Indonesian Journal of Electrical Engineering and Computer Science
Vol. 40, No. 2, November 2025, pp. 840∼849
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v40.i2.pp840-849

Intrusion detection system using hybrid CNN-LSTM model in cloud computing

Maha Mohammad Alshehri[1], Shoog Abdullah Alshehri[1], Samah Hazzaa Alajmani[3]
[1]Department of Cyber Security, College of Computer Science and Information Technology,
Taif University (TU), Taif, Saudi Arabia
[2]Department of Information Technology, College of Computer Science and Information Technology,
Taif University (TU), Taif, Saudi Arabia

Article Info

Article history:

Received Oct 30, 2024
Revised Jul 14, 2025
Accepted Oct 14, 2025

Keywords:

Cloud computing
CNN
CSE-CIC-IDS2018
Deep learning
Distributed denial of service
Internet of things
LSTM

ABSTRACT

Cloud computing has revolutionized online service delivery with its flexibility and cost efficiency. Nevertheless, the growing importance of stored data makes it a target for cyberattacks, posing security and privacy risks. This calls for effective solutions to safeguard data and infrastructure, particularly with regard to intrusion attacks and distributed attacks such as distributed denial of service (DDoS). Therefore, there is a need to develop an effective intrusion detection system (IDS) using deep learning to ensure the protection of cloud data and infrastructure. In this paper, a hybrid model aims to leverage the power of convolutional neural networks (CNNs) to analyze spatial features and extract complex patterns, while long short-term memory LSTMs are used to understand temporal data sequences and detect attacks that evolve over time to detect intrusions in cloud computing environments on the CSE-CIC-IDS2018 dataset. The model was trained and tested on DDoS attacks, and the results demonstrated high performance in detecting attacks with high accuracy and efficiency. This hybrid model achieved an accuracy of 99.88%, a precision of 99.83%, a recall of 99.94%, and an F1-score of 99.88%.

Corresponding Author:

Maha Mohammad Alshehri
Department of Cyber Security, Computer Science and Information Technology
Taif University
Email: mahaalshehri11@outlook.com

1. INTRODUCTION

A cloud infrastructure consists of a massive network with multiple internet of things (IoT)-enabled devices and applications that collect data from cloud networks, operations, real-time processing, underlying infrastructure, servers, and storage. Cloud infrastructures include services and standards for ensuring securing and controlling [1], [2]. Cloud computing has grown widely in recent years due to its dynamic and scalable nature [3]. Cloud computing is the use and delivery of resources and services over the Internet. With the adoption of cloud computing by the IoT, the need for storing and processing big data has increased [4], [5]. With the increased adoption of cloud computing with the increased adoption of cloud computing technology by cloud computing providers like Google, Amazon, and IBM, Amazon is considered a leader in the field due to its architectural features. However, the risks of targeting cloud computing have increased because it contains important and sensitive information about users or services [6], [7]. One of the essential functions of cloud computing is to deal with threats as quickly as possible, whether to users or the cloud services [8], [9]. Attacks pose serious security issues due to becoming more sophisticated. Since cloud is vulnerable to hacking

and has weak security defences, it is a target for attacks and data exposure. However, intrusion detection capabilities need to be improved. These systems often fail to recognize attack patterns, which may make them rely on traditional intrusion detection systems that may not be enough [10]. Distributed denial of service (DDoS) attacks are among the most serious attacks targeting cloud computing. DDoS is a cyberattack that aims to disrupt a website or network by overwhelming it with massive requests from multiple sources [11], [12]. Although traditional intrusion detection systems (IDS) exist, they are weak at detecting sophisticated attacks [13], [14]. Therefore, this research presents a hybrid intrusion detection system based on a sophisticated model that combines convolutional neural network (CNN) and long short-term memory (LSTM), enhancing the system's ability to analyze network data and detect attacks with higher accuracy and efficiency in cloud environments.

Numerous studies have investigated deep learning algorithms for cloud computing intrusion detection to improve cloud security using a CNN algorithm on the CSE-CIC-IDS 2018 dataset, which contains multiple attack scenarios. The CNN model is effective in detecting intrusions in cloud environments and achieved above 97% accuracy for both papers [15], [16]. Hagar and Gawali [17] proposes deep learning algorithms CNN and LSTM to improve intrusion detection systems. It uses upsampling and downsampling techniques to solve the imbalance problem in the CSE-CICIDS2018 dataset. The results showed that CNN outperformed LSTM with 98.31 accuracy. However, these papers use the algorithms separately, which may limit the model's capability to handle different data. The performance could be enhanced if the capabilities of algorithms are combined to reduce the weaknesses when used separately. Previous studies [18]–[20] have shown a hybrid deep learning model of CNN and LSTM to improve intrusion detection systems for cloud environments on CSE-CIC-IDS 2018, CIC-IDS2017 and IoTID20 datasets. Accuracy was above 97%. It was observed that balancing the dataset enhanced model performance. Al and Dener [21], the imbalance problem was solved by SMOTE and Tomek-Links algorithms on CIDDS-001 and UNSW-NB15 datasets. The accuracy was 99.83% in multi-class classification and 99.17% in binary classification.

Qazi *et al.* [22] presented a new concept based on combining CNN and RNN in a hybrid intrusion detection system (HDLNIDS). The model enhances accuracy and reduces false positives compared to traditional methods like machine learning on the CICIDS-2018 dataset. The proposed HDLNIDS system achieved an average accuracy of 98.90%. The RNN can not remember information for long periods of time, but it can solve this problem using LSTM. Khan and Haroon [23], the researchers studied how to detect intrusions in cloud computing networks using artificial neural networks (ANN). The model uses 19 features that were selected using the decision tree technique. Random oversampling and undersampling techniques were used on the CSE-CIC-IDS-2018 dataset. It reached an accuracy of 99.99% in detecting attacks. Farhan *et al.* [24], the deep learning deep neural network (DNN) model was tested to analyze the performance of flow-based attack detection. DNNs are more effective in improving intrusion detection systems. Rectified linear unit (ReLU) and Softmax activation functions achieve high classification accuracy for multiple attacks. The CSE-CIC-IDS2018 dataset was used and achieved an accuracy of 90%. It is advisable to use hybrid techniques based on CNN and RNN to refine the detection of temporal patterns of attacks. Table 1 compares previous studies regarding the models used, datasets, and the highest accuracy achieved.

Table 1. Comparison between intrusion detection systems with related studies

Reference	Year	Algorithm	Dataset	Result
[15]	2024	DL CNN	CSE-CIC-IDS 2018	Acc 98.67%
[16]	2024	DL CNN	CSE-CIC-IDS2018	Acc 97.07%
[18]	2024	DL CNN-LSTM hybrid	CSE-CIC-IDS 2018	Acc 98.53%
[22]	2023	DL CNN-RNN	CSE-CIC-IDS 2018	Acc 98.90%
[20]	2023	DL CNN-LSTM hybrid	CIC-IDS2017	Acc 97.63%
[23]	2023	DL ANN	CSE-CIC-IDS-2018	Acc 99.99%
[17]	2022	DL CNN, LSTM	CSE-CIC-IDS 2018	Acc 98.31%
[21]	2021	DL CNN-LSTM	CIDDS-001, UNSW-NB15	Acc 99.83%
[19]	2021	DL CNN-LSTM hybrid	IoTID20	Acc 98.80%
[24]	2020	DL DNN	CSE-CIC-IDS 2018	Acc 90%

This paper proposed an intrusion detection system based on a hybrid CNN-LSTM deep learning model to detect DDoS attacks, which are the most popular attacks on cloud computing. The proposed model was tested on the CSE-CICIDS2018 dataset to measure accuracy and other parameters like recall and F1-score.

The contributions of the research can be explained as follows:

- Developing a sophisticated CNN-LSTM hybrid intrusion detection model.
- Implement the proposed model on the CSE-CIC-IDS2018 dataset.
- Address common concerns such as data imbalance and feature redundancy to reduce bias and speed up training.
- Compare the performance of the hybrid model with traditional CNN and LSTM models.

This paper is divided into sections: the introduction presents the research background, the problem and previous studies. This is followed by the methodology, which describes the proposed model and evaluation mechanism. The results and discussion analyze the model's performance. The conclusion summarizes the main findings and future recommendations, and finally, the references.

2. MATERIALS AND METHOD

The research methodology used to develop an intrusion detection system in a cloud computing environment using hybrid deep learning model was explained in details.

2.1. Proposed model

The proposed system model that focuses on improving the security of clouds by using deep learning to detect attacks is presented. The model contains CNN and LSTM algorithms on the CES-CICIDS2018 dataset to achieve higher accuracy in detecting DDoS attacks. In Figure 1, all the stages followed in the proposed model are appeared sequentially. First, the data was prepared, and then important features were extracted. The data was then balanced and split into training and testing. Finally, the classifying ability of the model DDoS attacks and normal data was examined using evaluation criteria.

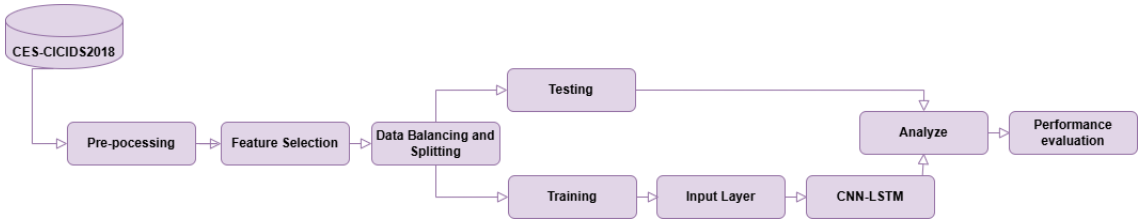


Figure 1. The proposed model

2.2. Data preprocessing

For machine and deep learning techniques, data preprocessing is a crucial step. Preprocessing transforms data into a format that works with any model: dataset cleaning, label encoding, feature selection, normalization, and data splitting. The dataset contains approximately 625,783 rows that include hundreds of normal network traffic, covering many different attack scenarios. Figure 2 shows the operations followed.

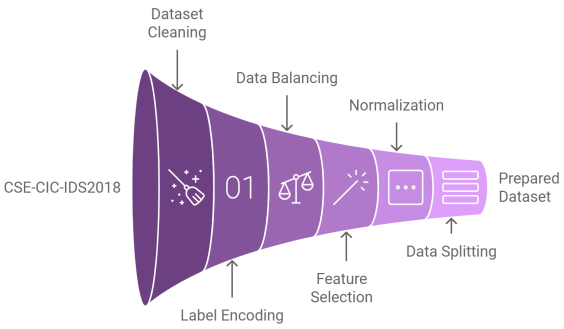


Figure 2. Data preprocessing steps

2.2.1. Data cleaning

It is essential to carefully review the dataset to ensure no null or undefined entries before starting model training. The Pandas library which is a built-in Python component, was utilized for dataset validation in this study. There were cases of incomplete data in the CSE-CIC-IDS2018 dataset, which was used in this investigation. Fixing this, all entries with missing values were removed from the dataset. The process of removing blank values from columns was implemented because they cause problems accessing columns, reduce model stability, and increase error. The missing values were replaced with zero to avoid calculation problems and impact the results.

2.2.2. Label encoding

It converts text data into numeric values that can be understood and handled by deep learning algorithms. It is a step that helps develop the model's performance by replacing the normal class with the numerical value zero and the DDoS class with the value one. After that, we separate the input features from the classification outputs by removing the Label column from the dataset to prepare the features as inputs. This step helps the model understand the data, as deep learning models cannot handle text data directly. It helps speed up computations and data classification and facilitates the process of separating features from labels, facilitating model training.

2.2.3. Normalization

It is a standard process used during the data preparation phase for deep learning models. It is an essential step to ensure that the numerical values of different features are standardized and thus improve the model's performance through training. The standardscaler object is used to calculate the mean and standard deviation. After that, the data is transformed using fit-transform so that the standard equation is applied to each value. The purpose of this process is to make the features comparable and to ensure that the differences between large and small values do not significantly affect the model.

2.2.4. Feature selection

The unique features from the two methods, MIC-features and FCF-features, were merged. MIC-features is a feature set that uses the mutual information technique (measures a nonlinear relationship). FCF-features is a feature set that uses the Pearson correlation coefficient technique (measures a linear relationship). The goal of the merge is to retain features related to intrusion (DDoS attacks). After combining the features, the duplicate columns were removed, resulting in 63 unique features. This ensures the model doesn't have to deal with duplicative data, thus reducing complexity and improving performance. Figure 3 shows the chosen features.

	DstPort	Protocol	FlowDuration	TotFwdPkts	TotBwdPkts	TotLenFwdPkts	TotLenBwdPkts	FwdPktLenMax	FwdPktLenMin	FwdPktLenMean	...
0	80.0	6.0	100000.0	5.0	3.0	405.0	972.0	405.0	0.0	81.000000	...
1	443.0	6.0	100000.0	39.0	95.0	995.0	100000.0	215.0	0.0	25.512821	...
2	445.0	6.0	100000.0	7.0	5.0	364.0	582.0	103.0	0.0	52.000000	...
3	53.0	17.0	38579.0	1.0	1.0	37.0	167.0	37.0	37.0	37.000000	...
4	22.0	6.0	100000.0	22.0	22.0	1928.0	2665.0	640.0	0.0	87.636360	...

5 rows × 63 columns

Figure 3. Feature selection of dataset

2.2.5. Data balancing

The problem of data imbalance was addressed when one of the dataset categories was very large compared to the other category of the same dataset. Data balancing techniques can save training time and storage and avoid under-fitting problems, thus further improving the model performance and reducing bias. A random sampling technique was used for the data balancing procedure. Figure 4 shows the data before and after the data balancing process. This method is simple, fast, and uses little computing resources. It is excellent for the proposed hybrid model, which requires more computing resources.

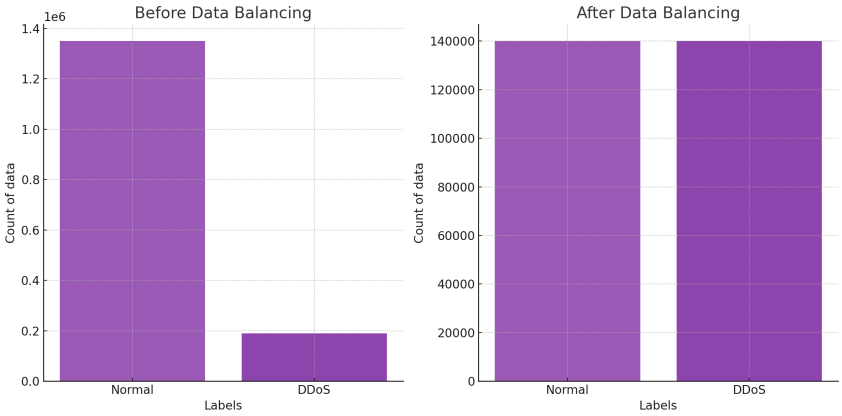


Figure 4. Before and after balance

2.2.6. Dataset splitting

Splitting the data into training and testing sets is a standard preprocessing step in evaluating the performance of deep learning models. We used the train-test-split function to address this issue by splitting the dataset into training and testing sets. This method divided the dataset into 70% training and 30% testing sets. This split provides sufficient data to train the model and understand the categories, prevents overfitting, and aids generalization. A 30% ratio offers enough data to represent each category in the test data.

2.3. Model training

When training a CNN-LSTM hybrid model, the focus is on improving the model's ability to extract essential features via CNN and understand time sequences using LSTM through iterations. Accuracy and loss are also monitored on the training data to fine-tune the model and avoid overfitting, which enhances the hybrid model's ability to predict correctly when new data is used. CNNs automatically extract important patterns and identify relationships in network data, enabling them to distinguish abnormal traffic behavior that could be a breach. LSTMs analyze temporal patterns and track changes in network traffic, enabling them to detect attacks that occur in stages. The CNN-LSTM hybrid model produces a robust model for handling cloud networks and their data. It also leverages the characteristics of each model to increase accuracy in classifying DDoS attacks from natural data. The hybrid model was built using an input layer, followed by two CNN layers, an LSTM layer, two fully connected layers, which acted as a transition between the LSTM and the output layer, and finally, an output layer that classifies the data as normal or DDoS. The ReLU activation function was used in the CNN and fully connected layers because it doesn't require complex computations makes training faster and allows the model to learn from complex data better. The sigmoid activation function was also used in the output layer because the classification is binary, as its output values are between 0 and 1. If the output is close to 1, it is classified as an attack; if it is closer to 0, it is classified as normal data. The binary_crossentropy loss function was used because it is ideal and suitable for binary classification. The Adam optimizer was used to speed up training without needing manual learning rate adjustments, helping the model reach optimal values quickly and making it more stable. The pre-processed training data is entered into the hybrid model via the input layer, then into CNN layers to extract spatial features, then into the LSTM layer to analyze how the features change over time, and then into fully connected layers to transform the features into a classifiable representation, and then into the output layer to decide whether this sample is an attack or normal data.

2.4. Model testing

When testing a CNN-LSTM hybrid model, the focus is on accuracy in predicting outcomes based on new data that has not been trained on and has not been seen before. The model's performance is tested through metrics like accuracy and confusion matrix to ensure its ability to generalize and provide accurate results. The confusion matrix shows the distribution of results between correct and incorrect predictions so that the performance of the model can be accurately evaluated [25]. Accuracy is calculated as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

The equation of precision is:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall is calculated as follows:

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-score is calculated using the equation:

$$F1\text{-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

All the previous equations were taken from [26].

3. RESULTS AND DISCUSSION

Experimental results on the CSE-CIC-IDS2018 dataset demonstrated excellent performance for the hybrid model combining LSTM and CNN. As shown in Table 2, CNN performed well, achieving an accuracy of 99.92%. LSTM achieved an accuracy of 99.83% but a lower classification accuracy than CNN. The hybrid model demonstrated high efficiency in detecting DDoS attacks, achieving an accuracy of 99.88%. These outstanding results indicate the model's ability to improve detection and confirm confidence in its exceptional performance by combining the features of LSTM and CNN compared to implementing them alone.

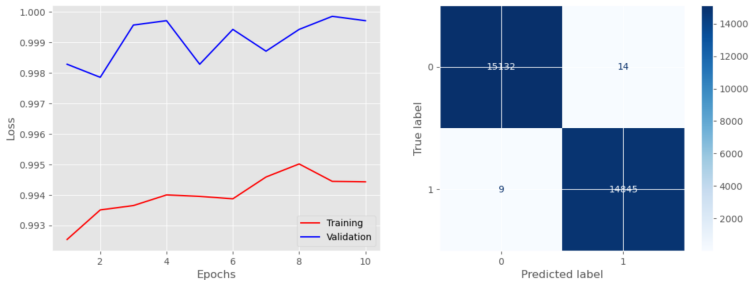
Table 2. Performance of deep learning models

Model	Accuracy	Precision	Recall	F1-score
CNN	0.99923	0.999076	0.999406	0.999241
LSTM	0.99833	0.997623	0.999074	0.998348
Hybrid LSTM-CNN	0.99887	0.998349	0.999405	0.998877

Figure 5 represents the (a) training and validation accuracy curve over several epochs or iterations and (b) confusion matrix of CNN. Training accuracy measures how well a model can classify the data it was trained on and is improved through iteration. Validation accuracy measures how well a model can distinguish data it was trained on to see if it performs well on new data. The model is learning well if the validation accuracy is close to the training accuracy. Suppose the training accuracy is much greater than the validation accuracy. In that case, the model is overfitting, which is when the model does well on training data but does not generalize well to new or test data. This graph shows the performance of the model and its evolution over time. The confusion matrix analyzes the detailed performance of the model to see if the model is having difficulty classifying certain classes. Figure 5(a) shows that the two lines (validation accuracy and training accuracy) are very close together. This means that the model is learning well from the training data, but there may be a slight overfitting due to the divergence of the curves. Figure 5(b) shows the number of correct and incorrect model classifications. Here, 14 normal samples were classified as DDoS, and 9 DDoS samples were classified as normal. The number of errors is small, indicating the model's ability to classify the data correctly.

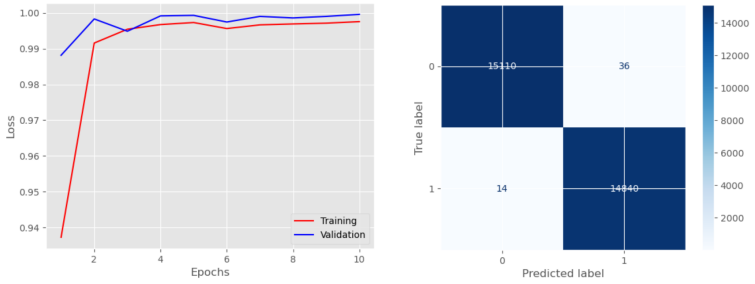
Figure 6 represents the (a) training and validation accuracy curve over several epochs or iterations and (b) confusion matrix of the LSTM. Figure 6(a) shows that the two lines (validation accuracy and training accuracy) are very close together. This means that the model learns well from the training data and can be generalized to new data. Figure 6(b) shows the number of correct and incorrect model classifications. Here, 36 normal samples were classified as DDoS, and 14 DDoS samples were classified as normal. The number of errors is very small, indicating that the model is able to classify the data correctly.

Figure 7 represents the (a) training and validation accuracy curve over several epochs or iterations and (b) confusion matrix of the hybrid model. Figure 7(a) shows that the two lines (validation accuracy and training accuracy) are very close. This indicates that the model is learning well to recognize DDoS attacks and can generalize to new data. Figure 7(b) shows the number of correct and incorrect classifications of the model. Here, 25 normal samples were classified as DDoS, and 9 DDoS samples were classified as normal. The number of errors is minimal, indicating that the model has no difficulty classifying the data, whether DDoS or normal.



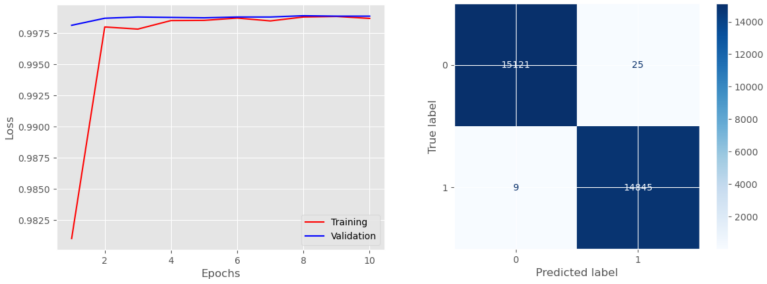
(a) (b)

Figure 5. CNN model: (a) training and validation accuracy curve and (b) confusion matrix



(a) (b)

Figure 6. LSTM model: (a) training and validation accuracy curve and (b) confusion matrix



(a) (b)

Figure 7. LSTM-CNN model: (a) training and validation accuracy curve and (b) confusion matrix

Outcomes reveal that the hybrid CNN-LSTM model enhances attack detection accuracy and reduces false alarms. This model can be used in real cloud networks to enhance their security. The performance of the model agrees with the primary objective of the study, which is to develop an intrusion detection system using a hybrid CNN-LSTM model. The first theory, which was that the combination of CNN and LSTM outperforms the individual algorithms in terms of performance and accuracy, was proven. This research especially contributes to cloud computing security by indicating the effectiveness of hybrid models. Also, it helps researchers to use other hybrid models to enhance security.

3.1. Discussion

This paper displays high accuracy in intrusion detection, especially for DDoS attacks, because of the combination of spatial feature extraction (CNN) and temporal pattern analysis (LSTM). This superiority is demonstrated by improved accuracy, precision, recall, and F1 score, as well as the model's ability to generalize without overfitting, making it effective in cloud computing environments. Comparing the results of the

CNN-LSTM hybrid model with previous studies, it is clear that it achieves accuracy comparable to or superior to similar hybrid models applied to different datasets. As shown in Table 3, the proposed model improves detection performance, making it a more efficient choice for intrusion detection systems in cloud environments.

Table 3. Comparison with similar studies

Reference	Algorithm	Dataset	Accuracy
[18]	CNN-LSTM hybrid	CSE-CIC-IDS2018	98.53%
[20]	CNN-LSTM hybrid	CSE-CIC-IDS2017	97.63%
[21]	CNN-LSTM hybrid	CIDDS-001 ,UNSW-NB15	99.83%
[19]	CNN-LSTM hybrid	IoTID20	98.80%
Our model	CNN-LSTM hybrid	CSE-CIC-IDS2018	99.88%

The CNN-LSTM hybrid model has several strengths, such as high accuracy in detecting sophisticated attacks and achieving significant performance improvements compared to traditional models. It also demonstrates strong generalization capabilities, making it reliable in various environments. On the other hand, the model suffers from drawbacks that may affect its use, such as consuming significant computing resources and lengthy training times due to the complex combination of CNN and LSTM. The study aims to improve intrusion detection systems using a hybrid CNN-LSTM model to enhance detection accuracy and reduce false alarms, especially in the face of DDoS attacks. The research is essential for enhancing cloud network security and opening new horizons for applying hybrid models in cybersecurity.

3.2. Limitations

- The model can be applied to various cloud environments but can be retrained on different datasets to be more adaptive to changing environments.
- The model demonstrates strong performance in detecting traditional and time-lapse attacks, while it could be further optimized to address advanced threats such as zero-day and APTs.

4. CONCLUSION

Cloud environments are among the most essential services that make users' lives easier and store their data. Securing the cloud is extremely important because it protects the services and their users from cyber threats. This research is about detecting attacks on cloud computing networks using a CNN-LSTM hybrid deep learning model. This hybrid approach is designed to detect DDoS attacks. It achieved 99.88% detection accuracy and reduced false alarms, which promotes the efficiency and effectiveness of intrusion detection systems in cloud computing networks. It is concluded that the hybrid model achieves a unique balance between the capabilities of CNN to bring out spatial features from the CES-CICIDS2018 dataset and the capabilities of LSTM to track temporal patterns of data. These advantages enhance cloud infrastructure security and reduce the resources required for threat detection. Although the hybrid model requires significant computing resources, it provides security benefits worth the investment in conjunction with the increase in cyber threats to cloud services. In future research, it is possible to test the hybrid model on different environments, such as smart grids and other IoT environments. Develop performance optimization techniques to reduce computational complexity while maintaining high accuracy. In addition to detecting unknown attacks using transformer models. It is possible to use newer and more diverse datasets.

ACKNOWLEDGEMENTS

We dedicate this work to our beloved parents, whose unwavering support and encouragement made this journey possible. A heartfelt thank you to my dear friend and partner, whose dedication and cooperation were key to achieving this milestone. Together, we accomplished this success.

FUNDING INFORMATION

Authors state no funding involved.

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] W. Ahmad, A. Rasool, A. R. Javed, T. Baker, and Z. Jalil, "Cyber security in IoT-based cloud computing: a comprehensive survey," *Electronics (Switzerland)*, vol. 11, no. 1, p. 16, Dec. 2021, doi: 10.3390/electronics11010016.

[2] N. M. Abdulkareem, S. R. M. Zeebaree, M. A. M. Sadeeq, D. M. Ahmed, A. S. Sami, and R. R. Zebari, "IoT and cloud computing issues, challenges and opportunities: a review," *Qubahan Academic Journal*, vol. 1, no. 2, pp. 1–7, Mar. 2021, doi: 10.48161/qaj.v1n2a36.

[3] M. Faheem, U. Akram, I. Khan, S. Naqeeb, A. Shahzad, and A. Ullah, "Cloud computing environment and security challenges: a review," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 10, 2017, doi: 10.14569/ijacsa.2017.081025.

[4] J. Surbiryala and C. Rong, "Cloud computing: history and overview," in *2019 IEEE Cloud Summit*, Aug. 2019, pp. 1–7, doi: 10.1109/CloudSummit47114.2019.00007.

[5] Z. Liu, B. Xu, B. Cheng, X. Hu, and M. Darbandi, "Intrusion detection systems in the cloud computing: a comprehensive and deep literature review," *Concurrency and Computation: Practice and Experience*, vol. 34, no. 4, p. e6646, Feb. 2022, doi: 10.1002/cpe.6646.

[6] M. N. Birje, P. S. Challagidad, R. H. Goudar, and M. T. Tapale, "Cloud computing review: concepts, technology, challenges and security," *International Journal of Cloud Computing*, vol. 6, no. 1, pp. 32–57, 2017, doi: 10.1504/IJCC.2017.083905.

[7] S. Rani, P. Bhambri, A. Kataria, A. Khang, and A. K. Sivaraman, *Big data, cloud computing and IoT: tools and applications*. Chapman and Hall/CRC, 2023.

[8] Maniah, E. Abdurachman, F. L. Gaol, and B. Soewito, "Survey on threats and risks in the cloud computing environment," *Procedia Computer Science*, vol. 161, pp. 1325–1332, 2019, doi: 10.1016/j.procs.2019.11.248.

[9] Y. I. Alzoubi, A. Mishra, and A. E. Topcu, "Research trends in deep learning and machine learning for cloud computing security," *Artificial Intelligence Review*, vol. 57, no. 5, p. 132, May 2024, doi: 10.1007/s10462-024-10776-5.

[10] S. El Kafhali, I. El Mir, and M. Hanini, "Security threats, defense mechanisms, challenges, and future directions in cloud computing," *Archives of Computational Methods in Engineering*, vol. 29, no. 1, pp. 223–246, Apr. 2022, doi: 10.1007/s11831-021-09573-y.

[11] A. Munshi, N. A. Alqarni, and N. Abdullah Almalki, "DDOS attack on IoT devices," in *2020 3rd International Conference on Computer Applications & Information Security (ICCAIS)*, Mar. 2020, pp. 1–5, doi: 10.1109/ICCAIS48893.2020.9096818.

[12] N. Mishra, R. K. Singh, and S. K. Yadav, "Detection of DDoS vulnerability in cloud computing using the perplexed bayes classifier," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–13, Jul. 2022, doi: 10.1155/2022/9151847.

[13] V. Chang *et al.*, "A survey on intrusion detection systems for fog and cloud computing," *Future Internet*, vol. 14, no. 3, p. 89, Mar. 2022, doi: 10.3390/fi14030089.

[14] D. Mohamed and O. Ismael, "Enhancement of an IoT hybrid intrusion detection system based on fog-to-cloud computing," *Journal of Cloud Computing*, vol. 12, no. 1, Mar. 2023, doi: 10.1186/s13677-023-00420-y.

[15] W. H. Aljuaid and S. S. Alshamrani, "A deep learning approach for intrusion detection systems in cloud computing environments," *Applied Sciences (Switzerland)*, vol. 14, no. 13, p. 5381, Jun. 2024, doi: 10.3390/app14135381.

[16] A. D. Vibhute and V. Nakum, "Deep learning-based network anomaly detection and classification in an imbalanced cloud environment," *Procedia Computer Science*, vol. 232, pp. 1636–1645, 2024, doi: 10.1016/j.procs.2024.01.161.

[17] A. A. Hagar and B. W. Gawali, "Apache Spark and deep learning models for high-performance network intrusion detection using CSE-CIC-IDS2018," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–11, Aug. 2022, doi: 10.1155/2022/3131153.

[18] J. A. Alzubi, O. A. Alzubi, I. Qiqieh, and A. Singh, "A blended deep learning intrusion detection framework for consumable edge-centric IoMT industry," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 2049–2057, Feb. 2024, doi: 10.1109/TCE.2024.3350231.

[19] H. Alkahtani and T. H. H. Aldhyani, "Intrusion detection system to advance internet of things infrastructure-based deep learning algorithms," *Complexity*, vol. 2021, no. 1, p. 5579851, Jan. 2021, doi: 10.1155/2021/5579851.

[20] N. Faruqui *et al.*, "SafetyMed: a novel IoMT intrusion detection system using CNN-LSTM hybridization," *Electronics (Switzerland)*, vol. 12, no. 17, p. 3541, Aug. 2023, doi: 10.3390/electronics12173541.

[21] S. Al and M. Dener, "STL-HDL: a new hybrid network intrusion detection system for imbalanced dataset on big data environment," *Computers and Security*, vol. 110, p. 102435, Nov. 2021, doi: 10.1016/j.cose.2021.102435.

[22] E. U. H. Qazi, M. H. Faheem, and T. Zia, "HDLNIDS: hybrid deep-learning-based network intrusion detection system," *Applied Sciences (Switzerland)*, vol. 13, no. 8, p. 4921, Apr. 2023, doi: 10.3390/app13084921.

[23] M. Khan and M. Haroon, "Artificial neural network-based intrusion detection in cloud computing using CSE-CIC-IDS2018 datasets," in *2023 3rd Asian Conference on Innovation in Technology (ASIANCON)*, Aug. 2023, pp. 1–4, doi: 10.1109/ASIANCON58793.2023.10269948.

[24] R. I. Farhan, A. T. Maolood, and N. F. Hassan, "Performance analysis of flow-based attacks detection on CSE-CIC-IDS2018 dataset using deep learning," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 20, no. 3, pp. 1413–1418, Dec. 2020, doi: 10.11591/ijeecs.v20.i3.pp1413-1418.

[25] Y. Xin *et al.*, "Machine learning and deep learning methods for cybersecurity," *IEEE Access*, vol. 6, pp. 35365–35381, 2018, doi: 10.1109/ACCESS.2018.2836950.

[26] M. Vakili, M. Ghamsari, and M. Rezaei, "Performance analysis and comparison of machine and deep learning algorithms for IoT data classification," *arXiv preprint arXiv:2001.09636*, Jan. 2020, [Online]. Available: http://arxiv.org/abs/2001.09636.

BIOGRAPHIES OF AUTHORS

Maha Mohammad Alshehri received her Bachelor's degree in information technology in 2021 with a GPA of 3.91/4. She is currently pursuing a Master's degree in cybersecurity. Her interests include IT security, network protection, and problem-solving in digital environments. She has participated in university volunteering activities, contributing to improving learning facilities. Her skills include programming in Python, teamwork, time management, and analytical thinking. She can be contacted at email: mahaalshehri11@outlook.com

Shoog Abdullah Alshehri received her Bachelor's degree in computer sciences from King Khalid University in 2022 with honors (GPA: 4.73/5). She is currently pursuing a Master's degree in cybersecurity. She has experience as a System Administrator at Al-Ameen Hospital, managing IT operations, and as an Instructor at Bisha University, teaching information systems. Her research interests include cybersecurity, system administration, and IT education. She has completed multiple certifications in digital marketing, cloud computing, artificial intelligence, and cybersecurity risk management. Her technical skills include programming in Python, ASP, and SQL. She can be contacted at email: as.alshehri@seu.edu.sa.

Samah Hazzaa Alajmani received the B.Sc. degree in computer science from King Abdulaziz University, Jeddah, Saudi Arabia, in 2004, and the Ph.D. degree in computer science from the same university in 2019. She earned her M.Sc. degree in information technology from the Queensland University of Technology, Brisbane, Australia. She is currently an Assistant Professor at Taif University, Taif, Saudi Arabia. Her research interests include cybersecurity, artificial intelligence, IoT, deep learning, and machine learning. She can be contacted at email: s.ajmani@tu.edu.sa.