Salud, Ciencia y Tecnología - Serie de Conferencias. 2024; 3:968
doi: 10.56294/sctconf2024968

Category: STEM (Science, Technology, Engineering and Mathematics)

ORIGINAL

Fine-Tuning CNN-BiGRU for Intrusion Detection with SMOTE Optimization Using Optuna

Ajuste Fino de CNN-BiGRU para Detección de Intrusos con Optimización SMOTE Utilizando Optuna

Asmaa Benchama[1] ✉, Khalid Zebbara[1] ✉

[1]IMISR Laboratory, Faculty of Science AM, Ibn Zohr University. Agadir, Morocco.

Cite as: Benchama A, Zebbara K. Fine-Tuning CNN-BiGRU for Intrusion Detection with SMOTE Optimization Using Optuna. Salud, Ciencia y Tecnología - Serie de Conferencias. 2024; 3:968. https://doi.org/10.56294/sctconf2024968

Submitted: 16-02-2024 **Revised:** 05-05-2024 **Accepted:** 28-06-2024 **Published:** 29-06-2024

Editor: Dr. William Castillo-González

ABSTRACT

Network security faces a significant challenge in developing effective models for intrusion detection within network systems. Network Intrusion Detection Systems (NIDS) are vital for protecting network traffic and preempting potential attacks by identifying signatures and rule violations.
This research aims to enhance intrusion detection using Deep learning techniques, particularly by employing the NSLKDD dataset to train and evaluate a hybrid CNN-BiGRU algorithm. Additionally, we utilize the Synthetic Minority Over-sampling Technique (SMOTE) to address imbalanced data and Optuna for fine-tuning the algorithm's parameters specific to NIDS requirements.
The hybrid CNN-BiGRU algorithm is trained and evaluated on the NSLKDD dataset, incorporating SMOTE to tackle imbalanced data issues. Optuna is utilized to optimize the algorithm's parameters for improved performance in intrusion detection.
Experimental results demonstrate that our approach surpasses classical intrusion detection models. Achieving an accuracy rate of 98,83 % on NSLKDD, the proposed model excels in identifying minority attacks while maintaining a low false positive rate.
The findings affirm the efficacy of our proposed approach in network intrusion detection, showcasing its ability to effectively discern patterns in network traffic and outperform traditional models.

Keywords: CNN-BIGRU; NSLKDD; NIDS; SMOTE; Hyper-Parameters Optimizer.

RESUMEN

La seguridad de redes enfrenta un desafío significativo en el desarrollo de modelos efectivos para la detección de intrusiones dentro de los sistemas de red. Los Sistemas de Detección de Intrusos en Red (NIDS) son vitales para proteger el tráfico de red y prevenir posibles ataques mediante la identificación de firmas y violaciones de reglas.
Esta investigación tiene como objetivo mejorar la detección de intrusiones utilizando técnicas de aprendizaje profundo, particularmente mediante el empleo del conjunto de datos NSLKDD para entrenar y evaluar un algoritmo híbrido CNN-BiGRU. Además, utilizamos la Técnica de Sobremuestreo Sintético de Minorías (SMOTE) para abordar datos desequilibrados y Optuna para ajustar finamente los parámetros del algoritmo específicos para los requisitos de NIDS.
El algoritmo híbrido CNN-BiGRU se entrena y evalúa en el conjunto de datos NSLKDD, incorporando SMOTE para abordar problemas de datos desequilibrados. Optuna se utiliza para optimizar los parámetros del algoritmo para mejorar el rendimiento en la detección de intrusiones.

Los resultados experimentales demuestran que nuestro enfoque supera a los modelos clásicos de detección de intrusiones. Alcanzando una notable tasa de precisión del 98,83 % en NSLKDD, el modelo propuesto sobresale en la identificación de ataques minoritarios mientras mantiene una baja tasa de falsos positivos. Los hallazgos confirman la eficacia de nuestro enfoque propuesto en la detección de intrusiones en redes, mostrando su capacidad para discernir efectivamente patrones en el tráfico de red y superar a los modelos tradicionales.

Palabras clave: CNN-BIGRU; NSLKDD; NIDS; SMOTE; Optimizador de Hiperparámetros.

INTRODUCTION

In the ever-evolving landscape of today's interconnected world, the internet has emerged as the primary conduit linking communities and individuals worldwide, profoundly influencing nearly every aspect of human existence. However, the rapid proliferation of internet services and the exponential growth in information traffic have catalyzed heightened concerns regarding network security in recent years. The expanding array of intrusion techniques underscores the critical importance of robust network defense mechanisms.

Numerous conventional strategies have been proposed to safeguard network integrity, encompassing methodologies such as firewalls, digital signatures, security gateways, and vulnerability scanning. Despite their widespread adoption, these traditional defense mechanisms often exhibit limitations in effectively identifying and thwarting network attacks. Many of these approaches maintain a passive stance, lacking the agility and responsiveness required to counter emerging threats effectively.

In response to the shortcomings of traditional security paradigms, Intrusion Detection Systems (IDS) have emerged as a proactive means of detecting and mitigating network threats. Representing a significant paradigm shift in network security, IDS systems offer real-time detection and response capabilities to suspicious network activities. Unlike conventional defense strategies, IDS solutions possess the flexibility to adapt to evolving threat landscapes through ongoing learning and adaptation.

Furthermore, contemporary IDS systems leverage advanced machine learning[1,2] and deep learning[3] techniques to enhance their efficacy. By harnessing large-scale datasets and innovative methodologies, these systems are capable of identifying both known and novel forms of network intrusions. Techniques like data oversampling to address imbalances and automated hyperparameter tuning further optimize IDS performance, ensuring robust network security in the face of evolving cyber threats.

This article explores our method, which harnesses a hybrid CNN-BiGRU model fortified with Synthetic Minority Over-sampling Technique (SMOTE) oversampling[4] and fine-tuned through Optuna optimization.[5] By leveraging the capabilities of deep learning and Deep learning techniques, our approach aims to tackle the inherent challenges posed by contemporary cyber threats, including the detection of both known and emerging intrusion patterns. Additionally, we elucidate how the integration of SMOTE enables effective handling of imbalanced data, while Optuna facilitates the automated refinement of model parameters tailored to the specific characteristics of network security.

Related works

Convolutional Neural Networks (CNN)[6,7] have revolutionized computer vision applications, particularly excelling in tasks such as image and video analysis, classification, face recognition, target recognition, and image processing. These neural networks employ convolutional layers to automatically learn hierarchical representations from data, making them highly effective in capturing complex visual patterns.

On the other hand, Gated Recurrent Units (GRU)[8] belonging to the family of recurrent neural networks (RNN)[3,9] specialize in handling sequential data processing. Unlike traditional RNN, GRU are designed to capture long-range dependencies more efficiently, making them well-suited for applications involving natural language understanding, time series analysis, and other scenarios where understanding temporal patterns is crucial.

In the context of intrusion detection, researchers have increasingly explored the adaptation of CNN and GRU to address security challenges. CNN, known for their ability to capture spatial features, are applied to analyze network traffic patterns. GRUs, with their proficiency in handling temporal dependencies, contribute to the detection of anomalies or intrusions over time. Combining these neural network architectures has led to innovative approaches that aim to improve the accuracy and efficiency of IDS.[10]

Various studies have proposed advanced intrusion detection algorithms leveraging CNN and GRU. These algorithms often incorporate sophisticated architectural elements such as Residual networks, inception modules, and weight-dropped Long Short-Term Memory network (LSTM) to enhance their capabilities. Additionally, hybrid approaches combining different neural network architectures have emerged, showcasing impressive accuracy and efficiency in detecting network intrusions.

https://doi.org/10.56294/sctconf2024968

Researchers have explored the potential of CNN, RNN, LSTM, and GRU to address the evolving nature of cyber threats, emphasizing the need for IDS capable of handling diverse and complex attack scenarios.

Li Y et al.[7] proposed an intrusion detection algorithm that integrates a deep Convolutional Neural Network (CNN) as its fundamental framework. Notably, their approach incorporates advanced architectural elements such as the Residual network and the inception module. These components expedite the convergence of the model, thereby enhancing the accuracy of intrusion detection.

Hassan et al.[11] presented an intrusion detection system that utilizes a combination of a CNN and a weight-dropped Long Short-Term Memory network (LSTM). This fusion of neural network architectures is designed to enhance the system's performance by enabling effective feature extraction and addressing long-term dependencies in the data.

ElSayed et al.[12] introduced a hybrid Deep Learning method based on CNN, achieving an impressive accuracy rate. Their approach outperforms single Deep Learning models across all assessment criteria, demonstrating high performance without compromising efficiency.

Mauro et al.[13] emphasized the limitations of existing datasets and introduced a neural-based approach for Network Intrusion, highlighting WiSARD as a solution designed to handle multivariable datasets.

Hao et al.[14] employed Bi-directional Long Short-Term Memory networks (Bi-LSTM) for analyzing HTTP requests, demonstrating commendable performance on the HTTP DATASET CSIC 2010.

Kishor et al.[9] investigated the RNN-LSTM method for intrusion detection, achieving promising performance results using multiple datasets. Their evaluation contributes valuable insights to the field of intrusion detection research.

METHOD

The methodology and materials section provides a comprehensive framework for understanding the design, implementation, and evaluation of our CNN-BiGRU model enhanced with an attention mechanism, SMOTE, and optimized using Optuna for hyperparameter tuning. This section lays the foundation for the subsequent discussion of experimental results and findings in the article.

Our methodology consists of a comprehensive overview of the architectural design of the CNN-BiGRU model with an attention mechanism. We elucidate the rationale behind integrating CNN, Bidirectional Gated Recurrent Units (BiGRU), and the attention mechanism, highlighting their complementary strengths in capturing spatial and temporal dependencies, and focusing on relevant features in the input data. The incorporation of the attention mechanism enhances the model's ability to selectively attend to important regions of input data, thereby improving its discriminative power.

Additionally, we delve into the hyperparameter tuning process using Optuna. We explore the hyperparameter space and optimize the model's parameters for enhanced performance. We discuss the objective function, search strategies, and evaluation criteria utilized in the hyperparameter tuning process, aiming to achieve the optimal configuration for our CNN-BiGRU model with the attention mechanism.

Architectural Overview of the Hybrid CNN-BiGRU Model

In contrast to conventional models lacking the integration of the SMOTE technique, attention mechanism[15,16] and Optuna optimization, the CNN-BiGRU model capitalizes on these cutting-edge advancements to significantly augment its performance. By harnessing these techniques, the CNN-BiGRU model is engineered to excel in various facets:

- Local Feature Extraction: proficiently extracts nuanced features from traffic data.
- Spatial Downsampling: reduces spatial dimensionality of the data for streamlined processing.
- Activation Normalization: ensures consistent activation levels for enhanced model stability.
- Bidirectional Feature Capture: captures features bidirectionally for a comprehensive understanding.
- Regularization: implements multiple regularization techniques to combat overfitting.
- Output Reshaping: reshapes the output for improved interpretability.
- Temporal Downsampling: downsamples temporal dimensions for efficient processing.
- Attention Mechanism: identifies and accentuates critical features automatically, allowing the model to prioritize relevant information.
- Feature Fusion for Classification: integrates features intelligently to facilitate accurate classification.
- Logit Conversion to Probabilities: converts logits to class probabilities for refined output interpretation.

The incorporation of the SMOTE technique proves instrumental in handling imbalanced data, a pervasive challenge in intrusion detection scenarios. Moreover, the attention layer assumes a pivotal role in discerning and highlighting pivotal features, enabling the model to concentrate on pertinent information throughout the

https://doi.org/10.56294/sctconf2024968

learning phase. Additionally, the integration of Optuna optimization fine-tunes hyperparameters, optimizing the model's architecture for superior performance.

Table 1. Attributes of the CNN-BiGRU Model

Characteristic	Specification
Model Architecture	Hybrid architecture comprising and BiGRU layers.
Feature Extraction	Utilizes CNN layers for extracting local features from traffic data.
Temporal Processing	BiGRU layers capture bidirectional temporal dependencies in the data.
Attention Mechanism	Attention layer highlights important features, allowing the model to focus on relevant information during training.
Regularization	Implements multiple regularization techniques to prevent overfitting.
Hyperparameter Optimization	Optuna optimization fine-tunes hyperparameters to optimize the model's architecture for improved performance.
Imbalanced Data Handling	Incorporates SMOTE to handle imbalanced data, enhancing model robustness in intrusion detection.
Activation Normalization	Normalizes activations to maintain consistent levels across the network, promoting stability during training.
Output Interpretation	Converts logits to class probabilities for refined output interpretation.
Simulation Environment	Python Version: 3,8 - TensorFlow Version: 2,5- Keras Version: 2,5 - CUDA Toolkit Version: 11,2 - cuDNN Version: 8,2

By amalgamating these advanced methodologies, the CNN-BiGRU model adopts a holistic learning approach, effectively encompassing both spatial and temporal dimensions. Consequently, this holistic integration translates into heightened accuracy in intrusion detection classification when juxtaposed against conventional methods. The delineated specifications and comparisons underscore the model's efficacy in mitigating various challenges associated with traditional intrusion detection methodologies.

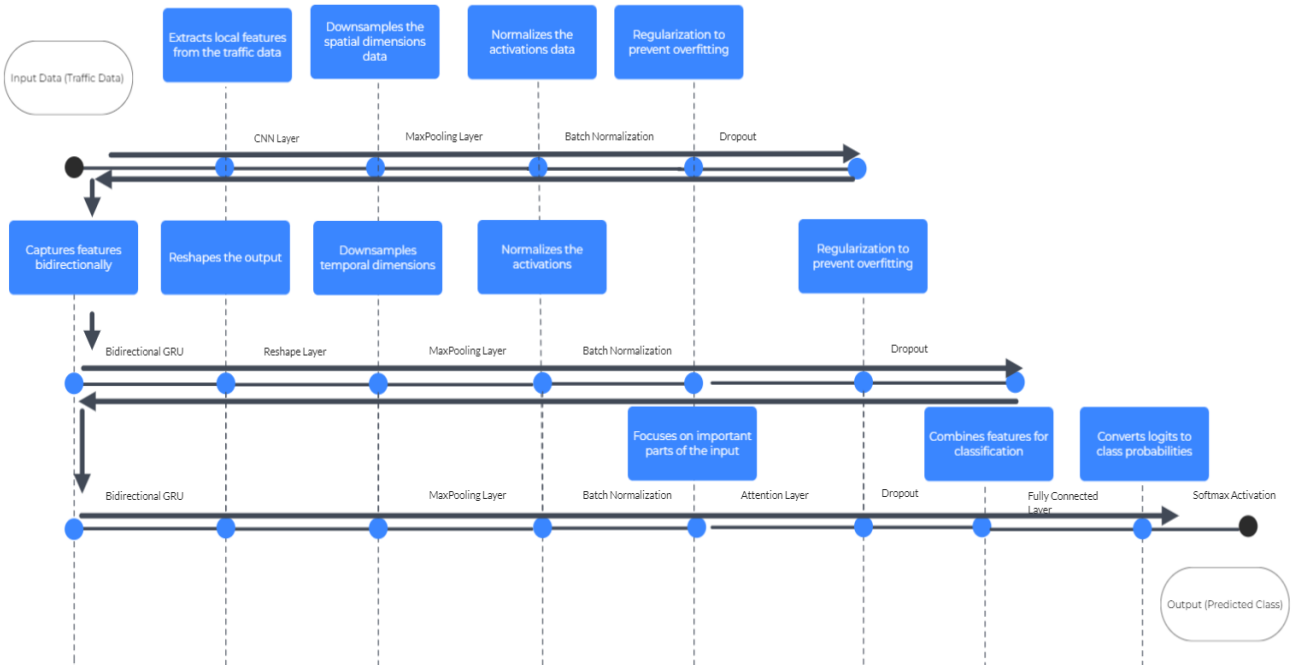

Figure 1. Overview of the Architecture: Hybrid CNN-BiGRU Model

Attention mechanism

In our model architecture, we integrate an Attention layer[17] to discern temporal patterns within the traffic data. This layer is pivotal as it enables the model to dynamically allocate attention across different segments of the input sequence. By doing so, the model can prioritize and focus on pertinent information at various time steps. This nuanced attention mechanism greatly enhances the model's capability to grasp the intricate temporal patterns inherent in network traffic data, fostering a more profound understanding of its dynamics and behavior. This architectural feature significantly enriches our model's capacity to discern and interpret temporal intricacies, ultimately bolstering its efficacy in intrusion detection and network analysis tasks.

https://doi.org/10.56294/sctconf2024968

Optuna: Hyperparameter Tuning

We leverage Optuna[11] to streamline the optimization of hyperparameters throughout the training phase. Optuna efficiently explores the hyperparameter space, systematically seeking the optimal combination that maximizes our predefined performance metric. By automating this process, Optuna alleviates the burdensome task of manual hyperparameter tuning. This automation not only saves time but also holds the promise of enhancing our model's performance. Specifically, Optuna's objective function is designed to iteratively evaluate different hyperparameter configurations, aiming to maximize our model's efficacy in detecting intrusions accurately while minimizing false positives. Through this automated optimization process, our model becomes finely tuned to the unique intricacies of intrusion detection tasks, ultimately bolstering its ability to discern malicious activities effectively.

SMOTE: addressing Imbalanced Data issues

During data preprocessing, we employ the SMOTE technique[18] to tackle the inherent challenge of imbalanced data. SMOTE dynamically generates synthetic samples for the minority class, a crucial step particularly in the realm of intrusion detection. By balancing the distribution between normal and intrusive instances, SMOTE ensures that the model is adequately trained on representative samples of both classes. This augmentation significantly improves the model's capacity to accurately detect intrusions, thereby enhancing the overall effectiveness of our intrusion detection system. SMOTE algorithm involves generating synthetic samples for the minority class by interpolating between existing minority class instances.

Mathematical representation of the SMOTE algorithm:

- Let *Xmin* be the set of minority class instances.
- Let K be the number of nearest neighbors to consider.
- Let K be the desired number of synthetic samples to generate.

For each minority class instance *xi* in *xmin*:

- Find its K nearest neighbors.
- Randomly select one of the neighbors *xnn*.

For each selected neighbor xnn:

- Compute the difference vector $\Delta = xnn - x\,i$.
    1. Generate N synthetic samples.

For each sample:

- Select a random value λ in the range [0,1].
- Compute the synthetic sample xsynthetic as $x\,i + \lambda \times \Delta$.

This process creates N synthetic samples by linearly interpolating between each minority class instance and one of its nearest neighbors. By introducing these synthetic samples, the imbalance between the minority and majority classes is mitigated, leading to a more balanced dataset for training the machine learning model.

While this representation captures the essence of the SMOTE algorithm, variations and additional parameters may exist in different implementations to further customize the sampling process.

In this research, we evaluate our proposed intrusion detection model using the NSL-KDD dataset for both training and testing purposes. Our primary objective is to comprehensively assess the model's performance across various attack types, ensuring a thorough evaluation of its effectiveness. The selection of the NSL-KDD dataset is crucial due to its richness in diverse attack scenarios, making it an ideal candidate for evaluating IDS.

NSL-KDD Dataset

For our experimentations with the CNN-BiGRU model, we utilized the NSL-KDD dataset. The NSL-KDD dataset[19] available at https://www.unb.ca/cic/datasets/nsl.html , is widely acknowledged in the field of intrusion detection. It represents an enhanced version of the original KDDCup 99 dataset[20] addressing several limitations present in its predecessor. Notably, the NSL-KDD dataset incorporates a broad spectrum of network traffic scenarios and attack types, making it a valuable asset for both training and testing IDS models. Its development aimed to rectify issues observed in the KDDCUP99 dataset, particularly addressing feature redundancy, thus rendering it more suitable for the preparation and evaluation of IDS.

Characterized by a substantial volume of data entries, the NSL-KDD dataset offers a foundation for training and evaluating intrusion detection models. It encompasses various legitimate entries and is classified into four main attack classes: DoS (Denial of Service), R2L (Root to Local attacks), U2R (User to Root attacks), and probe attacks. With a total of 41 features, each entry in the dataset provides a comprehensive representation of network traffic data, annotated with its corresponding attack class.

The table 2 shows the distribution of attack classes along with the corresponding number of instances and

https://doi.org/10.56294/sctconf2024968

their respective percentages within the dataset. Due to the presence of minority classes within the dataset, we employed the SMOTE technique to address class imbalance and enhance the effectiveness of our model.

Table 2. Distribution of Attack Classes

Attack Class	Number of Instances	(%)
Normal	67343	53,46
DoS	45927	36,52
Probe	11656	9,27
R2L	995	0,79
U2R	52	0,04

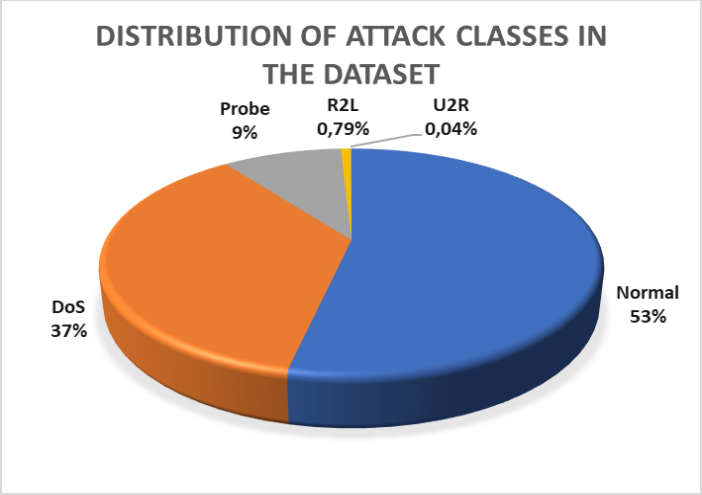

Figure 2. Distribution of the classes in NSL-KDD Dataset

The schema presented in figure 2 displays the percentage of attacks per class alongside the corresponding number of instances.

RESULTS AND DISCUSSION

The evaluation metrics utilized for assessing the efficacy of our network security intrusion detection model encompass four key indicators: precision, accuracy, recall, and F1-score. These metrics serve as essential benchmarks, providing insights into the model's performance across various dimensions. They are derived from the confusion matrix, a tabular representation that categorizes the classification outcomes of the model into four distinct scenarios: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

In our evaluation process, we leverage the NSLKDD dataset, which comprises instances classified into five distinct classes: 'DoS' (Denial of Service), 'Normal', 'Probe', 'R2L' (Root to Local attacks), and 'U2R' (User to Root attacks). Opting for a multi-class classification approach allows us to comprehensively examine the model's performance, enabling us to assess its ability to differentiate between various attack categories and normal instances effectively.

Multi-class classification facilitates a granular analysis of the model's precision, recall, and overall classification accuracy across the diverse array of classes present in the dataset. By scrutinizing the model's performance on each class individually, we gain valuable insights into its strengths and weaknesses in detecting different types of network intrusions.

Best trial

The optimization process entails defining an objective function aimed at either minimizing or maximizing a specific metric, and Optuna systematically searches for the hyperparameters that achieve this objective. In our case, the objective function $f(x)$ encompasses the parameters: convolutional kernel size, convolutional dropout rate, GRU dropout rate, attention dropout rate, and learning rate.

Each trial conducted in Optuna represents a unique set of hyperparameters sampled from the defined search space. In our study, we executed a total of 15 trials to identify the optimal configuration. However, recognizing the potential for further refinement, we intend to explore a larger number of trials in our future endeavors. By expanding the scope of our trial exploration, we aim to enhance the robustness and effectiveness of our model

https://doi.org/10.56294/sctconf2024968

optimization process, thereby maximizing its performance in intrusion detection tasks.

Best trial: Value: 0,028093338012695312.

Table 3. Optuna best trial parameters

Parameter	Value
conv_kernel_size	24
conv_dropout	0,45328375223588824
gru_dropout	0,026449575992992313
attention_dropout	0,08715283056018074
learning_rate	0,0018285312453719437

Table 3 presents the parameter values obtained from the best trial.

conv_kernel_size: with a value of 24, this parameter specifies the size of the convolutional kernel used in the model. A larger kernel size allows the model to capture more complex patterns in the input data.

conv_dropout: the dropout rate for the convolutional layers is determined to be approximately 0,45. Dropout is a regularization technique used to prevent overfitting by randomly dropping a fraction of input units during training.

gru_dropout: the dropout rate for the GRU (Gated Recurrent Unit) layer is found to be around 0,03. Similar to convolutional dropout, this parameter controls the rate at which units in the GRU layer are randomly dropped during training to prevent overfitting.

attention_dropout: this parameter, with a value of approximately 0,09, specifies the dropout rate for the attention mechanism. Dropout in the attention layer helps improve the generalization capability of the model by preventing it from focusing too heavily on specific parts of the input.

learning_rate: the learning rate, set to approximately 0,0018, determines the step size at which the model parameters are updated during training. It plays a crucial role in controlling the convergence and stability of the optimization process.

The values obtained for these parameters represent an optimal configuration that minimizes the objective function, leading to improved performance of the intrusion detection model.

Multiclass Classification on the NSLKDD Dataset

In this section, we present the results of our multiclass classification approach using the CNN-BiGRU model on the NSLKDD dataset. Leveraging advanced techniques such as CNN for local feature extraction and BiGRU for capturing temporal dependencies, our model aims to accurately classify instances into one of the five classes present in the NSLKDD dataset: ‘DoS’, ‘Normal’, ‘Probe’, ‘R2L’, and ‘U2R’. The evaluation of our model’s performance is based on key metrics including precision, recall, and F1-score, which provide insights into its ability to effectively distinguish between different attack categories and normal network traffic. Additionally, we employ optimization techniques such as Optuna to fine-tune hyperparameters and enhance the model’s overall performance. The results obtained from this comprehensive evaluation shed light on the efficacy of our CNN-BiGRU approach.

Table 4. Evaluation metrics average of the best trial

Metric	Rate
Precision	99,27 %
Recall	98,83 %
F1-score	98,98 %

As shown in table 4, the evaluation metrics for the CNN-BiGRU model on the NSLKDD dataset exhibit strong performance across various dimensions. Precision, measuring the accuracy of positive predictions made by the model, attains a score of 99,27 %. This implies that out of all instances classified as a particular class by the model, 99,27 % were correctly classified. Such high precision underscores the model’s ability to minimize false positives, instilling confidence in its classification outcomes.

Additionally, recall, also known as sensitivity, achieves a score of 98,83 %. This indicates that the model correctly identified 98,83 % of all instances belonging to a particular class. A high recall score suggests that the model effectively captures most instances of the class in the dataset, mitigating the risk of false negatives.

Furthermore, the F1-score, which represents the harmonic mean of precision and recall, stands at an 98,98

%. This high F1-score indicates that the model achieves both high precision and high recall simultaneously, striking a good balance between minimizing false positives and false negatives. Such balanced performance makes the CNN-BiGRU model well-suited for multiclass classification on the NSLKDD dataset.

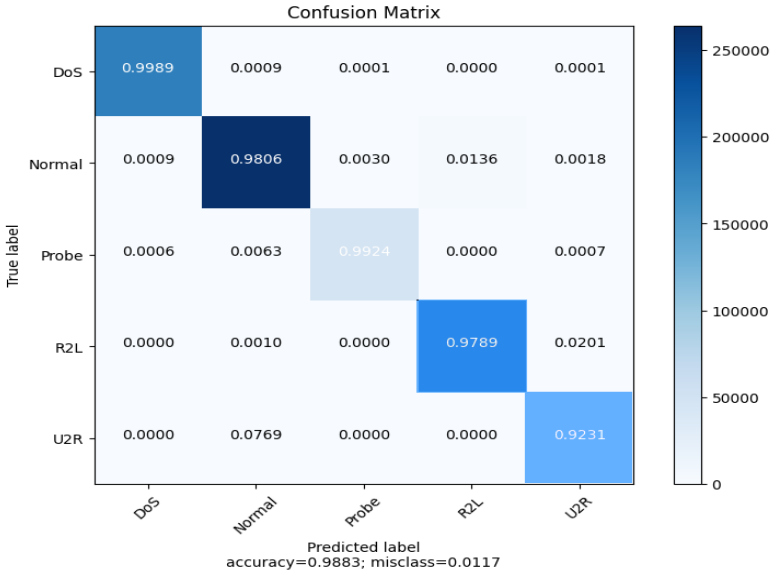

Figure 3. Best confusion matrix

Moreover, when considering additional metrics such as False Positive Rate (FPR) and Detection Rate (DR) across all folds, the model's effectiveness is further underscored. The False Positive Rate (FPR) for all folds ranges from 0,08 % to 0,73 %, indicating consistently low rates of false positives across different validation folds. Meanwhile, the Detection Rate (DR) for all folds ranges from 92,30 % to 99,88 %, highlighting the model's ability to detect instances of attacks accurately across different folds. The confusion matrix depicted in figure 3 demonstrates the model's proficiency in accurately classifying instances across various classes. These findings underscore the effectiveness of our intrusion detection approach, distinguishing it as a robust solution for identification and classification tasks. In comparison to other referenced methods[4,16,17] our approach stands out, offering an effective solution for identifying and classifying instances of network intrusion. These comparative results highlight the efficiency of our approach and showcase its competitive performance in the field of intrusion detection, especially when compared to established techniques mentioned in the literature.

CONCLUSIONS

This study delves into an effective deep learning approach tailored for the detection of attacks, with a specific emphasis on integrating CNN with Bi-GRU. The methodology is designed for the identification and classification of web attacks, commencing with data preprocessing on the NSLKDD dataset to extract relevant information. Subsequently, a CNN-BiGRU architecture is deployed, leveraging its automatic feature extraction capabilities to classify network traffic instances based on their attack class. This approach significantly enhances the model's proficiency in recognizing intricate patterns, thereby improving its ability to identify potential intrusions within the network.

An integral aspect of our approach is the incorporation of the Attention Layer mechanism, which plays a pivotal role in selectively focusing on crucial features during the learning process. This attention mechanism enables the model to assign varying levels of importance to different parts of the input sequence, contributing to the accurate identification of relevant patterns associated with web attacks. Additionally, by oversampling the minority class, SMOTE ensures that the model is exposed to a more balanced representation of both normal and attack instances, leading to improved generalization and detection performance.

The experiments conducted on the NSLKDD dataset demonstrate that the proposed method is easily trainable, achieving a high detection rate while maintaining low false alarms in identifying web attacks. In future studies, we aim to evaluate the model across diverse intrusion detection datasets to examine its ability to generalize across a range of scenarios.

BIBLIOGRAPHIC REFERENCES

1. Emad E. Abdallah WE. Intrusion Detection Systems using Supervised Machine Learning Techniques. Procedia Computer Science. 2022;Volume 201:Pages 205-212.

2. D S, L S. Sentence level Classification through machine learning with effective feature extraction using deep learning. Salud, Ciencia y Tecnología - Serie de Conferencias. avr 2024;3:702.

3. C. Yin YZ. A Deep Learning Approach for Intrusion Detection Us ing Recurrent Neural Networks,. in IEEE Access, vol 5, pp 21954-21961, 2017, doi: 101109/ACCESS20172762418.

4. Cui J. A novel multi-module integrated intrusion detection system for high-dimensional imbalanced data. Appl Intell 53, 272–288 (2023) https://doi.org/101007/s10489-022-03361-2. 2023.

5. Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, Masanori Koyama. Optuna: A Next-generation Hyperparameter Optimization Framework. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining; 2019; Anchorage, AK, USA.

6. Amir El-Ghamry AD. An optimized CNN-based intrusion detection system for reducing risks in smart farming. Internet of Things. 2023;volume 22.

7. Li Y ZB. An Intrusion Detection Algorithm Based on Deep CNN[J]. Computer Application and Software,37(4):324-328. 2020.

8. Kumar PM, Vedantham K, Selvaraj J, Kavin BP. Enhanced Network Intrusion Detection System Using PCGSO-Optimized BI-GRU Model in AI-Driven Cybersecurity. In: 2024 IEEE 3rd International Conference on AI in Cybersecurity (ICAIC). 2024. p. 1-6.

9. Kishor P. Jadhav TA. Intrusion Detection System Using Recurrent Neural Network-Long Short-Term Memory. Journal of In telligent Systems and Applications in Engineering, 11(5s), 563–573. 2023.

10. Vinayakumar R, Alazab M, Soman KP, Poornachandran P, Al-Nemrat A, Venkatraman S. Deep learning approach for intelligent intrusion detection system. Ieee Access. 2019;7:41525-50.

11. Hassan SA, Khalil MA, Auletta F, Filosa M, Camboni D, Menciassi A, et al. Contamination Detection Using a Deep Convolutional Neural Network with Safe Machine—Environment Interaction. Electronics. 2023;12(20):4260.

12. S. ElSayed NALK. A novel hybrid model for intru sion detection systems in sdns based on cnn and a new regularization technique. J Netw Comput Appl, 191 (2021), p 103160, 101016/j.jnca2021103160.

13. M. D. Mauro GG. Experimental review of neural-based approaches for network intrusion management. IEEE Trans Netw Serv Manage, 17 (4) (2020), pp 2480- 2495, 101109/TNSM20203024225.

14. Hao SL. BL-IDS: Detecting Web Attacks Using Bi-LSTM Model Based on Deep Learning. Crossref DOI link: https://doi.org/101007/978-3-030-21373-2_45 Published Online: 2019-06-08 Published Print: 2019.

15. Chen W, Shi K. Multi-scale Attention Convolutional Neural Network for time series classification. Neural Networks. 2021;136:126-40.

16. Zhang J, Zhang X, Liu Z, Fu F, Jiao Y, Xu F. A Network Intrusion Detection Model Based on BiLSTM with Multi-Head Attention Mechanism. Electronics. 2023;12(19):4170.

17. Song Y, Luktarhan N, Shi Z, Wu H. TGA: A Novel Network Intrusion Detection Method Based on TCN, BiGRU and Attention Mechanism. Electronics. 2023;12(13):2849.

18. Fu Y; D. A Deep Learning Model for Network Intrusion Detection with Imbalanced Data. Electronics 2022, 11, 898 https://doi.org/103390/electronics11060898. 2022.

19. Herve Nkiama SZMS. A Subset Feature Elimination Mechanism for Intrusion Detection System. (IJACSA) International Journal of Advanced Computer Science and Applications,. 2016;Vol. 7, No. 4.

20. Knowledge Discovery and Data Mining. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. The Fifth International Conference on Knowledge Discovery and Data Mining.

FINANCING
No financing.

CONFLICT OF INTEREST
Authors declare that there is no conflict of interest.

AUTHORSHIP CONTRIBUTION
Conceptualization: Asmaa Benchama, Khalid Zebbara.
Data curation: Asmaa Benchama, Khalid Zebbara.
Formal analysis: Asmaa Benchama, Khalid Zebbara.
Research: Asmaa Benchama, Khalid Zebbara.
Methodology: Asmaa Benchama, Khalid Zebbara.
Software: Asmaa Benchama, Khalid Zebbara.
Writing - proofreading and editing: Asmaa Benchama, Khalid Zebbara.