediction errors. Future research may look in the following directions: investigating various deep learning architectures and techniques; and incorporating domain expertise and contextual information in the anomaly detection process.

Yang et al. [21] adopted a brand-new technique for detecting abnormal network traffic in a cloud computing setting. Six types of network traffic features were taken into consideration in this work, including the number of source IP addresses, number of source port counts, number of destination IP addresses, number of destination port counts, number of packet types, and number of network packets. The framework of the anomaly network traffic detection system was first illustrated. Second, by normalizing the values of network feature values and utilizing SVM to identify anomalous network behaviors, a novel hybrid information entropy, and SVM model has been presented to address the proposed problem. Other network functions will be added as part of this future work and further discussed about whether or not the suggested approach can be used in parallel.

Mahdavi et al. [22] provided an anomaly-based DDoS attack detection methodology

for cloud environments by employing a third-party auditor (TPA). Next, a variety of fundamental presumptions and cloud environment setups for creating simulation tests were offered to assess the suggested framework. Then, the outcomes of simulation experiments were presented for evaluating the viability of this strategy. The experiment thus explained this identification of DDoS assaults in CSPs by efficiency, rapidity and precision. Future research is needed on the topic of analyzing the effects of various CSPs' deployment methods in detecting DDoS assaults using TPANG.

Savaridassan et al. [23] evaluated an Integrated Deep Auto-Encoder and Q-learning-based Deep Learning (IDEA-QLDL) Scheme to achieve the highest prediction accuracy while examining log data and identifying it as authentic or anomalous. Based on ongoing research into behavioral patterns that were highly suitable for classification, it starts the process of acceptance or denial. The proposed IDEA-QLDL Scheme's performance results confirmed that it outperformed the benchmarked schemes under consideration in terms of classification accuracy, precision, recall, and detection time. The architecture of ResNets and AlexNets will be used to create various anomaly detection techniques in the near future.

Zhang et al. [24] proposed an automatic multi-view feature fusion and discriminative model optimization for increasing the accuracy. To increase detection effectiveness, this model makes use of extreme learning machines (ELM). ELM was a single hidden layer neural network that avoided the local optimal solution by converting the iterative solution of the output weights to the solution of linear equations. Additionally, the weights were applied for ranking anomalies with respect to the distance between samples and the classification boundary before retraining the classification model. This proposed approach effectively utilizes the complement information between subsystems, eliminates the influence of imbalance distribution, and addresses a variety of cloud computing platform issues. The future research is to overcome the overfitting problem and reduce the computational requirements.

Nawrocki et al. [25] presented a novel hybrid anomaly detection system and a new method for automatic long-term cloud resource utilization planning. It examined the current solutions for anomaly detection, potential upgrades, and the effect on the precision of resource utilization planning. The proposed anomaly detection method was a crucial component of the research since it enables long-term accuracy improvements. The suggested method dynamically modifies reservation plans to lessen the needless demand for resources and avoid the cloud running out of them. Further study could potentially result in higher accuracy by dynamically estimating confidence based on the performance of each sub-algorithm.

Rahumath et al. [26] offered resource scalability and security utilizing Entropy-based Adaptive Krill herd optimization for auto-scaling in cloud infrastructure. To do this, trust-based anomaly detection was first established. The scheduler schedules the job in response to the anomaly detection. Following that, the scheduled jobs were scaled based on execution time predictions, and workload predictions were completed. Finally, the scaled data was optimized using the entropy-based krill herd technique. The Krill Herd algorithm has difficulty adapting to quickly changing environments or dynamic workloads. In cases where resource needs shift often, the algorithm struggles to successfully re-optimize resources

Sivamohan et al. [26] presented TEA-EKHO-IDS, a unique intrusion detection

system that used enhanced krill herd optimization (EKHO) and reliable explainable artificial intelligence (XAI) to find cyber-physical systems breaches. By calculating the decision weighting factor, the suggested technique used XAI-EKHO for feature selection, which provided more robust global searching capabilities and quicker convergence time. By combining explainable AI, bi-directional LSTM, and Bayesian optimization (BO-Bi-LSTM) for effective detection and classification, intrusion detection performance was maximized. However, the slower convergence speed of the KH method results in longer processing times, which is important in real-time applications that need quick responses, such as intrusion detection systems. This is especially true in complicated search spaces.

The above discussion stated that [16] needs to be tested in the future using more attacks, and hence make sure to cover more intricate networks and keep an eye out for criminal activity. For [17], future research aims to enhance data center infrastructure metrics by incorporating new machine-learning models that focus on end-host factors. For [18], the most recent research will be incorporated into StreamAD in the future. For [19], in the same hybridization scenario, future studies of various evolutionary algorithms will compare their performance. For [20], further process the work by investigating various deep learning architectures and techniques and incorporating domain expertise and contextual information in the anomaly detection process. For [21], other network functions will be added as part of this future work and further discussed about whether or not the suggested approach can be used in parallel. For [22], more research on analyzing the effects of various CSPs' deployment methods in detecting DDoS assaults using TPANG. For [23], the architecture of ResNets and AlexNets will be used to create various anomaly detection techniques in the near future. For [24], future research is to overcome the overfitting problem and reduce the computational requirements. For [25], further research could enhance accuracy by dynamically estimating confidence based on the performance of each sub-algorithm. For [26] KH algorithm has difficulty adapting to quickly changing environments or dynamic workloads and in [27] slower convergence speed of the KH method results in longer processing times.

3. ANOMALY DETECTION USING DENSITY ESTIMATED LSTM AND STOCHASTIC KRILL HERD ALGORITHM

Anomaly detection is a crucial method in cloud-based services to ensure security and reliability. It uses advanced algorithms and machine learning techniques to analyze large volumes of data generated by various cloud components. These algorithms learn from historical data to establish a baseline of normal behavior and continuously monitor for deviations. Anomalies can include unusual network traffic, resource consumption, or access patterns. By identifying anomalies, network providers can take proactive measures to mitigate potential security breaches, performance issues, or system failures, ensuring the integrity, availability, and confidentiality of cloud services. To do this, an innovative concept is introduced named “Density Estimated LSTM and Stochastic Krill Herd”. Moreover, in anomaly detection, the dynamic nature of cloud environments presents a significant challenge, because it is difficult to establish accurate baselines for normal behavior in a landscape where resources and workloads are constantly changing based on demand. To overcome this issue of anomaly detection because of the changing cloud environment, a breathtaking novel change detection algorithm called Density Estimation

DOI: 10.6688/JISE.20220X_38(X).00XX

Long Short-Term Memory, which captures long-term dependencies and sequential patterns in time series data, by training an LSTM model on historical cloud data. The DE-LSTM model estimates the density distribution of cloud structures by learning the underlying probability distribution from the training data. This enables the model to estimate the density of future data points by capturing the variations and changes in cloud patterns, such as the formation, dissipation, and movement of clouds in real-time. Further, the traditional methods lack the adaptability and learning capabilities necessary to recognize emerging threats, as they can't update themselves based on new data and trends, which become outdated earlier in the face of evolving attack strategies. For identifying emerging threats in the anomaly detection process, an exceptional concept is presented a novel Fractional Stochastic Krill Herd Algorithm, which incorporates the behavior of krill swarms and incorporates stochastic elements for improved performance. This algorithm extends the traditional SKH by introducing fractional decision variables, which enhance the algorithm's flexibility and performance in detecting both known and unknown anomalies. FSKHA differs from existing SKH approaches by incorporating stochastic differential equations, which allow the algorithm to better adapt to emerging threats in real-time environments. This enhances the algorithm's ability to detect emerging threats that not be easily identified using traditional methods and is effective in cloud computing by capturing both known and unknown anomalies.

The step-by-step procedure of the dynamic cloud nature and its anomaly detection in cloud computing is given in Fig. 1.

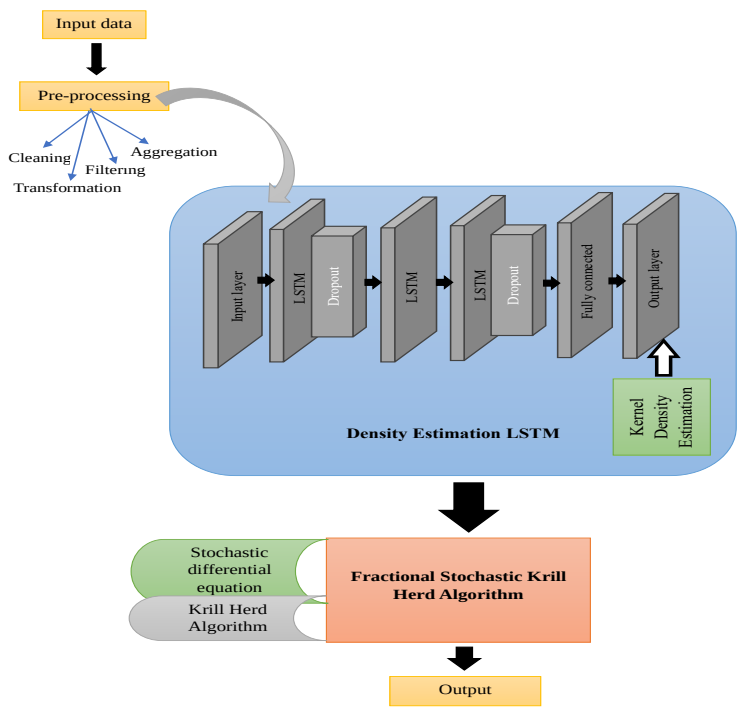

Fig. 1. Architectural diagram of Anomaly Detection in Dynamic Cloud Environment

DOI: 10.6688/JISE.20220X_38(X).00XX

3.1. Data Collection

Data collection is the initial step of the process, which is done by compiling data from various sources within the cloud environment that includes metrics, which provide quantifiable measurements of resource utilization and performance, events, which document notable incidences or changes, logs, which record various actions and events, and network traffic, which includes source and destination IP addresses, protocols, data volume, and ports.

3.2. Data Preprocessing

Data preprocessing involves steps like cleaning, filtering, aggregation, and transformation, where data discrepancies are fixed by cleaning out any incorrect or inconsistent entries. Filtering is used to obtain only the pertinent data by removing unnecessary or redundant records. Aggregation combines data elements to create more thorough summaries, which makes it easier to spot larger trends and patterns. Finally, transformation involves transforming data into a standardized format or normalizing it to a constant scale to facilitate precise and insightful analysis. This preprocess procedure is thus crucial for getting the data ready for the next phases and making sure it is trustworthy and properly formatted for anomaly identification in the dynamic cloud environment.

3.3. Change Detection using Density Estimation LSTM

The pre-processed data is then used for analyzing the dynamic changing behavior of the cloud environment, where a change detection algorithm named Density Estimation LSTM is deployed, LSTM is well known for capturing the long-term dependencies and sequential patterns in time series data, and then the density estimation is used to estimate the probability distribution of the predicted values. The Density Estimation LSTM structure is explained in Fig. 2.

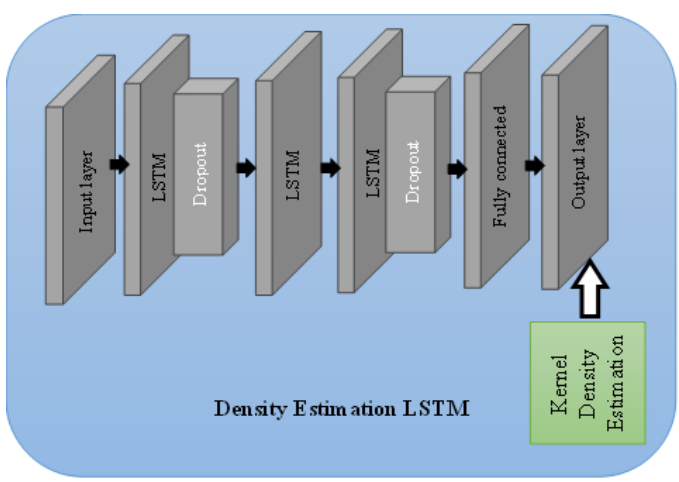

Fig. 2. Architectural Diagram of Density Estimation LSTM

The model uses time-series data as input to forecast future states based on previous

DOI: 10.6688/JISE.20220X_38(X).00XX

findings. The Density Estimation LSTM consists of a standard LSTM architecture with an additional output layer incorporating kernel density estimation to characterize the distribution of prediction errors. The LSTM architecture is made up of numerous layers, including an input, LSTM units, fully linked layers, and a density estimation layer. The LSTM model alternatively be seen as a function $g(.)$ that uses past values to forecast future ones. The following equation (1) $\grave{y_{ti}}$: illustrates how the model uses the previous m time-series data to estimate the value at the time ti.

$$\grave{y_{ti}} = g(y_{ti-1}, y_{ti-2}, \dots, y_{ti-m}) \tag{1}$$

A given time series, $y_1, y_2, \dots, y_m$, must be segmented into discrete sub-sequences using a sliding window of length m to train the LSTM model. Eq. (2) shows the resulting sub-sequences of observations, Y, and labels, Z, for training. Each time a training iteration is finished, a message queue is added with the newly created model. The message is then consumed by the change detection program, which uses it to get the most recent model for online anomaly detection. The training procedure takes place on a single machine, even when the model is disseminated over several machines. Message queues are used to deliver model parameters and data streams to various machines.

$$Y = \begin{bmatrix} y_{ti-m} & \cdots & y_{ti-2} & y_{ti-1} \\ y_{ti-m+1} & \cdots & y_{ti-1} & y_{ti} \\ \vdots & \vdots & \vdots & \vdots \\ y_{n-m} & \dots & y_{n-2} & y_{n-1} \end{bmatrix} \tag{2}$$

Finding the right time to update the model is crucial in the context of dynamic change detection. Iterative update (IUpdate), regular update (RUpdate), and demand update (DUpdate) are the three most used approaches for updating models. The model is updated via an iterative process per the training cycle. In contrast, the regular technique updates the model every minute, hour, or day at predetermined intervals. Finally, the demand method changes the model in response to cues from the online algorithm, such as when the total prediction error exceeds a preset threshold value expressed in terms of root mean square error (RMSE) or mean square error (MSE). The LSTM model is trained to minimize the difference between predicted and actual values, usually using the MSE loss function. The MSE is computed as follows in equation (3):

$$\mathcal{L}_{MSE} = \frac{1}{n}\sum_{i=i}^{n}(y_{ti} - \hat{y}_{ti})^2 \tag{3}$$

Where y_{ti} is the actual value at time t_i, $\hat{y}_{ti}$ is the predicted value at time t_i, and n is the number of predictions. Minimizing this loss function helps the LSTM model improve its accuracy in predicting future cloud behaviours.

Sub-sequences of length $m + 1$ are fitted into the model to train the LSTM network, allowing us to make short-term predictions and retrieve the projected time series, $\hat{Y} = (\hat{y}_1, \hat{y}_2, \dots, \hat{y}_{ti})$. First, the point-wise difference (or error) $err_{ti} = y_{ti} - \hat{y}_{ti}$ | between the ground truth and the anticipated value at each time step ti is calculated. However, concept drift may appear in data streams over time, presumably as a result of modifications to the contextual environment. For that, a so-called flexible sliding window approach is used for distribution modelling to boost adaptability. In this method, mistakes inside a sliding window of adjustable size are modeled. The following equation (4) is a definition of a flexible sliding window.

DOI: 10.6688/JISE.20220X_38(X).00XX

$$flex = \begin{cases} [ti_{ch}, ti], & ti_{ch} < ti - m_{min} \\ [ti - m_{min}, ti], & ti_{ch} \geq ti - m_{min} \end{cases} \quad (4)$$

By adjusting the window size based on context, the flexible sliding window enables the model to dynamically adapt to changes in the surrounding environment. This flexibility makes ensures long-term trends and slow changes in behavior are recorded without losing track of significant events.

Once the LSTM model has been trained and predictions have been made, Kernel Density Estimation is applied to the prediction residuals (errors) to estimate their probability density function. Here, as a novelty, kernel density estimation is used within the output layer of LSTM to characterize the errors and the distribution density function to generate a change detection score that represents the probability. As a result, a time series' points have each assigned a change detection score that indicates how much the features of the cloud have changed at that particular point. The dynamic metrics are defined as the points that have change detection scores that surpass a given threshold.

Let $u_1, u_2, \ldots, u_n$ be identically and independently distributed samples taken at any given point u from a univariate distribution with an unknown density d. Its estimation of kernel density is given in Eq. (5).

$$\hat{d}_b(u) = \frac{1}{j}\sum_{i-1}^{j} Ke_b\,(u - u_i) = \frac{1}{jb}\sum_{i-1}^{j} Ke\left(\frac{u-u_i}{b}\right) \quad (5)$$

Where, a non-negative function kernel is Ke and the bandwidth is given as the smoothing parameter as $b > 0$. Hence, this density estimation LSTM thus compares the density estimation results with the changing threshold, and the data points that fall below or exceed the threshold are identified as the changing pattern of the cloud structures, which is taken as the normal behavior. The patterns that do not match this change behavior are defined as anomalies in the changing cloud structure, which is explained in the next section as detail.

3.4. Anomaly detection using Fractional Stochastic Krill Herd Algorithm

To detect the emerging threats in dynamic cloud structure, because of the emerging of new data and trends, FSKHA has evolved that is a variant of the Krill Herd Algorithm (KHA) that is designed to handle optimization problems with continuous decision variables, where solutions can take fractional values. The SKH algorithm has an ability to efficiently explore the search space and local optima by incorporating stochastic elements. The reason behind selecting SKH is its distinct capacity for imitating the collective feeding habits of krill, hence facilitating efficient exploration and utilization of the search domain. In dynamic cloud systems, where adaptation is essential this feature is very helpful for anomaly detection. Furthermore, its fractional variations generalize smooth integration. FSKHA operates by modeling normal behaviour as the "optimal" state in the context of anomaly detection and recognizes deviations from this norm as potential anomalies.

Motion tempted by other krill individuals: Individual krill retain a high density and migrate in a motion direction (β_j) as a result of mutual effects assessed by local, target, and repulsive swarm densities stochastically and is given as per Eq. (6)

$$P_j^{New} = P^{Max}\beta_j + \delta_n P_j^{Old} \quad (6)$$

Where β_j is given in Eq. (7), where the novelty is added by using the stochastic differential equation.

$$\beta_j = \mu(\beta_j^{Loc}) + \sigma(\beta_j^{Tar}) \quad (7)$$

DOI: 10.6688/JISE.20220X_38(X).00XX

Where, $\mu = 0.1$ and $\sigma = 0.2$ are the expectation and variance in stochastic differential equations, and are independent of the process's past behavior. The extreme-induced speed is P^{Max}, the motion's inertia weight is δ_n tempted between [0,1], the last motion tempted is P_j^{Old}, provided neighbor's local effect is β_j^{Loc} and the provided target direction effect by the best krill individual is β_j^{Tar}.

For a local search β_j^{Loc}, the influence of the neighbours might be interpreted as an attractive/repulsive tendency between the individuals and is given in Eq. (8). According to this study's findings, a krill movement individual's neighbours have the following effects.

$$\beta_j^{Loc} = \sum_{k=1}^{NN} \hat{L}_{j,k}\, \hat{Y}_{j,k} \tag{8}$$

Where $\hat{Y}_{j,k}$ and $\hat{L}_{j,k}$is given in Eq. (9) and (10)

$$\hat{Y}_{j,k} = \frac{Y_k - Y_j}{\|Y_k - Y_j\| + \varepsilon} \tag{9}$$

$$\hat{L}_{j,k} = \frac{L_j - L_k}{L^{worst} - L^{best}} \tag{10}$$

Where, the best and the worst fitness values of the krill individuals is L^{best} and L^{worst}; fitness or the objective function value of the jth krill individual is L_j; the fitness of kth ($k = 1,2,\ldots,NN$) neighbor; the related position is Y; and the neighbor's number is NN. ε is the small positive number added to avoid the singularities. In dynamic cloud environments, the current state of the system is often influenced by previous states. This is where fractional calculus becomes important. By introducing fractional derivatives, FSKHA model memory, and hereditary properties, the algorithm is more responsive to evolving cloud structures. The Caputo fractional derivative utilized in the FSKHA introduces a fractional component to the optimization process, allowing for more precise control of krill movement. This helps in detecting slowly evolving anomalies. The Caputo fractional derivative of a function $g(t)$ is defined as in equation (11):

$$D^{\alpha} g(t) = \frac{1}{\Gamma(n-\alpha)} \int_0^t \frac{g^n(t-\tau)}{\tau^{1-\alpha}} d\tau \tag{11}$$

Where $\alpha = 1$ provided the optimal balance between capturing short-term fluctuations and long-term trends in the cloud data. By using fractional calculus, FSKHA captures both short-term and long-term dependencies in cloud behavior, improving its ability to detect anomalies that evolve slowly over time. This fractional component helps the algorithm achieve more precise control over krill movements, improving both the exploration and exploitation phases of the optimization process.

Foraging motion: Two primary effective parameters are used to formulate the foraging motion. The first is the location of the food, and the second is prior knowledge of the location of the food. For the j^{th} krill, this motion can be described as follows in Eq. (12)

$$D^{\alpha} H_j = F_s \gamma_j + \delta_s H_j^{Old} \tag{12}$$

Where γ_j is given in Eq. (13)

$$\gamma_j = \delta_j^{Food} + \delta_j^{Best} \tag{13}$$

Where the foraging speed is F_s, the inertia weight of the foraging motion between range [0,1] is δ_s, the last foraging motion is H_j^{Old}, the food attractive is δ_j^{Food} and the effect of the best fitness of the jth krill is δ_j^{Best}. This equation, now involving the fractional derivative $D^{\alpha} H_j$, accounts for historical data in the search process, improving the accuracy of the search for optimal solutions.

Physical diffusion: The distribution of the krill individuals physically is thought to be a random phenomenon. A maximum diffusion speed and a random directed vector can be

DOI: 10.6688/JISE.20220X_38(X).00XX

used to describe this motion and it is stated in Eq. (14).

$$R_j = R^{Max}\alpha \tag{14}$$

Where the maximum diffusion speed is R^{Max} and the arbitrary directional vector and its arrays are the arbitrary values between -1 to 1. A diffusion speed of 0.3 is found to balance exploration and convergence time, enabling the algorithm to efficiently detect anomalies while avoiding local optima. With more time (iterations), the effects of the motion caused by other krill individuals and foraging motion gradually diminish. By Eq. (14), a random vector of the physical diffusion does not decrease continuously as the iteration number rises. As a result, Eq. (14) includes a new term and is given in Eq. (15), which is based on a geometrical annealing schedule and the random speed is linearly reduced with time, where J is the identity matrix.

$$R_j = R^{Max}\left(\frac{J}{J_{Max}}\right)\alpha \tag{15}$$

Genetic operator: Genetic reproduction processes are added to the algorithm to enhance performance. Crossover and mutation, are two new adaptive genetic reproduction methods that draw inspiration from the traditional DE algorithm. GA introduces the crossover operator as a powerful technique for overall optimization. The crossover is also employed in DE, which can be seen as an advancement above GA, in a vectorized form and the crossover is controlled by a crossover probability. In evolutionary algorithms like ES and DE, the mutation is crucial and a mutation probability governs the mutation. The Algorithm for FSKHA is explained in Algorithm 1.

Algorithm 1: FSKHA

Initialize the population of krill with random positions
Evaluate the fitness of each krill in the population
Motion calculation
    Motion tempted by other krill individuals using stochastic differential equation
    Foraging motion
    Physical diffusion
Genetic operator implementation
In the search space, update the individual position of krill
Repeat from step 2 up to the end of stopping criteria
End

The overall concept of this anomaly detection in the dynamic cloud environment is by applying the change detection algorithm to evaluate the changes in cloud structure over time by analysing the various parameters, which in turn provides the dynamic changes as normal behaviors. Further, the anomalies over the dynamic cloud structure are identified by using a novel anomaly detection algorithm, which detects the patterns that do not match with the normal behavior as an anomaly. The analysis of this work is explained in the next section.

4. RESULT AND DISCUSSION

DOI: 10.6688/JISE.20220X_38(X).00XX

This section includes a thorough analysis of the performance of the proposed anomaly detection with proposed Density Estimated LSTM and Fractional Stochastic Krill Herd. The implementation results were simulated in the Python platform, and a comparison section to make sure the proposed framework successfully determines the anomalies in the changing cloud structure over time.

4.1. Experimental Setup

This work has been implemented in the working platform of Python with the following system specification and the simulation results are discussed below.

Platform	**:** Python
OS	**:** Windows 10
Processor	**:** 64-bit Intel processor
RAM	**:** 8 GB RAM

4.2 Dataset description

In this research Network Traffic Dataset is used in cloud environments to capturing both normal and anomalous behaviors. The data shown here was gathered on a Kali Machine from the University of Cincinnati in Cincinnati, Ohio, by using Wireshark to record packets for one hour in the evening on October 9th, 2023.This dataset contains 394137 occurrences that were collected and recorded in a CSV (Comma Separated Values) file. This network traffic dataset consists of 7 features. Each one provides information on the source and destination IP addresses. The bulk of the attributes are integer in nature, but there are also nominal and date types due to the timestamp. This big dataset is utilized for a variety of ML applications, such as network traffic categorization, network performance monitoring, network security management, network traffic management, intrusion detection, and anomaly detection.

This dataset is available at: https://www.kaggle.com/datasets/ravikumargattu/network-traffic-dataset

4.3. Performance metrics of proposed Density Estimated LSTM and Fractional Stochastic Krill Herd

The performance of the proposed Density Estimated LSTM and Fractional Stochastic Krill Herd for anomaly detection in the dynamic cloud environment is evaluated in detail in this section.

DOI: 10.6688/JISE.20220X_38(X).00XX

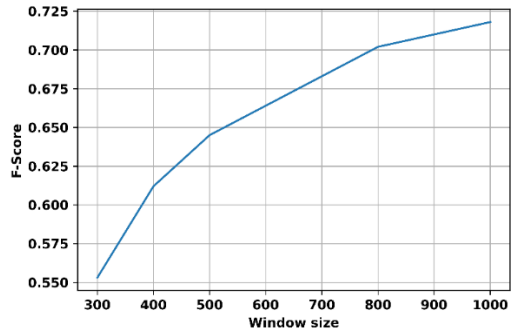

Fig. 3. Performance analysis of F-score of Implemented technique

Fig. 3. depicts the F-score performance of the implemented technique regarding the window size. The sliding time series' window size ti significantly affects the F-score. The findings show that the proposed FSKHA model's ability to capture the pattern of the input data is reduced when the window size value is short. From Figure 3, the proposed model conceived the lowest F-score of 0.55, when the window size is at 300, and achieves a maximum of 0.72 when the window size is 1000. Larger window sizes provide more data points, allowing the model to capture long-term dependencies and patterns in the time series data. This improved context leads to better identification of normal behavior and more accurate anomaly detection.

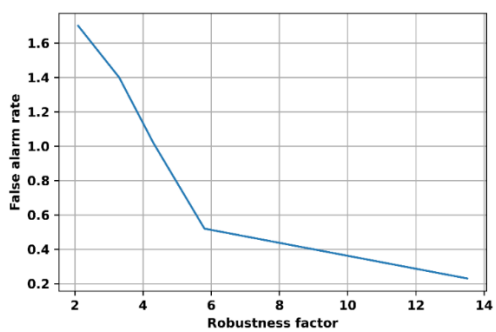

Fig. 4. Performance analysis of False Alarm Rate of Implemented technique

Fig. 4. depicts the false alarm rate performance of the proposed technique regarding the robustness factor. The optimum value of the robustness factor is 0.57 to gain a lower false alarm rate. From the graph, when the robustness factor is 2, the false alarm rate is highest at 1.6, and when the robustness factor is 14, the false alarm rate achieves its lowest at 0.2. The robustness factor is a term for a parameter that helps lower false positives by adjusting the anomaly detection algorithm's sensitivity to changes in data. The algorithm's capacity to distinguish between normal and abnormal data points is enhanced by a larger robustness factor, which reduces false alarms. By adding the robustness factor, the detection system's total flexibility is increased, providing its stability even in busy and dynamic cloud settings. The proposed FSKHA achieves better discrimination between normal and anomalous data points, leading to a reduction in false alarms. False alarm rate drops along with the increase in robustness factor.

DOI: 10.6688/JISE.20220X_38(X).00XX

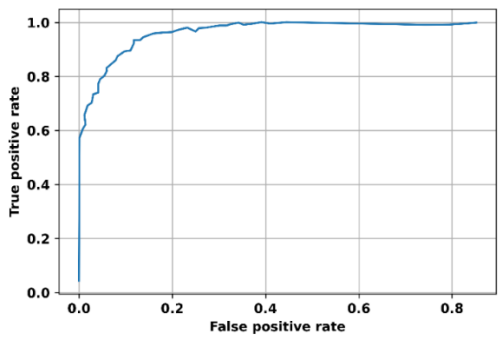

Fig. 5. Performance analysis of Area Under Curve of Implemented technique

Fig. 5. analyses the area under curve metrics of the proposed technique regarding true positive rate and false positive rate. Area under the curve (AUC) is a performance metric that measures performance across all classification thresholds, which is a measure of how well a classifier performs when there is more area under the curve. In this, the proposed Density Estimation LSTM effectively chooses the threshold, thereby attaining a perfect accuracy when AUC equals 1. The graph thus shows that the proposed algorithm achieves the highest AUC in terms of true and false positive rates.

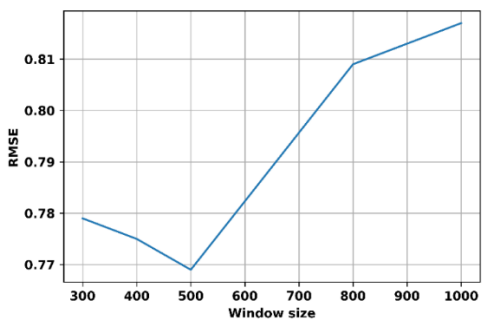

Fig. 6. Performance analysis of RMSE of Implemented technique

Fig. 6. depicts the performance regarding the RMSE with the window size for the implemented technique. Here the analysis is done by varying the number of Density estimation LSTM units for calculating the RMSE. This analysis is done to optimally predict the sliding window of Density estimation LSTM. To get the RMSE findings, the window size is altered between 300 and 1000. The line of RMSE obtained the lowest value at window size 500, which is the optimal value of sliding window of Density estimation LSTM.

DOI: 10.6688/JISE.20220X_38(X).00XX

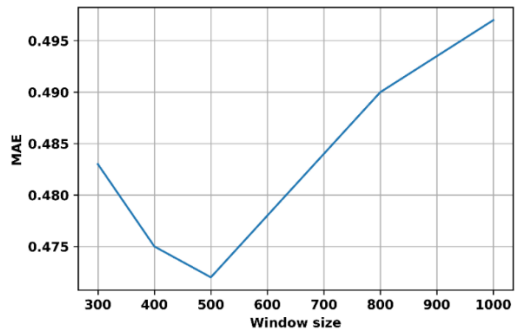

Fig. 7. Performance analysis of MAE of Implemented technique

Fig. 7. depicts the performance regarding the MAE with the window size for the implemented technique. Here the analysis is done by varying the number of Density estimation LSTM units for calculating the MAE. This analysis is done to optimally predict the sliding window of Density estimation LSTM. To get the MAE findings, the window size is altered between 300 and 1000. The line of MAE obtained the lowest value at window size 500, which is the optimal value of sliding window of Density estimation LSTM.

4.4. Comparative Analysis of Proposed Density Estimated LSTM and Fractional Stochastic Krill Herd

This section highlights the proposed Density Estimated LSTM and Fractional Stochastic Krill Herd with the traditional models and the achieved outcome was explained in detail in this section by comparing it with Fuzzy C Means (FCM), Support Vector Machine (SVM), Machine Learning Intrusion Detection System (ML-IDS), Multi-Step outlier-based Anomaly Detection Approach (MS-ADA) and Ensemble Artificial Bee Colony (En-ABC) [19], and showing their results based on various metrics.

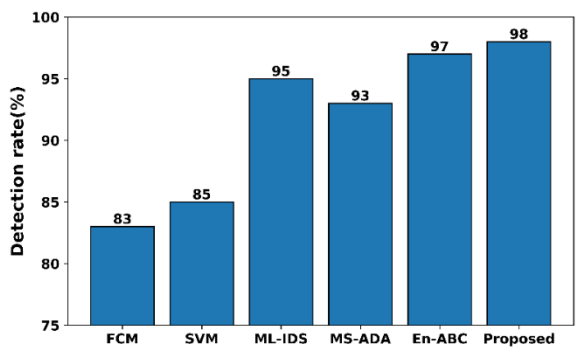

Fig. 8. Comparative analysis of the Detection Rate of Implemented technique

Fig. 8. depicts the comparison of detection rate metrics for the proposed model over the traditional models. In this, the existing techniques FCM, SVM, ML-IDS, MS-ADA, and En-ABC attains the detection rate of 83%, 85%, 95%, 93%, and 97%, respectively, whereas the proposed techniques attain the maximum detection rate of 98%, which shows

DOI: 10.6688/JISE.20220X_38(X).00XX

the effective detection rate of an anomaly in dynamic cloud environment.

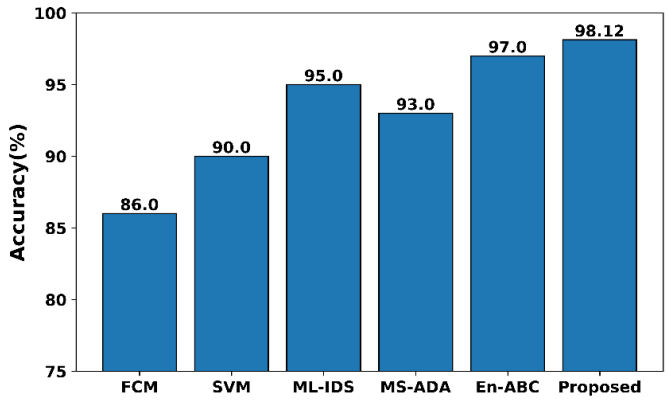

Fig. 9. Comparative analysis of Accuracy of Implemented technique

Fig. 9. explains the comparison of accuracy metrics for the proposed model over the traditional models like FCM, SVM, ML-IDS, MS-ADA, and En-ABC. In this, the existing techniques FCM, SVM, ML-IDS, MS-ADA, and En-ABC attain the detection rate of 86%, 90%, 95%, 93%, and 97%, respectively, whereas the proposed model attains the maximum accuracy of 98.12%, which shows the effective accuracy on the overall system performance.

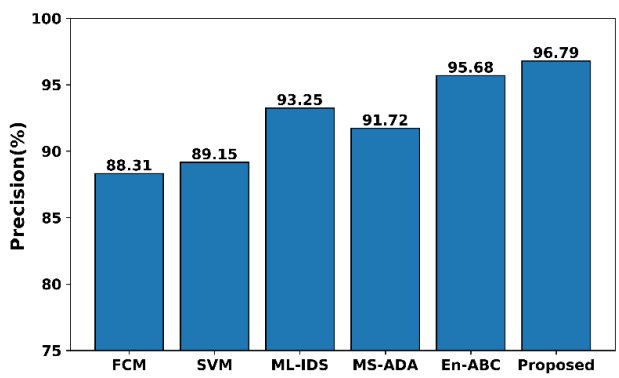

Fig. 10. Comparative analysis of Precision of Implemented technique

Fig. 10. illustrates the comparison of precision metrics for the proposed model over the traditional models like FCM, SVM, ML-IDS, MS-ADA, and En-ABC. In this, the existing techniques FCM, SVM, ML-IDS, MS-ADA, and En-ABC attain the detection rate of 88.31%, 89.15%, 93.25%, 91.72%, and 95.68%, respectively, whereas the proposed technique gains the maximum precision of 96.79%, which shows the effective precision with better performance than existing techniques.

DOI: 10.6688/JISE.20220X_38(X).00XX

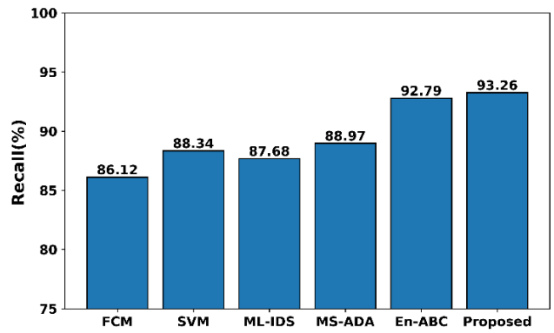

Fig. 11. Comparative analysis of Recall of Implemented technique

Fig. 11. evaluates the comparison of recall metrics for the proposed model over the traditional models like FCM, SVM, ML-IDS, MS-ADA, and En-ABC. In this, the existing techniques FCM, SVM, ML-IDS, MS-ADA, and En-ABC attain the detection rate of 86.12%, 88.34%, 87.68%, 88.97%, and 92.79%, respectively, whereas the proposed model achieves the maximum recall of 93.26%, which shows the effectiveness of the proposed technique in terms of recall over conventional concepts.

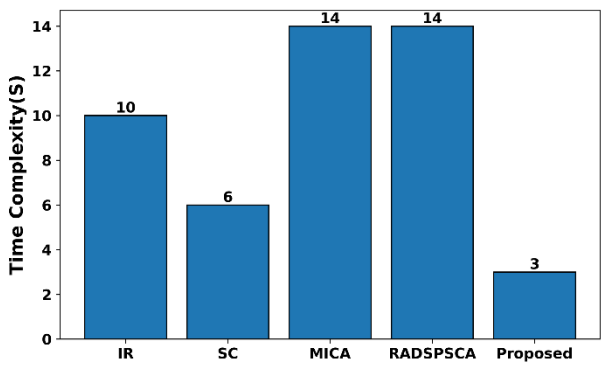

Fig. 12. Comparative analysis of time complexity of Implemented technique

Fig. 12. demonstrates the time complexity comparison of the suggested model with the existing models. The existing models such as IR, SC, MICA, and RADSPSCA attain a time complexity value of 10s, 6s, 14s, and 14s respectively. In comparison, the proposed framework has a significantly reduced time complexity of only 3 seconds. This indicates the proposed approach's efficiency in processing and identifying anomalies in cloud settings, demonstrating its potential to outperform established approaches in terms of. speed and adaptability.

DOI: 10.6688/JISE.20220X_38(X).00XX

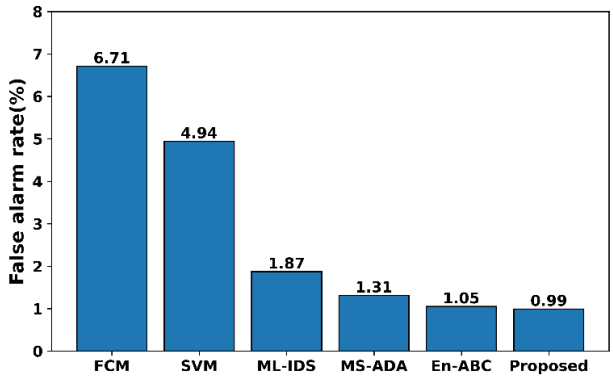

Fig. 13. Comparative analysis of False Alarm Rate of Implemented technique

Fig. 13. depicts the comparison of false alarm rate metrics for the proposed model over the traditional models like FCM, SVM, ML-IDS, MS-ADA, and En-ABC. In this, the existing techniques FCM, SVM, ML-IDS, MS-ADA, and En-ABC attain the false alarm rate of 6.71%, 4.94%, 1.87%, 1.31%, and 1.05%, respectively, whereas the proposed technique obtains the minimum false alarm rate of 0.99%, which shows minimized false alarm rate with better prediction of anomalies over other techniques.

4.5 Ablation study

The focus of the ablation study is to assess the performance of various parts and combinations of this anomaly detection system, with a specific focus on the FSKHA, the Density Estimated LSTM, and the proposed combined model that combines both techniques. The metrics assessed include Accuracy, Precision, Recall, and Detection Rate, which are critical for evaluating the effectiveness of anomaly detection methods.

Table 1: Ablation study

Model Configuration	Accuracy (%)	Precision (%)	Recall (%)	Detection rate (%)
Density Estimated LSTM	95.27	91.39	90.37	96.38
FSKHA	97.82	93.61	91.95	97.1
Proposed (Density Estimated LSTM + FSKHA)	98.12	96.79	93.26	98

The ablation study evaluates the performance of three anomaly detection models, which are shown in Table 1. The Density Estimated LSTM achieves an accuracy of 95.27%, precision of 91.39%, recall of 90.37%, and a detection rate of 96.38%. FSKHA improves these metrics with an accuracy of 97.82%, precision of 93.61%, recall of 91.95%, and a detection rate of 97.1%. The proposed model, integrating Density Estimated LSTM with FSKHA, further enhances performance to an accuracy of 98.12%, precision of 96.79%, recall of 93.26%, and a detection rate of 98%. By combining the best features of both approaches, this strategy captures long-term interdependence and responds to new

threats more accurately and effectively. The results demonstrate that integrating these techniques provides superior anomaly detection compared to using either method alone.

Overall, the proposed Density Estimated LSTM and FSKHA outperform the traditional methods FCM, SVM, ML-IDS, MS-ADA, and En-ABC with a better detection rate of 98% and accuracy with a maximum value of 98.12% and a false alarm rate of minimized value 0.99%. Therefore, the density estimated LSTM thus effectively captures the dynamic changes of cloud structures with respect to the relevant parameters such as resources, memory, and network traffic. In contrast, FSKHA effectively identifies the normal behavior from the anomalies under the dynamic condition that are captured early by density estimated LSTM.

5. CONCLUSION

The novel Density Estimated LSTM and Stochastic Krill Herd algorithm was introduced to capture the changing behavior of cloud structure over time and to detect the anomalies in the dynamic cloud environment. By examining the probability distribution from the output of LSTM, Density Estimation LSTM is designed to assess the density distribution of cloud forms. Consequently, the alterations and differences in cloud patterns were captured. FSKHA takes into account the behavior of krill swarms and contains stochastic aspects, was also applied to detect anomalies from developing threats, which captures both known and unknown anomalies that emerge with the new trends and data. The proposed model thus attains a higher detection rate of 98% with a maximum accuracy of 98.12%, and a low false alarm rate of 0.99%, with maximum recall and precision of 93.26% and 96.79%. Thus, the proposed model was utilized to provide a better performance and precisely detect the known and unknown anomalies from the emerging threats in dynamic cloud structures. The overall performance analysis shows the outperformance of this proposed Density Estimated LSTM and Stochastic Krill Herd approach.

REFERENCES

[1] M. Muhic, L. Bengtsson, and J. Holmström, "Barriers to continuance use of cloud computing: Evidence from two case studies," *Information & Management*, Vol. 60, No. 5, 2023, pp. 103792.

[2] P. Tamilarasu, and G. Singaravel, "Quality of Service Aware Improved Coati Optimization Algorithm for Efficient Task Scheduling in Cloud Computing Environment," *Journal of Engineering Research*, 2023.

[3] F.A. Silva, I. Fe, C. Brito, G. Araujo, L. Feitosa, T.A. Nguyen, K. Jeon, J.W. Lee, D. Min, and E. Choi, "Aerial computing: Enhancing mobile cloud computing with unmanned aerial vehicles as data bridges–A Markov chain based dependability quantification," *ICT Express*, 2023.

[4] L. Ismail, H. Materwala, and H.S. Hassanein, "QoS-SLA-Aware Adaptive Genetic Algorithm for Multi-Request Offloading in Integrated Edge-Cloud Computing in Internet of Vehicles," *arXiv preprint arXiv:2202.01696,* 2022.

[5] K.S. Saraswathy and S.S. Sujatha, "Secure data storage and access for fish monitoring in cloud environment," *Measurement: Sensors*, Vol. 27, 2023, pp. 100693.

[6] B.H. Banimfreg, "A comprehensive review and conceptual framework for cloud computing adoption in bioinformatics," *Healthcare Analytics*, 2023, pp. 100190.
[7] Y. Zhang, F. Zhang, S. Tong and A. Rezaeipanah, "A dynamic planning model for deploying service functions chain in fog-cloud computing," *Journal of King Saud University-Computer and Information Sciences,* Vol. 34, No. 10, 2022, pp.7948-7960.
[8] X. Li, C. Su, M. Ghobaei-Arani, and M.F. Albaghdadi, "Dynamic service function chain placement with instance reuse in Fog–Cloud Computing," *ICT Express*, 2022.
[9] N. Menaka and J. Samraj, "A hybrid convolutional neural network with long short-term memory (HCNN-LSTM) model based Edge System Recommendation (ESR) for cloud service providers," *Measurement: Sensors*, Vol. 29, 2023, pp.100886.
[10] J. Ruuskanen, T. Berner, K.E. Arzen, and A. Cervin, "Improving the mean-field fluid model of processor sharing queueing networks for dynamic performance models in cloud computing," *ACM SIGMETRICS Performance Evaluation Review*, Vol. 49, No. 3, 2022, pp. 69-70.
[11] N. Mc Donnell, E. Howley, and J. Duggan, "Dynamic virtual machine consolidation using a multi-agent system to optimise energy efficiency in cloud computing," *Future Generation Computer Systems*, Vol. 108, 2020, pp.288-301.
[12] Z. Wang, Y. Zhang, T. Lv and L. Luo, "GTAINet: Graph neural network-based two-stage anomaly identification for locking wire point clouds using hierarchical attentive edge convolution," *International Journal of Applied Earth Observation and Geoinformation,* Vol. 115, 2022, pp. 103106.
[13] Y. Yang, S. Ding, Y. Liu, S. Meng, X. Chi, R. Ma and C. Yan, Fast wireless sensor for anomaly detection based on data stream in an edge-computing-enabled smart greenhouse, *Digital Communications and Networks*, Vol. 8, No. 4, 2022, pp. 498-507.
[14] F.J. Abdullayeva, "Distributed denial of service attack detection in E-government cloud via data clustering," *Array*, Vol. 15, 2022, pp. 100229.
[15] X. Yu, X.Yang, Q. Tan, C. Shan, and Z. Lv, "An edge computing based anomaly detection method in IoT industrial sustainability," *Applied Soft Computing*, Vol. 128, 2022, pp. 109486.
[16] K.N. Qureshi, G. Jeon, and F. Piccialli, "Anomaly detection and trust authority in artificial intelligence and cloud computing," *Computer Networks*, Vol. 184, 2021, pp. 107647.
[17] A.M. El-Shamy, N.A. El-Fishawy, G. Attiya, and M.A. Mohamed, "Anomaly detection and bottleneck identification of the distributed application in cloud data center using software–defined networking," *Egyptian informatics journal*, Vol. 22, No. 4, 2021, pp.417-432.
[18] J. Xu, C. Lin, F. Liu, Y. Wang, W. Xiong, Z. Li, H. Guan and G. Xie, "StreamAD: A cloud platform metrics-oriented benchmark for unsupervised online anomaly detection," *BenchCouncil Transactions on Benchmarks, Standards and Evaluations*, Vol. 3, No. 2, 2023, pp. 100121.
[19] S. Garg, K. Kaur, S. Batra, G.S. Aujla, G. Morgan, N. Kumar, A.Y. Zomaya, and R. Ranjan, "En-ABC: An ensemble artificial bee colony based anomaly detection scheme for cloud environment," *Journal of Parallel and Distributed Computing*, Vol. 135, 2020, pp.219-233.

[20] R. Wang, H. Qiu, X. Cheng, and X. Liu, "Anomaly detection with a container-based stream processing framework for Industrial Internet of Things," *Journal of Industrial Information Integration*, Vol. 35, 2023, pp. 100507.

[21] C. Yang, "Anomaly network traffic detection algorithm based on information entropy measurement under the cloud computing environment," *Cluster Computing*, Vol. 22, 2019, pp. 8309-8317.

[22] S. Mahdavi Hezavehi, and R. Rahmani, "An anomaly-based framework for mitigating effects of DDoS attacks using a third party auditor in cloud computing environments," *Cluster Computing*, Vol. 23, No. 4, 2020, pp. 2609-2627.

[23] P. Savaridassan, and G. Maragatham, "Integrated deep auto-encoder and Q-learning-based scheme to detect anomalies and supporting forensics in cloud computing environments," *Wireless Personal Communications,* 2021, pp.1-19.

[24] J. Zhang, "Anomaly detecting and ranking of the cloud computing platform by multi-view learning," *Multimedia Tools and Applications*, Vol. 78, No. 21, 2019, pp.30923-30942.

[25] P. Nawrocki, and W. Sus, "Anomaly detection in the context of long-term cloud resource usage planning," *Knowledge and Information Systems*, Vol. 64, No. 10, 2022, pp. 2689-2711.

[26] A.S. Rahumath, M. Natarajan, and A.R. Malangai, "Resource Scalability and Security Using Entropy Based Adaptive Krill Herd Optimization for Auto Scaling in Cloud", *Wireless Personal Communications*, Vol. 119, pp. 791-813, 2021.

[27] S. Sivamohan, S.S. Sridhar, and S. Krishnaveni, "TEA-EKHO-IDS: An intrusion detection system for industrial CPS with trustworthy explainable AI and enhanced krill herd optimization", *Peer-to-Peer Networking and Applications*, Vol. 16, No. 4, pp. 1993-2021, 2023.