International Journal of Data Science and Artificial Intelligence (IJDSAI)
Volume 3, Issue 1, January – February (2025)

RESEARCH ARTICLE

AT-DKNN: DETECTING ATTACK TRAFFIC USING DEEP LEARNING TO IMPROVE CLOUD SECURITY

Joshua Bapu J [1, *], Ashis Kumar Mishra [2]

[1] Associate Professor, Department of Electronics and Communication Engineering, Vel Tech Multi Tech Dr. Rangarajan Dr. Sakunthala Engineering College, Avadi.
[2] Assistant Professor, Department of Computer Science and Engineering (Presently Known as School of Computer Science), Odisha University of Technology and Research (OUTR), Bhubaneswar, Odisha, India.
[*]*Corresponding e-mail: joshuababu@gmail.com*

Abstract – Attack traffic is the presence of malicious network traffic within a dataset, suggesting that at least one attack occurred. This type of traffic is typically generated by cybercriminals, hackers, or automated bots and can take many forms, including DDoS attacks, malware infections, and phishing efforts. To overcome this, a novel Attack Traffic detection (AT-DKNN) is proposed in this paper for effectively detecting and categorizing the types of attacks to improve cloud security. The process begins with data from the IoT device. Preprocessing of the collected dataset involves data normalization to standardize the inputs. The Spatio-Temporal Graph Neural Network (STGNN) is used for feature extraction, exploiting both spatial and temporal connections to create detailed feature representations. The feature extraction was followed by feature selection using Red Kite optimization to determine the most relevant attributes and reduce dimensionality. The optimized features are fed into a Deep Kronecker Neural Network, which classifies network data as normal or attack traffic. Attack traffic is recognized for further action, whereas normal traffic is safely routed to the cloud environment. The AT-DKSNN method achieves 98.33% accuracy, while the DNN, LSTM, and CNN approaches reach low accuracy of 96.26%, 92.32%, and 94.25%, respectively.

Keywords *– Deep learning, red kite optimization, Spatio-Temporal Graph Neural Network, Deep Kronecker neural network, Attack detection.*

1. INTRODUCTION

The cloud has revolutionized data innovation by providing unparalleled flexibility, adaptability, and efficiency in information management and preparation. As enterprises migrate their operations and information to the cloud, cybersecurity challenges are also expanding [1]. IDPS architectures are essential for securing cloud environments against many threats, including traditional cyber-attacks and new targeted interruptions. This introduction explores the role of IDPS in cloud security, including its components, problems, and advancements within the cloud ecosystem [2]. Cloud computing technology becoming increasingly important for efficiently and securely managing large amounts of data generated through communication networks [3]. The main concern is the security of national data. As the globe becomes more globalized, cloud computing infrastructures are increasingly being used commercially across borders. Security mechanisms play a significant role in determining the quality of cloud computing services [4]. Even while cloud computing is seen as a significant and beneficial change to IT infrastructure, there is still a lot of security work to be done to address its shortcomings. Cloud security threats and vulnerabilities need to be identified and fixed because cloud data centers house a significant amount of personal and business information. Cloud infrastructure is vulnerable to attacks since it uses virtualization methods and common Internet protocols [6].

The DL system is particularly effective in distinguishing DDoS attacks from benign traffic by extracting representations of high-level features from low-level communication. DL and computers provide efficient deployment of security technologies such as access control, cloud encryption, malware identification, and secure uploading [7]. Implementing deep learning in cloud security has significant benefits, including automated threat detection. DL algorithms automatically identify anomalies and security issues by analyzing vast volumes of data, such as system logs, network traffic logs, and user behavior [8]. Deep learning-powered automated systems improve security effectiveness by ensuring consistent and accurate task performance. Deep learning algorithms identify patterns and relationships in data, leading to more effective threat identification and classification [9]. DL models are the most effective optimization technique due to their ability to adapt changing threats and learn from fresh data in real-time. Efficient and cost-effective security systems minimize response times, eliminate false positives, and need less manual setting. DL analyzes past data and patterns to predict future threats [10]. To overcome these challenges the proposed AT-DKNN framework contribution are as follows:

- Initially, the process starts with collecting data from IoT devices and pre-processed data,

including data cleaning, normalization, to remove irrelevant data.

- The AT-DKNN method utilizes a STGN to extract the relevant features.
- Red Kite Optimization to minimize dimensionality and find the most pertinent features.
- Finally, AT-DKNN approach utilizes a Deep Kronecker Neural Network to classify intrusions like Normal, Attack Traffic. The suggested technique efficiency is assessed utilizing accuracy, precision, recall, f1score, and MSE.

From the above literature survey, several shortcomings are faced by existing methods such as potentially impacting system performance, security, scalability, and efficiency. This research introduces a novel method called AT-DKNN**,** which will be covered in more depth in the next part to overcome these problems.

2. LITERATURE SURVEY

In 2021, Selvapandian, D. and Santhosh, R., [11] developed an integrating cloud with IoT networks is gaining popularity because it minimizes sensor node resource limits. IDS are frequently used to assure network security and operational reliability. IDS reliably finds anomalies in complicated network environments while also ensuring network security. Deep learning approaches have recently become widely used in a variety of image and signal processing, as well as security applications.

In 2022, Saadi, C. and Chaoui, H., [12] proposed a new architecture for cloud infrastructure that integrates mobile agent-based intrusion detection systems (IDS) with three types of honeypots to identify attacks, examine the behavior of attackers, increase the value of honeypots and IDS-based mobile agents, resolve the limitations of intrusion detection systems, improve IDS knowledge bases, and eventually raise the detection rate in cloud environments. However, the IDS system requires an efficient security architecture in order to enhance cloud security.

In 2022, Akgun, D., Hizal, S. and Cavusoglu, U., [13] proposed an intrusion detection system based on preprocessing methods and a deep learning model. Various models based on DNN, CNN) and LSTM were examined for detection and real-time performance. Using the CIC-DDoS2019 dataset, a popular dataset in the area, the suggested model was developed. The applied preprocessing methods of CIC-DDoS2019 dataset, including feature deletion, random subset selection, feature selection, duplication removal, and normalizing. Improved recognition performance was demonstrated by the assessments of training and testing.

In 2023, Srilatha, D. and Thillaiarasu, N., [14] proposed different cyber-attacks and network irregularities that contribute to the development of an effective intrusion detection system play an important part in modern security. The NSL-KDD benchmark dataset is frequently used in academia despite the fact that it was developed nearly a decade ago and does not accurately reflect network traffic or low-footprint assaults nowadays. The Canadian Institute of Cyber Security published the CICIDS2017 network dataset to tackle the NSL-KDD issue. There is no doubt that the successful IDPS is developed and evaluated in a network environment using a variety of machine learning techniques. To create model that can replicate IDS-IPS system by identifying either a stream of network data is malicious or benign.

In 2023, Salvakkam, D.B., et al., [15] developed to look for possible dangers in the cloud, and the emergence of attacks using quantum computing necessitates the adoption of an intrusion detection system (IDS) to evaluate the risks to cloud security. In order to address this issues, this paper offers a novel method for identifying cloud computing breaches utilizing the KDDcup 1999, UNSW-NB15, and NSL-KDD datasets. Two goals are supposed to be achieved by this proposed system. It proposes an accuracy improvement model of IDS after first assessing the weaknesses of the current IDS.

In 2023, Kavitha, C., [16] developed a filter-based ensemble feature selection (FEFS) and used a deep learning model (DLM). The suggested methodology was validated using the data. The features that could aid in intrusion prediction were selected from the obtained database. Filtering, wrapping, and embedding algorithms are the three feature extraction procedures that make up the FEFS. The key features were selected to facilitate the DLM's training process based on feature extraction process described above. Ultimately, chosen features pass into the classifier. The DLM integrates TDO with an RNN.

In 2024, Ali, S.Y., et al., [17] suggested system exhibits its capacity to react to shifting threats by updating its knowledge on a frequent basis with new data. The CNN-based IDS is evaluated using detailed experiments that compare its performance to older approaches. The findings demonstrate that the CNN-based strategy performs better than conventional IDS methods, underscoring its potential dependable and effective IDS for cloud computing settings. DL architecture was created to handle particular security concerns associated with cloud computing. In contrast to conventional intrusion detection systems that depend on rule- or signature-based methods, this study describes a CNN-based intrusion detection system that automatically learns hierarchical characteristics from raw data while utilizing the network's capacity

3. PROPOSED COVNET METHODOLOGY

In this study a novel AT-DKNN method approach is to detect intrusions based deep learning to improve cloud security in IOT device. This process begins with data collection which is gather from the iot device. The collected dataset goes under preprocessing, which includes data normalization to normalized the inputs. Spatio-Temporal Graph Neural Network (STGNN) is used for feature extraction which exploiting both spatial and temporal connection to generate rich feature representations. Then the feature extraction passed through feature selection by using Red Kite optimization to identify the most revelent features

and reduce dimensionality. Optimized features fed into Deep Kronecker Neural Network for classification and network traffic is classified as normal traffic, attack traffic. Attack traffic is identified for further action, while normal traffic is safely sent to the cloud environment. The suggested technique shows a hacker injecting harmful traffic into the system, which the framework is intended to detect and mitigate. Figure 1 shows the overall workflow of the suggested AT-DKNN methodology.

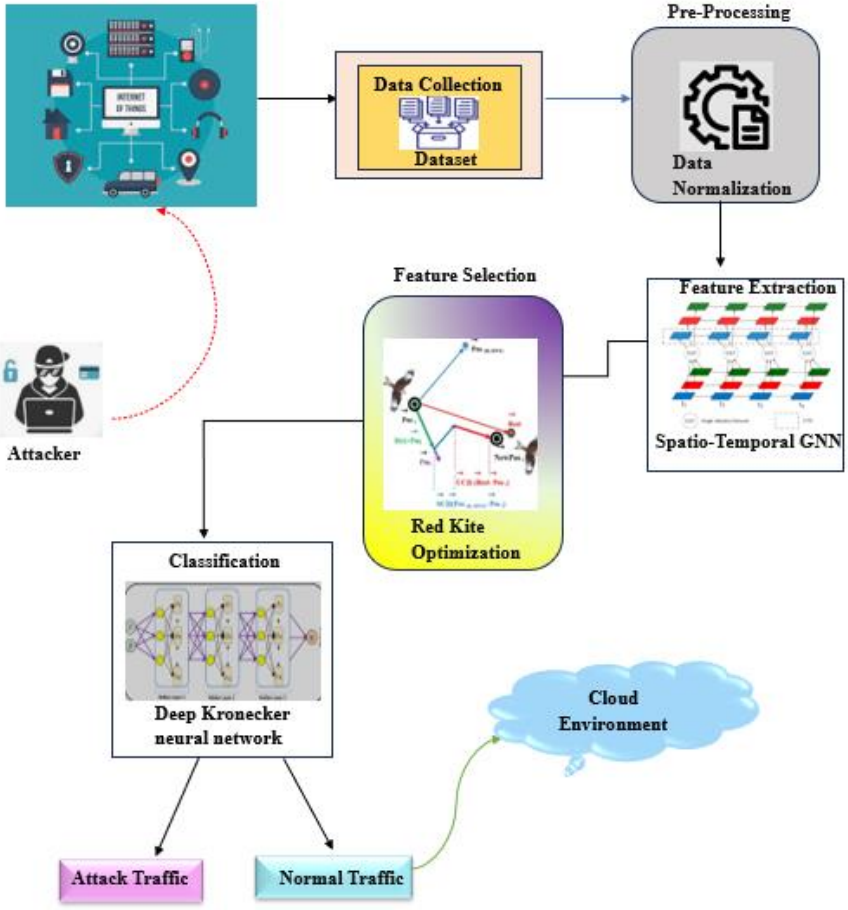


Figure 1. AT-DKNN Proposed method

3.1 Data Collection

Data collection is gathering data from IoT devices, sensors, and network logs in order to monitor activity and detect potential dangers. This data could comprise network traffic, device behavior, system logs, or environmental characteristics that are critical for detecting abnormalities and security issues.

3.2 Data Preprocessing

Preprocessing transforms incoming data into usable shape by removing unnecessary information. The model's accuracy and efficiency can only be increased by modifying and preparing data for the learning process.

Z-Score Normalization

The value of a z-score represents standard deviation from mean. Where the zero represents the mean, when the z-score is positive. Compared to the mean average, the raw score is higher. Normalization is a preprocessing technique that breaks down numerical features and turns them into a specific range.

Normalizing data can be done using a variety of approaches, including decimal scaling, z-score, and min-max. Equation 1 shows how Z-score normalization transfers a previously unknown set of vi values from attribute E.

$$w'(j) = \frac{w^{(j)} - \mu^{(j)}}{\partial^{(j)}} \quad (1)$$

w' represents the normalized value. v denotes the value to be normalized in the attribute. μ(j) refers to the attribute's mean value, while $\partial(j)$ represents its standard deviation.

3.3. Feature extraction using Spatio-Temporal GNN

Spatio-Temporal Graph Neural Networks

Graphs are widely used to depict data in several domains, including social, biological, and financial. GNN-based deep learning approaches have demonstrated higher performance in semi-supervised node classification, community detection, graph classification, and recommendations. Spatio-temporal graphs, which model relationships between nodes based on time and space, are widely used in several domains. GNNs have been successfully used to traffic graphs and influenza predictions,

which is pertinent regarding our work. In the latter two situations, temporal dependencies were integrated at the model level, using a dynamic Laplacian matrix or a recurrent neural network.

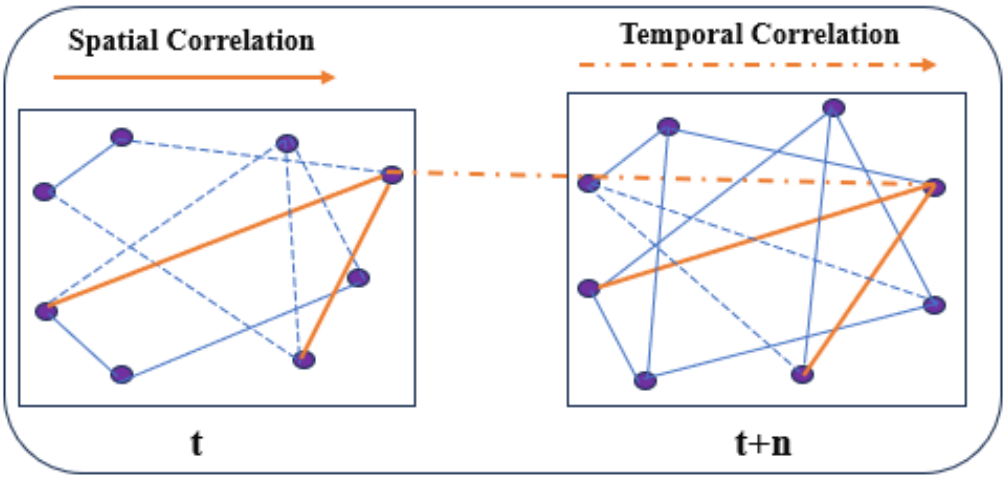


Figure 2. ST-GNN Models

3.4. Feature selection using Red Kite Optimization

The RKO is a type of metaheuristic inspired the social lives of red kites. Usually, red kites build their nests close to lakes and wooded areas that are good for hunting. They hunt at high speeds, reside together, and move erratically when flying. Their voices are referred to as the "sound of unity, which were created during moments such as finding good bait, locating a water supply, migrating, and giving birth. The sound of danger also refers to the sounds made at times of peril, like an enemy attack, the death of another animal, an earthquake, or a storm.

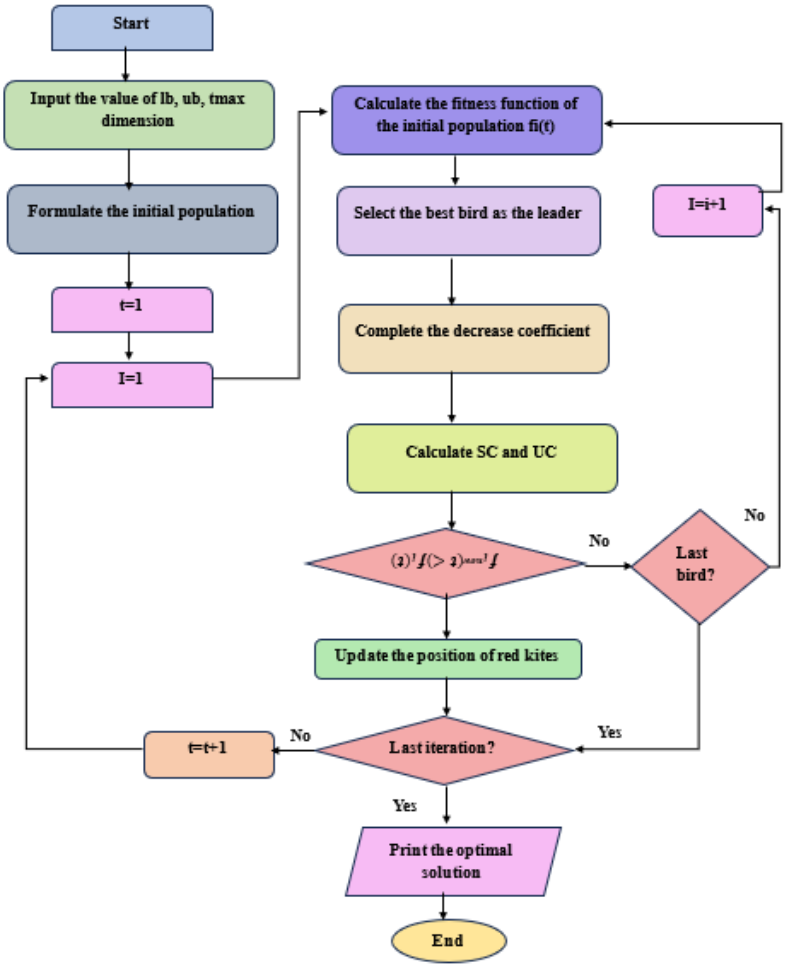


Figure 3. Red Kite Optimization Flow chart

It consists of three main steps, which are outlined below:

$Posi,\ j(t)=lb+rand\times(ub-lb),\ i=1,\ 2,\ \ldots,\ n\ and\ jnd\ j=1,\ 2\ldots,\ d$ (1)

where $Posi, j(t)$ is ith red kite's location at iteration t, lb and ub are lower and upper limits, respectively.

$$\overrightarrow{Best(t)} = \overrightarrow{pos_t}(t) if\ f_1(t) < f_{best}(t) \quad (2)$$

In iteration t, $\boldsymbol{N}est(t)$ represents the position of the best bird, $Posi(t)$ represents position of ith red kite, $fi(t)$ represents value of bird evaluation function in iteration t, and $fbest(t)$.

$$D = (\exp\left(\frac{t}{t\_max}\right) - \left(\frac{t}{t\_max}\right)^{-10} \quad (3)$$

where t current iteration and t_max is maximum iteration

$$\overrightarrow{pos_t^{new}}(t+1) = \overrightarrow{pos_t}(t) + \overrightarrow{p_{mi}}(t\text{-}1) \quad (4)$$

$$\overrightarrow{p_{mi}}(t+1) = D(t) \times \overrightarrow{p_{mi}}(t) + \overrightarrow{SC}(t) \odot \left(\overrightarrow{pos_{rws}(t)} - \overrightarrow{pos_i(t)}\right) + \overrightarrow{UC(t)} \odot \overrightarrow{(Best\ (t) - pos_i(t)} \quad (5)$$

In this equation, $Posrws(t)$ represents the bird location chosen by the roulette wheel in iteration t, $posnewi(t+1)$ represents the bird's new position, and $\boldsymbol{SO}$ and UC are random vectors.

$$\overrightarrow{pos_t^{new}(t-1)} = \max\left(\min\left(\overrightarrow{pos_t^{new}}(t+1) + ub\right), ib\right) \quad (6)$$

In this scenario, $Posi(t+1)$ equals $posnewi(t+1)$. As previously noted, SC and UC random vectors that represent each bird's voice of danger and solidarity. They are obtained using the following relation:

$$\begin{Bmatrix} \overrightarrow{SC\ (t+1)=r_1} \\ \overrightarrow{UC\ (t+1)=r_2} \end{Bmatrix} if\ rand \le 0.5$$

$$\begin{Bmatrix} \overrightarrow{SC\ (t+1)=r_3} \\ \overrightarrow{UC\ (t+1)=r_1} \end{Bmatrix} otherwise \quad (7)$$

The random vectors $r1\rightarrow$, $r2\rightarrow$, and $r3\rightarrow$ are in [1, 2], [1, 3], and [0, 1], respectively. Figure 3. Shows the flow chart of red kite Optimization

3.5 Deep Kronecker neural network Via Classification

DKNN is a neural network architecture that uses the Kronecker product to represent complicated dependencies and reduce number of parameters in DL models. The theoretical research demonstrates that, under appropriate conditions, KNNs cause a faster decline of the loss than feed-forward networks. Figure 4 Schematic of a three hidden-layers Deep Kronecker neural network.

Let's begin by supposing that we view n samples of images that are represented by matrices. $X_i \in R^{d\times p}$ and Scalar replies y_i, i=1…, n The answer y_i, is assumed to fit a generalized linear model:

$$y_i|X_i) \sim P(y_i|X_i) = \rho(y_i)exp\{y_i < X_i, C > -\psi(< X_i, C >)\}, \quad (8)$$

where p(.) and ψ(.) are specific known univariate functions, and $C \in R^{d\times p}$ is the target unknown coefficients matrix.

Given equation 8, for certain known link function g(.),

$$g\left(E(y_i)\right) =< X_i, C > \quad (9)$$

With (L≥2) items, a rank-R Kronecker product decay is used, and the coefficients C:

$$C = \sum_{r=1}^{R} B_L^T \otimes B_{L-1}^T \otimes .. \otimes B_1^T \quad (10)$$

Where $B_l^T \in R^{d_i\times p_i}, l = 1, \ldots., L, r = 1, \ldots., R$ are known as Kronecker factors and are unidentified matrices. It is not assumed that the sizes of B_l^r are known. But because of the Kronecker product's characteristics, they unquestionably.

Need to classify $d = \Pi_{i=1}^{L} d_l\ and\ p = \Pi_{l=1}^{L} p_i$, To make notation easier

$$B_{l,} \otimes B_{l'-1} \otimes \ldots. \otimes B_{l''} = \otimes_{l=l,}^{l''} B_k \quad (11)$$

For any matrices $B_{l,}, \ldots., B_{l''}$ with $l l' \ge l''$. Therefore, the decomposition equation 10 is written as $C = \sum_{r=1}^{R} \otimes_{l=L}^{1} B_l^T$.

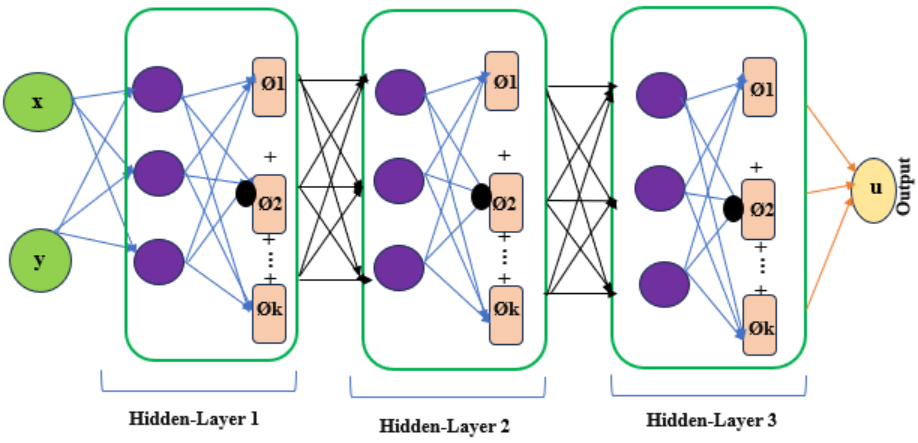


Figure 4. Deep Kronecker neural

4. RESULTS AND DISCUSSIONS

The suggested AT-DKNN method performance is compared with the existing method regarding Recall, precision, F-measure, and Rand accuracy. The efficacy of the proposed method for intrusion detection is measured using the publicly available large datasets, UNSW-NB15

4.1 Dataset Description

UNSW-NB15 is a network intrusion dataset. It employs nine various methods, including viruses, backdoors, fuzzers, and denial-of-service attacks. The collection includes raw network packets. The training set has 175,341 records, whereas the testing set contains 82,332 records, including normal and attack records. The majority of researchers have independently evaluated their developed intrusion detection system using these datasets. Fig.5.Show the attack detection of UNSW-NB15 Dataset.

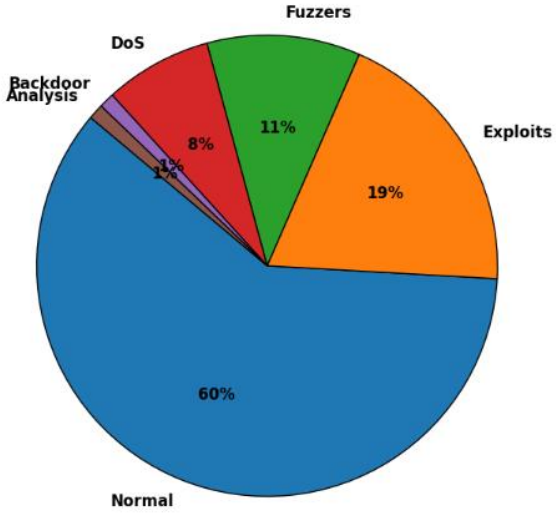


Figure 5. UNSW-NB15 Dataset

4.2 Performance analysis

Evaluation Measures, like recall, precision, F-measure, and Rand accuracy, are biased and should not be utilized unless the biases and the base case or chance values of the statistic are expressly understood.

$$\text{Accuracy}=\frac{TrueP+TrueN}{(TrueP+TrueN+FalseP+FalseN} \quad (12)$$

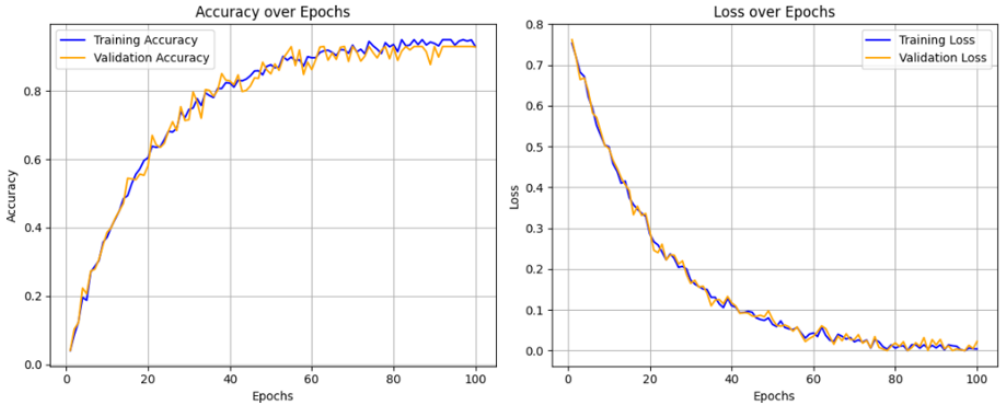
$$\text{Precision} = \frac{TP}{(TP+FP)} \quad (13)$$

$$DR = \frac{TP}{(TP+FN)} \quad (14)$$

$$\text{F1 score} = 2 \times \frac{\text{PR.DR}}{\text{PR+DR}} \quad (15)$$

$$\text{FPR} = \frac{\text{FP}}{\text{TP+FP}} \quad (16)$$

These metrics offer a thorough evaluation of model's intrusion detection capabilities. The model makes reliable predictions by utilizing detection rate and precision. In order to avoid incorrect classification, the model makes use of accuracy and false positive rate.



(a) **(b)**

Figure 6. Accuracy and Loss curve of UNSW-NB15 dataset

Figures 6(a) and 6(b) show the accuracy and loss plots of suggested intrusion detection algorithms used to validate and test classifications. The model's performance and intrusion detection capabilities are displayed in these charts. Furthermore, low loss values suggest successful learning with less overfitting throughout the training period. Figure 7 shows a performance evaluation of the proposed model.

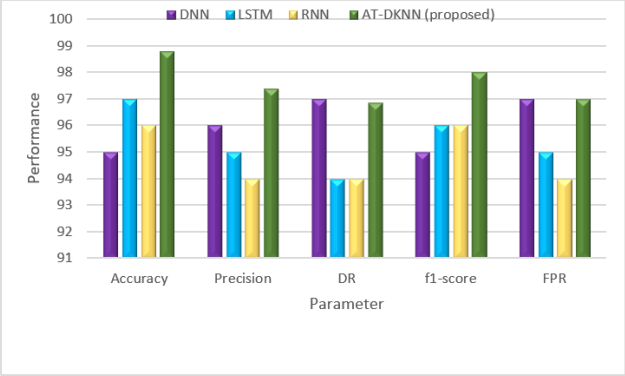


Figure 7. Performance Evaluation of AT-DKNN model

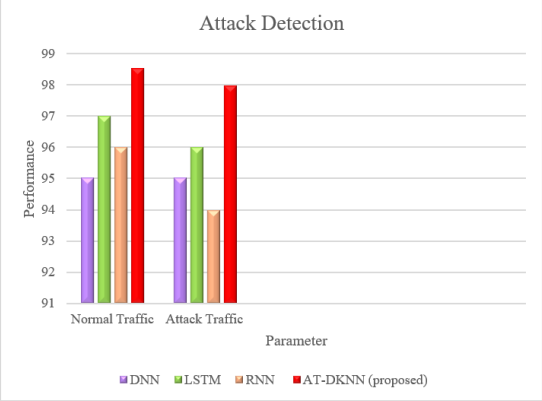


Figure 8. Attack detection of Proposed Method

Performance assessment of the proposed system using a variety of techniques relating to attack detection rate. Figure 8 illustrates Attack Detection Rate (ADR) for various methods, comparing Normal Traffic and Attack Traffic, are the Attack detection performance. The proposed range is 98.54% detection rate, while existing methods achieved only 93.85% for DNN, 96.8% for LSTM, 97.12% for RNN.

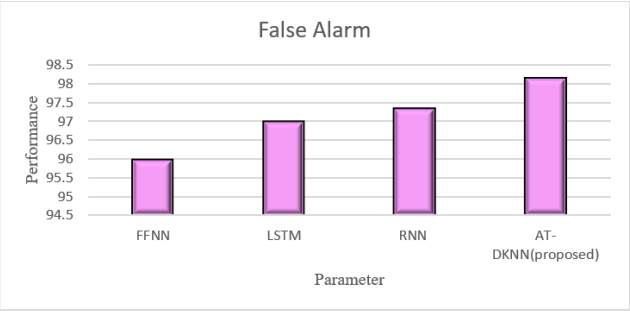


Figure 9. False alarm rate for Proposed Method

Figure 9 represents the false alarm rate 98.33% for various IDS across the false detection rates. Overall, the proposed AT-DKSNN demonstrates superior effectiveness in minimizing false alarms compared to the other methods.

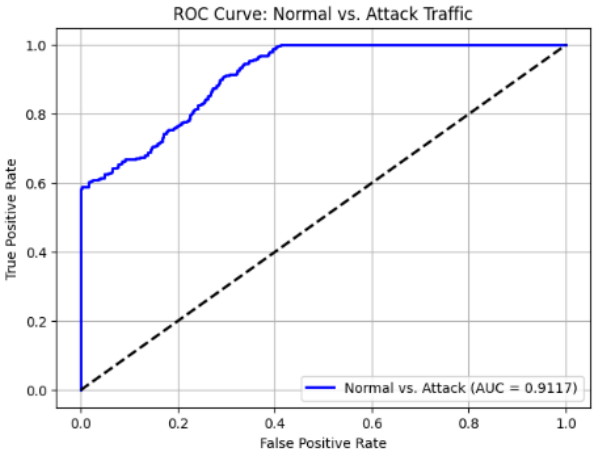


Figure 10. Graph of ROC curve

Figures 10 show the ROC of Normal Attack and Attack traffic classification results using UNSW-NB15. The suggested AT-DKSNN achieved an AUC of 0.9901 for Normal Attack and 0.9710 for Attack traffic cases that were established by FPR parameters for UNSW-NB15 dataset.

4. CONCLUSION

In this study a novel AT-DKNN framework is to detect attack traffic based on deep learning to improve cloud security. The process begins with data collecting from the IoT device. The acquired dataset undergoes preprocessing, which includes data normalization to standardize the inputs. The Spatio-Temporal Graph Neural Network (STGNN) is utilized for feature extraction, leveraging both spatial and temporal connections to build rich feature representations. The feature extraction was then followed by feature selection via Red Kite optimization to identify the most relevant characteristics and reduce dimensionality. The optimized features are sent into a Deep Kronecker Neural Network for classification which are classified as normal or attack traffic. Attack traffic is identified for further action, while normal traffic is safely sent to the cloud environment. The accuracy approach for the AT-DKSNN method is 98.33% and DNN, LSTM, CNN methods achieve a low accuracy of 96.26%, 92.32%, and 94.25%, respectively. The suggested technique efficiency is assessed utilizing accuracy, precision, recall, f1score, and FPR. Future work will further enhance the scalability and adaptability of the AT-DKSNN system to handle growing intrusion patterns and larger, more diverse datasets.

CONFLICTS OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

FUNDING STATEMENT

Not applicable.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

REFERENCES

[1] T.A. Devi, and A. Jain, "Enhancing Cloud Security with Deep Learning-Based Intrusion Detection in Cloud Computing Environments", *In 2024 2nd International Conference on Advancement in Computation & Computer Technologies (InCACCT),* pp. 541-546, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[2] D.P.R. Sanagana, and C.K. Tummalachervu, "Securing Cloud Computing Environment via Optimal Deep Learning-based Intrusion Detection Systems", *In 2024 Second International Conference on Data Science and Information System (ICDSIS),* pp. 1-6, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[3] S. Mohammed, and S. Rangu, "To secure the cloud application using a novel efficient deep learning-based forensic framework", *Journal of Interconnection Networks*, vol. 24, no. 01, pp. 2350008, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[4] S. Hizal, Ü. Çavuşoğlu, and D. Akgün, "A new deep learning-based intrusion detection system for cloud security", *In 2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA),* pp. 1-4, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] K. Devi, D. Paulraj, and B. Muthusenthil. "Deep Learning Based Security Model for Cloud based Task Scheduling", *KSII Transactions on Internet & Information Systems*, vol. 14, no. 9, 2020[CrossRef] [Google Scholar] [Publisher Link]

[6] S. Ahmad, M. Arif, S. Mehfuz, J. Ahmad, and M. Nazim, "Deep Learning-based Cloud Security: Innovative Attack Detection and Privacy Focused Key Management", *IEEE Transactions on Computers*, vol. 99, pp. 1-12, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[7] G. Loukas, T. Vuong, R. Heartfield, G. Sakellari, Y. Yoon and D. Gan, "Cloud-based cyber-physical intrusion detection for vehicles using deep learning", *Ieee Access*, vol. 6, pp. 3491-3508, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[8] P. Varun, and K. Ashokkumar, "Intrusion detection system in cloud security using deep convolutional network", *Appl. Math. Inf. Sci,* vol. 16, no. 4, pp. 581-588, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] E. Balamurugan, A. Mehbodniya, E. Kariri, K. Yadav, A. Kumar, and M.A. Haq, "Network optimization using defender system in cloud computing security-based intrusion detection system withgame theory deep neural network (IDSGT-DNN)", *Pattern recognition letters*, vol. 156, pp. 142-151, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] A. Aldallal, and F. Alisa, "Effective intrusion detection system to secure data in cloud using machine learning', *Symmetry,* vol. 13, no. 12, pp. 2306, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[11] D. Selvapandian, and R. Santhosh. Deep learning approach for intrusion detection in IoT-multi cloud environment. Automated Software Engineering, vol. 28, no. 2, pp. 19, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[12] C. Saadi, and H. Chaoui. "Cloud computing security using IDS-AM-Clust, Honeyd, honeywall and Honeycomb", *Procedia Computer Science*, vol. 85, pp. 433-442, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] D. Akgun, S. Hizal, and U. Cavusoglu, "A new DDoS attacks intrusion detection model based on deep learning for cybersecurity", *Computers & Security*, vol. 118, pp. 102748, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] D. Srilatha, and N. Thillaiarasu, "Implementation of Intrusion detection and prevention with Deep Learning in Cloud Computing", *Journal of Information Technology Management*, vol. 15(Special Issue), pp.1-18, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[15] D.B. Salvakkam, V. Saravanan, P.K. Jain, and R. Pamula, "Enhanced quantum-secure ensemble intrusion detection techniques for cloud based on deep learning", *Cognitive Computation*, vol. 15, no. 5, pp.1593-1612, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[16] C. Kavitha, T.R. K, N. Gadekallu, B.P. Kavin, and W.C. Lai, "Filter-based ensemble feature selection and deep learning model for intrusion detection in cloud computing", *Electronics*, vol. 12, no. 3, pp. 556, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[17] S.Y. Ali, U. Farooq, L. Anum, N.A. Mian, M. Asim, and T. Alyas. Securing cloud environments: a Convolutional Neural Network (CNN) approach to intrusion detection system. Journal of Computing & Biomedical Informatics, 6(02), pp.295-308, 2024. [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

J. Joshua Bapu received his B.E degree in Electronics and Communication Engineering in 2006 from Anna University, Chennai, India , M.E degree in Applied Electronics in 2009 from Anna University, Tirunelveli, India and Ph.D in Information and Communication Engineering in 2022 from Anna University, Chennai, India . Presently he is working as an Associate Professor in Department of Electronics and Communication Engineering at Vel Tech Multi Tech Dr.Rangarajan Dr.Sakunthala Engineering College, Chennai, India. His research interests includes Image Processing and Embedded systems. He has published several Research papers in International Journals. He is a life time member

Ashis Kumar Mishra received his B. Tech degree in Computer Science and Engineering from Biju Patnaik University of Technology (BPUT). He obtained his M.Tech degree in Computer Science and Engineering from KIIT University, Bhubaneswar, Odisha and Ph.D. in Computer Science and Engineering in Biju Patnaik University of Technology (BPUT), Rourkela, Odisha, India. He is currently working as an Assistant Professor in Computer Science and Engineering Department (Presently Known as School of Computer Science) at Odisha University of Technology and Research (OUTR), Bhubaneswar, Odisha, India. He Teaches courses in Programming in C, Database Management System, Embedded System, Real-Time Systems, Parallel and Distributed Systems, IOT, Cloud Computing, Big Data Analytics, Cryptography and Network Security, Operating System, Human Computer Interaction and etc. He has more than 17 years of experience in teaching and research. His research area of interest includes Cloud Computing and Big Data Analytics. He has published1 book and many articles in international journals, conferences and Patent. He guides undergraduate and Postgraduate students in different project.

Arrived: 12.01.2025
Accepted: 22.02.2025