International Journal of Engineering Sciences & Research Technology

(A Peer Reviewed Online Journal)
Impact Factor: 5.164

IJESRT

IJESRT
INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

USING AI AND MACHINE LEARNING TO SECURE CLOUD NETWORKS: A MODERN APPROACH TO CYBERSECURITY

Siddharth Konkimalla[1], Manikanth Sarisa[2], Mohit Surender Reddy[3], Janardhana Rao Sunkara[4], Shravan Kumar Rajaram [5], Sanjay Ramdas Bauskar[6]
[1]Amazon Com LLC Network Development Engineer
[2]Ally Financial Inc Principal Software Engineer
[3]Microsoft Support Escalation Engineer
[4]Siri Info Solutions Inc. Sr. Oracle Database Administrator
[5]Support Escalation Engineer
[6]Sr. Database Administrator

DOI: 10.5281/zenodo.14066056

ABSTRACT

The need to ensure the safety of network resources has grown in tandem with the lightning-fast advancement of computing power and the exponential growth of network sizes. The design and optimisation of the method for detecting network security events to adapt to the characteristics of a cloud environment is the basis for the effectiveness of OpenFlow and is a key issue in the cloud architecture approach to detecting network security events. This paper's goal is to examine the cloud architecture method for network security detection. The purpose of this research is to examine how well four different classification models—DT, SVM, and CNN-LSTM—protect cloud networks against cyberattacks. We test the models' detection and prevention capabilities on the 374,661-sample, 19-characteristic WSN-DS dataset. The CNN-LSTM model stands out with the highest accuracy of 94.4%, complemented by precision and recall values of 95.9%, demonstrating its robust classification capabilities. Our findings reveal critical insights into the effectiveness of AI and ML techniques in securing cloud environments. This research contributes to the ongoing efforts to improve cloud security through advanced analytical methods and highlights the superiority of the CNN-LSTM model for practical applications in threat detection.

KEYWORDS: Cloud Security, Data Privacy, Cybersecurity, Artificial Intelligence, WSN-DS data, Machine Learning

1. INTRODUCTION

Robust cybersecurity is now a must for organisations globally in this digital age when cyber threats and data breaches are ever-present. The rapid advancement of technologies, particularly artificial intelligence (AI) and cloud engineering, has opened new horizons in cybersecurity, offering unprecedented opportunities to bolster defences against cyber-attacks[1][2]. Nevertheless, there are significant legal and ethical questions that arise when AI and ML are integrated into cybersecurity. To make sure these technologies are used responsibly, we need to examine issues like data privacy, algorithmic bias, and the consequences of automated decision-making thoroughly[3][4]. Cloud computing must be protected for user data in order for services to be dependable. The usual suspects in cloud computing security include data misuse, hostile insiders, unsecured interfaces and access points, common technical issues, data loss, and hijacking. Thus, installing cloud computing successfully necessitates a precise comprehension of cloud security [5][6].

The deployment of potential solutions required by consumers is hindered by various sorts of attacks, which affect cloud providers and administrators [7]. The reason for this is that various attack types provide different risks, and the relative importance of these threats varies based on other cloud service customers' security needs[8][9]. To fulfil critical security needs as service providers, security administrators will assess threats and put safeguards in place. It is almost impossible to design a system that is totally secure[10], yet security may be enhanced [11][12]. As a result, identifying security risks and the corresponding remedies, such as accountability, authentication, and privacy protection, are essential [13][14]. However, in addition to other common services like computing, storage, and networking, cloud providers have recently started to provide a variety of AI tools and frameworks to make it simple to create and utilise new ML models[15]. The ability of Cloud Boost to make resources and knowledge about AI available and thus accessible to everyone is one of the advantages of Cloud Boost[16].

Artificial Intelligence (AI) has fundamentally transformed the landscape of cybersecurity, offering advanced capabilities that significantly enhance threat detection and response. By leveraging machine learning and predictive analytics, AI systems can analyse vast amounts of data to identify patterns and anomalies indicative of potential cyber threats. This capability is crucial for managing the complexity and volume of modern cyber threats, providing organisations with a powerful tool to detect and mitigate risks more effectively. Machine learning algorithms, a subset of AI, are particularly valuable in cybersecurity. These algorithms are designed to learn from historical data and identify patterns that may signal malicious activit

Cybersecurity has been revolutionised by Artificial Intelligence (AI) which delivers superior elements that have increased the capabilities of threat identification and mitigation[17]v. AI systems integrate ML and big data analytics to search through the large volumes of data for symptoms of cyber threats[18]. It becomes necessary in the age of high complexity and quantity of threats to provide organisations the powerful instruments to observe and respond to risks. Machine learning as a type of AI is most useful in cybersecurity. These programs may study past data in order to spot trends that might indicate harmful behaviour[19].

2. MOTIVATION AND CONTRIBUTION OF STUDY

The study's impetus stems from the growing complexity and frequency of cyberattacks that target cloud networks, which are difficult for conventional security techniques to manage. As cloud environments host more sensitive data, there is a critical need for advanced, automated solutions. This study seeks to leverage AI and machine learning to enhance the detection and prevention of cyberattacks, ensuring more robust security in cloud-based infrastructures. The area of cloud network security has benefited greatly from this study's several important contributions.

- Employ the ML models with the help of the WSN-DS dataset.
- Implements advanced data preprocessing, including SMOTE for balancing, Min-Max scaling, and Chi-Square feature selection, to optimise model performance.
- Demonstrates the effectiveness of ML models (CNN-LSTM, DT, and SVM) for detecting various types of cloud-based cyberattacks.
- Comprehensive evaluation of model performance using accuracy, precision, and recall metrics, offering insights into each model's strengths.

3. STRUCTURE OF PAPER

The following is a synopsis of the remaining paper. In Section II, we provide a literature review of cloud security based on AI and ML. While Section IV delves into the analysis and discussion of the outcomes, Section III lays out the methodology and strategy. The study's conclusions and suggestions for further research are detailed in Section V.

4. LITERATURE REVIEW

This section provides some previous work on cloud security networks for cybersecurity based on machine learning.

In this paper, Fang, Zhang and Huang, (2021) was presented CyberEyes, a model for cybersecurity entity recognition that makes use of graph CNNs to extract non-local relationships. Our model outperformed the typical CNN-BiLSTM-CRF mode, which achieved an F1 score of 86.49% on the cybersecurity corpus, in the assessment trials, reaching a score of 90.28% under the gold standard for NER[20].

This research used, Umamaheshwari, Kumar and Sasikala, (2021) by use of a DTC. Feature selection utilising the MRMR algorithm, the Relief algorithm, the Kruskal-Wallis (KW) test for statistical analysis, and the Fisher score were all tested in an effort to shorten the time it takes to identify attacks. Relevant performance indicators are used to assess the suggested feature selection approaches. The following metrics were measured using MRMR feature selection: accuracy (98.58%), sensitivity (92.81%), specificity (93.86%), and training time (15.12 seconds), in that order [21].

This research, Krishnan and Singh, (2021) developed a classifier using cost-sensitive ML and trained it on the WSN-DS dataset, which includes examples of flooding, TDMA/scheduling, black-hole, and grey-hole attacks. A Cost-Sensitive Bootstrapped Weighted Random Forest (CSBW-Random Forest) was suggested in light of this, and it outperformed previous efforts. The accuracy, precision, recall, and F1-score of our approach are all 0.997, and the per-class performance scores fall between 0.95 and 0.99, which is a considerable improvement over previous research [22].

In this paper, Yasarathna and Munasinghe, (2020) primary emphasis was on employing one-class classification algorithms, namely Autoencoder and OCSVM, to analyse data from cloud networks in order to spot abnormalities. Our results show that Autoencoder is 96.02% accurate while OCSVM is 79.05% accurate when it comes to identifying outliers. Furthermore, they delve further into the efficacy of a one-class classification system by using an additional benchmarked data set, UNSW-NB15. A 99.10% accuracy rate for Autoencoder and a 60.89% accuracy rate for OCSVM were achieved there[23].

This research, Hachimi et al., (2020) emphasises the implementation of a multi-stage ML-IDS in 5G C-RAN capable of detecting and categorising four distinct jamming assault types: reactive, continuous, random, and deceptive. Simplifying C-RAN structures and reducing false negatives is how this deployment improves security. Experimental testing of the proposed method is carried out using WSN-DS, a wireless dataset developed for intrusion detection purposes. A FNR of 7.84% contributes to the final assault classification accuracy of 94.51%[24]

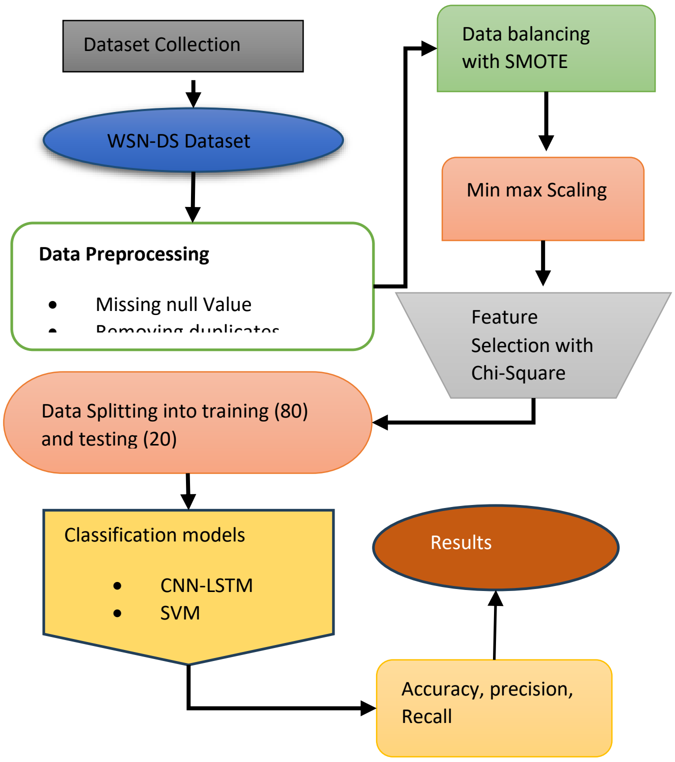
Table I includes detailed insights, including dataset limitations and possible future research directions, for each study on cloud security network enhancements using machine learning.

Table i. Summary of previous work on cloud security networks for cybersecurity using machine learning

Authors	Data	Methods	Findings	Limitations	Future Work
Fang, Zhang, and Huang (2021)	Cybersecurity corpus	CyberEyes model with Graph Convolutional Neural Networks (GCN) for NER using non-local dependencies	Achieved an F1 score of 90.28%, outperforming the CNN-BiLSTM-CRF model (F1 score of 86.49%) in NER tasks	Limited to NER tasks in cybersecurity, requires labelled data for gold-standard evaluation.	Potential integration with other NER tasks and domains, enhancing graph-based dependency extraction capabilities
Umamaheshwari, Kumar, and Sasikala (2021)	WSN-DS dataset	Feature selection using Correlation Score, Fisher Score, Kruskal-Wallis test, MRMR, and Relief; Decision Tree classifier.	Using MRMR, achieved 98.58% accuracy, 92.81% sensitivity, 98.46% specificity, and 93.86% precision, with 15.12 sec training time.	Limited to attack detection in WSNs; computation time may still be a concern in some resource-constrained scenarios	Further optimise feature selection to reduce training time improve adaptability for other attack scenarios.
Krishnan and Singh (2021)	WSN-DS dataset	Cost-sensitive bootstrapped Weighted Random Forest (CSBW-RF) for handling class imbalance.	Achieved 0.997 accuracy; per-class precision, recall, and F1 scores between 0.95 and 0.99, demonstrating improved performance over existing methods	Focused on WSN-specific attacks, potential limitation in broader applications, like other network types	Extending the CSBW-RF to diverse datasets and integrating with multi-class and unsupervised learning techniques
Hachimi et al. (2020)	WSN-DS	Multi-stage ML-based IDS	Achieved 94.51% accuracy with a 7.84% false negative rate in detecting jamming attacks.	Relatively high false negative rate for critical attack types.	Enhance false negative mitigation; explore adaptability for non-5G network environments.
Yasarathna and Munasinghe (2020)	YAHOO Synthetic, UNSW-NB15	OCSVM, Autoencoder	Achieved 96.02% accuracy on YAHOO data and 99.10% on UNSW-NB15 with Autoencoder.	Kernel-based OCSVM had lower accuracy, especially on UNSW-NB15.	Further, refine neural networks for cloud data anomalies and test on real-world cloud datasets.

5. METHODOLOGY

The general workflow for securing cloud networks using AI and machine learning begins with data collection, where the WSN-DS dataset, comprising 19 features and 374,661 samples, is used. In the pre-processing stage, missing values (if any) are handled, and duplicates are removed to ensure data integrity. The dataset is balanced using SMOTE, which generates synthetic samples for minority classes to alleviate class imbalance and ensure that all attack types are represented equally. After that, scaling is done using Min-Max scaling, which normalises the features by transforming them to a range between 0 and 1. Afterwards, the Chi-Square approach is used for feature selection in order to reduce irrelevant variables and discover the most important characteristics for the classification task. The models used for this task include DT, SVM, and CNN-LSTM, all trained on the preprocessed and balanced data. Lastly, the performance metrics—accuracy, precision, and recall—are evaluated to measure the models' effectiveness in securing cloud networks, with values closer to 1 indicating better performance. The following workflow of research design is shown in Figure 1.

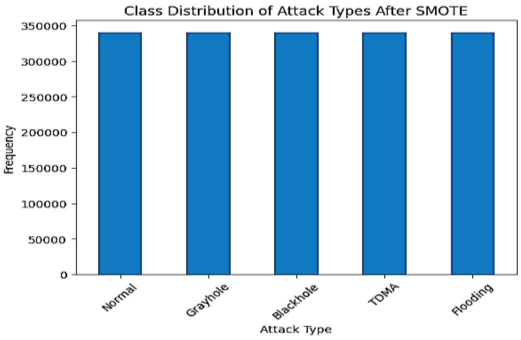


Flowchart for cloud security network

Each step and phase of Figure 1 Flowchart for cloud security network are listed below:

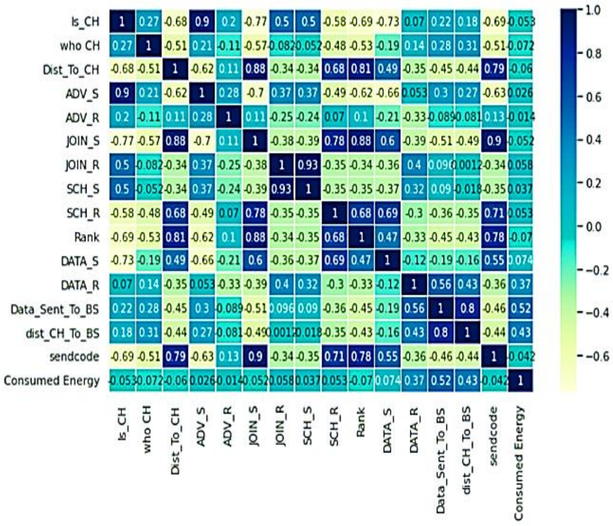
Data Collection

The dataset utilised in the experiment was created by Almomani and is a simulation of a WSN-DS. The target variable (Attack Type) is one of nineteen attributes. There were no missing or null characteristic values among the 3,74,661 data samples. The analysis of the dataset is such that insights into data are visible in the following Figures 2,3 and 4.



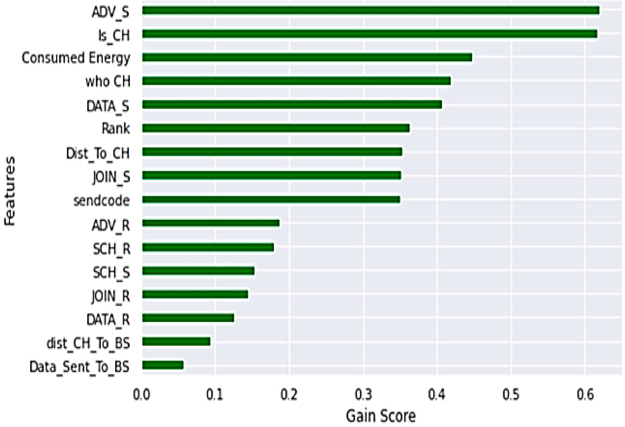
Class distribution of attack

Figure 2 shows the Class Distribution of Attack Types After SMOTE illustrates the balanced frequency of various attack types shows in x-axis. The y-axis shows the frequency 0 to 350000. Each category has a nearly identical representation, showing that SMOTE successfully addressed the class imbalance in the dataset.



Correlation matrix for features

Figure 3 shows a heatmap of a correlation matrix with a colour gradient that runs from dark blue to dark red, indicating values ranging from -1.0 to 1.0. The rows and columns are labelled with numerous acronyms, like "is_CH," "Dist_to_CH," "ADV_S," "JOIN_R," and more. Each cell in the grid displays a numerical value that correlates to the colour scale, showing the magnitude and direction of the correlation between the variables. The bottom row and the far-right column are labelled “Consumed Energy,” with appropriate values and colours, indicating their interaction with other variables in the matrix.



Bar graph for feature score

Figure 4 is a horizontal bar graph with several characteristics plotted against a 'Gain Score' on an x-axis that runs between 0 and 0.6. A y-axis displays characteristics such as 'ADV_S,' 'whois_CH,' 'Consumed Energy,' 'Rank,' 'DATA_S,' 'Dist_To_CH,' 'JOIN_S,' and others, with certain bars approaching 0.6 indicating a greater gain score for those qualities. This graph is employed in data analysis or machine learning to determine important characteristics in a model.

Dataset Preprocessing

Data preparation is a continuous process that tries to transform the raw data into more usable and comprehensible form. Where specific data points are absent in a dataset that were referred to as missing values, it can be expressed by blank cells or null values and sometimes by special characters such as “NA” or “unknown”. Lack of these data hampers the analysis of data and also introduces the biassing or wrong conclusion. In order to ensure that the data is correct and trustworthy for further analysis or modelling, removing duplicates is a crucial step in data cleaning and preprocessing.

Balancing with SMOTE

A small dataset is ideal for SMOTE's performance. To make matters worse, SMOTE's efficiency plummets as the dataset size increases since it takes a long time to generate false data points. In addition, SMOTE has a significant probability of overlapping data points for the minority class while making fictional data points[25]. A following Eq. (1) of smote is:

$$x_{new,\ attr} = x_{i,attr} + rand(0,1)x\ (x_{ij,attr} - x_{i,attr} \qquad (1)$$

Min-Max Scaling

Equation 2 shows how the Min-max Scaler changes an attribute's scale by dragging its values down the X-axis until the new attribute fits in the range of [0, 1].

$$x_1' = \frac{x_1 - x_{min}}{x_{max} - x_{min}} \qquad (2)$$

This approach uses the range of a feature as a scaling factor and the lowest value of the characteristic as a translational term.

Feature Selection with Chi-Square

To decrease a number of irrelevant variables, feature selection approaches in WSN data pre-processing aim to filter the input variables down to those most likely to be related with the intrusion assault. For this reason, choose Chi-squar feature selection methods. The independence of characteristics with regard to the class is measured by chi-squared. A score is not computed until the feature and class are believed to be independent [26]. A highly reliant connection is indicated by a high score. The following Eq. (3)

$$x^2 = \frac{(observed\ frequency - expected\ frequency)^2}{expected\ frequency} \quad (3)$$

Where:

Observed frequency = An amount of class observations.

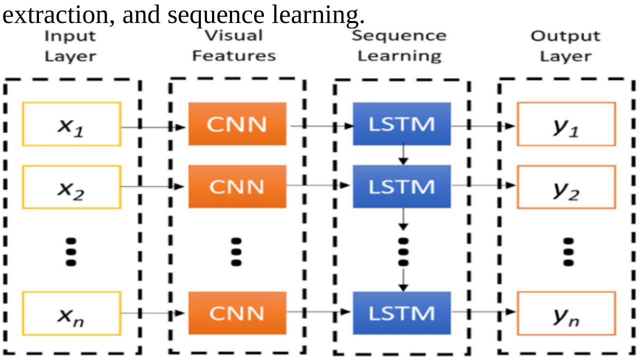
Expected frequency = The predicted number of class observations in the absence of a link among the feature and the target attribute.

Data Splitting

The dataset was divided into two parts: a testing set that is totally concealed by the training process and a training set utilised to train the detection algorithm. The 80:20 techniques are used by the two subgroups. The test set uses 20% of the whole dataset, whereas the training and validation sets utilise 80%.

Classification with CNN-LSTM model

CNN-LSTM blends the ability of CNNs to retrieve features with the ability of LSTM layers to guess sequences. The CNN-LSTM is often used for image and video tagging and activity recognition. The two work together to solve problems with visual time series forecasting and text annotation creation from image sequences. The CNN-LSTM network's layers are shown in Figure 5 in the following order: input, output, visual feature extraction, and sequence learning.



Architecture of the CNN-LSTM Network

To differentiate between malicious and benign users, a CNN-LSTM model is recommended. This may be a great way for many companies to keep hackers out of their systems. The attack label is unnecessary for testing the proposed model[27][28]. The model takes features as input so that it may correctly associate labels with input properties. The CNN-LSTM model is highly recommended because of its extensive range of features.

This ensemble model used CNN layers to extract features and LSTM layers to handle the sequential nature of the input. For multi-class classification, the model then uses a fully connected layer with sigmoid activation, producing five different output classes. We will next use the "Adam" optimiser and a learning rate of 0.001 to construct our model.

Performance Metrics

Four criteria are used to assess the findings of this study: recall (RE), accuracy (ACC), precision (PR), and precision (PR). Each of these standards has a numeric value between zero and one. Performance improves as it gets closer to 1, and it drops as it gets closer to 0. The formula for calculating these performance assessment measures is:

Accuracy: Accuracy (Acc), a frequently used indicator for classification performance, is expressed as the proportion of correctly classified samples to all samples, as shown in Equation (4).

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \quad (4)$$

Where:

- **True Positives (TP):** TP stands for positive classes that were accurately forecasted.
- **False Positives (FP):** A positive class that was incorrectly forecasted is FP.
- **True Negatives (TN):** A negative class that was accurately forecasted in TN.
- **False Negatives (FN):** FN denotes the negative classes that were incorrectly anticipated.

Precision: The capacity of a model to recognise only relevant things is known as precision. It represents the percentage of predictions that come true. The precision is calculated as (5):

$$Precision = \frac{TP}{TP+FP} \qquad (5)$$

Recall: It is calculated by dividing the total number of relevant samples by the number of accurate positive outcomes. Here is the mathematical representation (6):

$$Recall = \frac{TP}{TP+FN} \qquad (6)$$

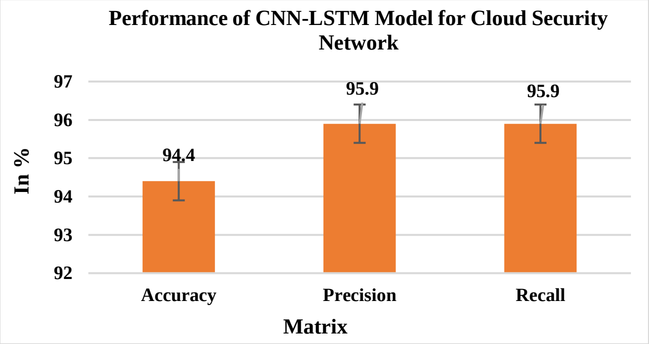
The following matrix is useful for generalising the model performance for cancer prediction.

6. RESULTS & DISCUSSION

The section evaluates the model's effectiveness. Every single trial ran on a Windows 11 PC with a 3.80 GHz Intel Core i7 CPU, 16 GB of RAM, and all the necessary hardware components. This section discusses the simulated outcomes of cloud security using ML approaches. The following models, like DT[29], SVM[30], and CNN-LSTM, are implemented on the WSN-DS dataset across performance matrices like accuracy, precision, and recall.

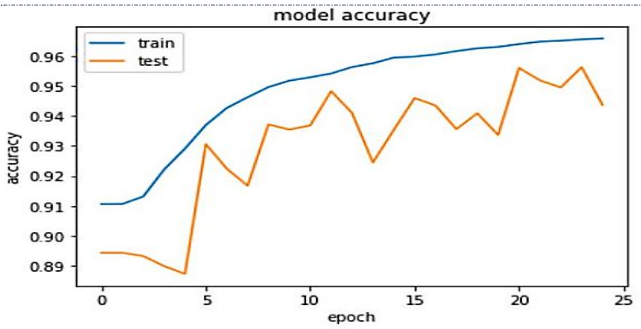
CNN-LSTM MODEL PERFORMANCE ON WSN-DS DATASET

Metric	CNN-LSTM
Accuracy	94.4
Precision	95.9
Recall	95.9



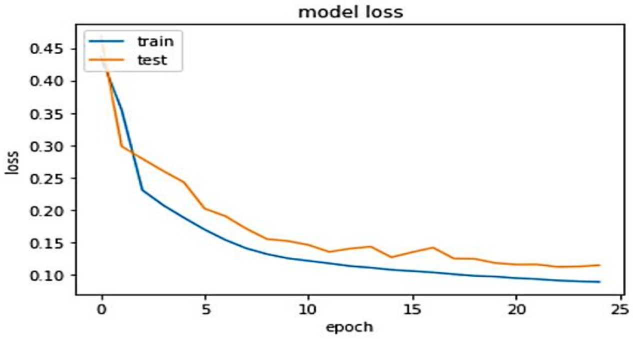
Bar Graph for CNN-LSTM model Performance

The results of running the CNN-LSTM model on the provided data are shown in Figure 6 and Table II. Employing a CNN-LSTM model, an accuracy level was calculated to be 94.4%; therefore, it duly classified 94.4% of occurrences of the dataset. It also obtained a precision of 95.9%, meaning that 95.9% of the predicted positives were actual positives. Also, the specificity of the model was 98%, which means the model labelled 98% of correctly identified negative instances as negative. The above metrics point that the efficiency of the CNN-LSTM model in terms of distinguishing between true positive and false positive rates is quite high and hence the proposed technique is adequate enough for the classification of the given task.



CNN-LSTM Model Training and Validation Accuracy with 25 Epochs

A line graph representing model accuracy throughout 25 epochs of training and testing is shown in Figure 7. On the horizontal axis, it is possible to discover epochs from 0 to 25, on the vertical axis, there are results regarding accuracy from 0.89 to 0.97. The plot has 2 curves: blue colour for training data accuracy, which increases from 0.90 to around 0.97, and orange colour for testing data accuracy, which fluctuates, yet it increases from 0.89 to approximately 0.93 in the 25th epoch. This graph helps to make out how the performance of the model grows with time.



CNN-LSTM Model Training Loss with 25 Epochs.

Figure 8 shows a line graph for CNN-LSTM model loss, which depicts the loss value across 25 epochs for both the training and testing datasets. An x-axis indicates epochs (0–25), whereas the y-axis represents loss (0–0.45). The blue line for training loss indicates a dramatic dip at first, then gradually levels out, demonstrating that loss decreases as training continues. The orange line for testing loss lowers as well, but with more oscillations, indicating model performance variability on unknown data. Throughout the testing and training phases, the loss of the model changes with time, as seen in this graph.

COMPARATIVE ANALYSIS OF CLOUD SECURITY NETWORK ON WSN-DS DATASET

Models	Accuracy	Precision	Recall
DT	84.3523	82.1	98.0
SVM	89	88	92
CNN-LSTM	94.4	95.9	95.9

Table III above displays the outcomes of comparing a model's performance. An CNN-LSTM model shines out when compared to others, demonstrating its better performance in classification tests with an accuracy of 94.4% and great precision and recall values of 95.9% each. In contrast, the DT model has the lowest accuracy at 84.35%, although it achieves a high recall of 98.0%, which means it is better at identifying actual positives but less precise at 82.1%. The SVM accuracy score of 89%, with SVM slightly outperforming. The CNN-LSTM model is a most efficient for a task at hand since it offers the greatest overall balance between accuracy, precision, and recall.

7. CONCLUSION & FUTURE WORK

The current trend in Internet growth, towards cloud computing, has caused a great deal of anxiety among internet users. Research on the best practices for constructing a safe cloud computing environment is now at the forefront of the computer science community. This research illustrates the significant potential of leveraging AI and ML techniques for securing cloud networks against cyber threats. The comparative performance analysis of various classification models, including DT, SVM, and CNN-LSTM, underscores a superior efficacy of the CNN-LSTM model, which achieved an accuracy of 94.4% along with high precision of 95.9 and recall of 95.9 metrics. The study does admit to certain caveats, however, such as the fact that it only used one dataset—which could not be representative of the variety of cyber threats that exist in the actual world. Another consideration is that models with high computational complexity, like CNN-LSTM, could be difficult to implement in settings with limited resources. Future work should focus on exploring more diverse datasets to validate the model's effectiveness across different contexts, as well as investigating the integration of ensemble learning techniques to enhance classification performance further.

REFERENCES

1. V. K. Yarlagadda and R. Pydipalli, “Secure Programming with SAS: Mitigating Risks and Protecting Data Integrity,” *Eng. Int.*, vol. 6, no. 2, pp. 211–222, 2018.
2. R. Arora, S. Gera, and M. Saxena, “Mitigating Security Risks on Privacy of Sensitive Data used in Cloud-based ERP Applications,” in *2021 8th International Conference on Computing for Sustainable Global Development (INDIACom)*, 2021, pp. 458–463.
3. M. A. Rassam, M. A. Maarof, and A. Zainal, “A survey of intrusion detection schemes in Wireless Sensor Networks,” *Am. J. Appl. Sci.*, 2012, doi: 10.3844/ajassp.2012.1636.1652.
4. L. Fei, Y. Chen, Q. Gao, X. H. Peng, and Q. Li, “Energy hole mitigation through cooperative transmission in wireless sensor networks,” *Int. J. Distrib. Sens. Networks*, 2015, doi: 10.1155/2015/757481.
5. X. Fan, J. Yao, and N. Cao, “Research on cloud computing security problems and protection countermeasures,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2019. doi: 10.1007/978-3-030-37337-5_44.
6. A. P. A. Singh, “Streamlining Purchase Requisitions and Orders : A Guide to Effective Goods Receipt Management,” *J. Emerg. Technol. Innov. Res.*, vol. 8, no. 5, pp. g179–g184, 2021.
7. R. Kumar and R. Goyal, “On cloud security requirements, threats, vulnerabilities and countermeasures: A survey,” *Computer Science Review*. 2019. doi: 10.1016/j.cosrev.2019.05.002.
8. V. K. Y. Nicholas Richardson, Rajani Pydipalli, Sai Sirisha Maddula, Sunil Kumar Reddy Anumandla, “Role-Based Access Control in SAS Programming: Enhancing Security and Authorization,” *Int. J. Reciprocal Symmetry Theor. Phys.*, vol. 6, no. 1, pp. 31–42, 2019.
9. J. Thomas and V. Vedi, “Enhancing Supply Chain Resilience Through Cloud-Based SCM and Advanced Machine Learning: A Case Study of Logistics,” *J. Emerg. Technol. Innov. Res.*, vol. 8, no. 9, 2021.
10. X. Wang, D. Li, X. Zhang, and Y. Cao, “MCDM-ECP: Multi criteria decision making method for emergency communication protocol in Disaster Area Wireless Network,” *Appl. Sci.*, 2018, doi: 10.3390/app8071165.
11. M. Haus, M. Waqas, A. Y. Ding, Y. Li, S. Tarkoma, and J. Ott, “Security and Privacy in Device-to-Device (D2D) Communication: A Review,” *IEEE Commun. Surv. Tutorials*, 2017, doi: 10.1109/COMST.2017.2649687.
12. R. Goyal, “THE ROLE OF BUSINESS ANALYSTS IN INFORMATION MANAGEMENT PROJECTS,” *Int. J. Core Eng. Manag.*, vol. 6, no. 9, pp. 76–86, 2020.
13. S. Tu *et al.*, “Reinforcement Learning Assisted Impersonation Attack Detection in Device-to-Device Communications,” *IEEE Trans. Veh. Technol.*, 2021, doi: 10.1109/TVT.2021.3053015.
14. M. S. Rajeev Arora, Sheetal Gera, “Impact of Cloud Computing Services and Application in Healthcare Sector and to provide improved quality patient care,” *IEEE Int. Conf. Cloud Comput. Emerg. Mark. (CCEM), NJ, USA, 2021*, pp. 45–47, 2021.
15. S. K. R. Anumandla, V. K. Yarlagadda, S. C. R. Vennapusa, and K. R. V Kothapalli, “Unveiling the Influence of Artificial Intelligence on Resource Management and Sustainable Development: A Comprehensive Investigation,” *Technol. \& Manag. Rev.*, vol. 5, no. 1, pp. 45–65, 2020.
16. S. Deng, L. C. Yang, D. Yue, X. Fu, and Z. Ma, “Distributed Global Function Model Finding for Wireless Sensor Network Data,” *Appl. Sci.*, 2016, doi: 10.3390/app6020037.

17. V. V. Kumar, A. Sahoo, and F. W. Liou, "Cyber-enabled product lifecycle management: A multi-agent framework," in *Procedia Manufacturing*, 2019. doi: 10.1016/j.promfg.2020.01.247.
18. J. Yan, M. Zhou, and Z. Ding, "Recent Advances in Energy-Efficient Routing Protocols for Wireless Sensor Networks: A Review," *IEEE Access*. 2016. doi: 10.1109/ACCESS.2016.2598719.
19. M. Alqahtani, A. Gumaei, H. Mathkour, and M. M. Ben Ismail, "A genetic-based extreme gradient boosting model for detecting intrusions in wireless sensor networks," *Sensors (Switzerland)*, 2019, doi: 10.3390/s19204383.
20. Y. Fang, Y. Zhang, and C. Huang, "CyberEyes: Cybersecurity Entity Recognition Model Based on Graph Convolutional Network," *Comput. J.*, 2021, doi: 10.1093/comjnl/bxaa141.
21. S. Umamaheshwari, S. A. Kumar, and S. Sasikala, "Towards Building Robust Intrusion Detection System in Wireless Sensor Networks using Machine Learning and Feature Selection," in *2021 International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation, ICAECA 2021*, 2021. doi: 10.1109/ICAECA52838.2021.9675609.
22. D. Krishnan and S. Singh, "Cost-Sensitive Bootstrapped Weighted Random Forest for DoS attack Detection in Wireless Sensor Networks," in *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, 2021. doi: 10.1109/TENCON54134.2021.9707254.
23. T. L. Yasarathna and L. Munasinghe, "Anomaly detection in cloud network data," in *Proceedings - International Research Conference on Smart Computing and Systems Engineering, SCSE 2020*, 2020. doi: 10.1109/SCSE49731.2020.9313014.
24. M. Hachimi, G. Kaddoum, G. Gagnon, and P. Illy, "Multi-stage jamming attacks detection using deep learning combined with kernelized support vector machine in 5G cloud radio access networks," in *2020 International Symposium on Networks, Computers and Communications, ISNCC 2020*, 2020. doi: 10.1109/ISNCC49221.2020.9297290.
25. K. Patel, "Quality Assurance In The Age Of Data Analytics: Innovations And Challenges," *Int. J. Creat. Res. Thoughts*, vol. 9, no. 12, pp. f573–f578, 2021.
26. K. Lakshmi Devi, P. Subathra, and P. N. Kumar, "Tweet sentiment classification using an ensemble of machine learning supervised classifiers employing statistical feature selection methods," in *Advances in Intelligent Systems and Computing*, 2015. doi: 10.1007/978-3-319-27212-2_1.
27. S. B. and S. C. and S. Clarita, "AN ANALYSIS: EARLY DIAGNOSIS AND CLASSIFICATION OF PARKINSON'S DISEASE USING MACHINE LEARNING TECHNIQUES," *Int. J. Comput. Eng. Technol.*, vol. 12, no. 01, pp. 54-66., 2021, doi: http://www.iaeme.com/IJCET/issues.asp?JType=IJCET&VType=12&IType=1.
28. S. R. Bauskar and S. Clarita, "Evaluation of Deep Learning for the Diagnosis of Leukemia Blood Cancer," *Int. J. Adv. Res. Eng. Technol.*, vol. 11, no. 3, pp. 661–672, 2020, doi: https://iaeme.com/Home/issue/IJARET?Volume=11&Issue=3.
29. L. Panwar and S. Panwar, "Implementation of Machine Learning Algorithms on CICIDS-2017 Dataset for Intrusion Detection using WEKA," vol. 8, p. 2195, 2019, doi: 10.35940/ijrte.C4587.098319.
30. S. Ifzarne, H. Tabbaa, I. Hafidi, and N. Lamghari, "Anomaly Detection using Machine Learning Techniques in Wireless Sensor Networks," in *Journal of Physics: Conference Series*, 2021. doi: 10.1088/1742-6596/1743/1/012021.