

INERCIA: An Advanced System for Cyberattack Detection through NLP using a Fine-Tuned LLM

Juan José Domínguez-Jiménez^{*,†}, Inmaculada Medina-Bulo[†], Juan Boubeta-Puig[†], Pedro Delgado-Pérez[†], Francisco Palomo-Lozano[†], Gabriel Guerrero-Contreras[†], Sara Balderas-Díaz[†] and Leopoldo Gutiérrez-Galeano[†]

UCASE Software Engineering Research Group, Department of Computer Science and Engineering, University of Cádiz, Avda. Universidad de Cádiz, 10, Puerto Real, 11510, Cádiz, Spain

Abstract

Conventional network intrusion detection systems present limitations in identifying emerging attack vectors. To address this limitation, this paper presents INERCIA, a project that proposes a novel Network Intrusion Detection System (NIDS) that applies Natural Language Processing (NLP) techniques by fine-tuning Large Language Models (LLMs). Departing from signature-based tools and machine learning classifiers, the approach adapts the pre-trained “Text-To-Text” Transfer Transformer (T5) encoder-decoder architecture to characterise network flow statistics. This is achieved by converting numerical features into an artificial language, allowing the model to treat threat detection as a sequence-to-sequence translation task. Validation on the CIC-IDS-2017 and CSE-CIC-IDS-2018 datasets yielded accuracy and F-scores exceeding 99.94% and 99.84%, respectively. The methodology demonstrates generalisation across attack vectors while maintaining a detection latency between 0.032s and 0.037s, making this novel NIDS highly effective for real-time cyberattack detection.

Keywords

Large Language Models, Fine-Tuning, Cybersecurity, Network Intrusion Detection, NLP applications

1. Introduction

Cybersecurity has become an essential field due to the constant emergence of new risks. Given the broad array of threats that can impact a computer system, various techniques and tools must be employed to minimise the impact of the persistent attacks occurring across data networks. Nevertheless, these measures often remain insufficient, as cybercriminals stay at the forefront, continually developing new types of cyber-threats [1].

Consequently, protecting information is vital, as it represents any organisation’s most valuable asset. To this end, a wide range of tools is available to mitigate the risks these systems face.

In terms of organisational network protection [2], there are Security Information and Event Management (SIEM) systems [3], Intrusion Prevention Systems (IPS) [4], and Intrusion Detection Systems (IDS) [5].

The primary characteristic of all aforementioned tools is the detection of cyber-attacks. However, the continuous appearance of new attack vectors necessitates detection systems capable of inferring new threats through artificial intelligence techniques [6]. To develop such systems, deep learning algorithms can be utilised, specifically models based on neural networks [7]. This can be addressed by defining and

SEPLN 2026: 42nd International Conference of the Spanish Society for Natural Language Processing, León, Spain, 22-25 September 2026.

*Corresponding author.

[†]These authors contributed equally.

✉ juanjose.dominguez@uca.es (J. J. Domínguez-Jiménez); inmaculada.medina@uca.es (I. Medina-Bulo); juan.boubeta@uca.es (J. Boubeta-Puig); pedro.delgado@uca.es (P. Delgado-Pérez); francisco.palomo@uca.es (F. Palomo-Lozano); gabriel.guerrero@uca.es (G. Guerrero-Contreras); sara.balderas@uca.es (S. Balderas-Díaz); leopoldo.gutierrez@uca.es (L. Gutiérrez-Galeano)

ORCID: 0000-0002-0877-7063 (J. J. Domínguez-Jiménez); 0000-0002-7543-2671 (I. Medina-Bulo); 0000-0002-8989-7509 (J. Boubeta-Puig); 0000-0003-1568-9288 (P. Delgado-Pérez); 0000-0002-6773-205X (F. Palomo-Lozano); 0000-0002-7509-3673 (G. Guerrero-Contreras); 0000-0003-1998-5826 (S. Balderas-Díaz); 0000-0001-8322-8470 (L. Gutiérrez-Galeano)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

training a model from scratch, or by selecting an existing pre-trained model and applying fine-tuning or transfer learning techniques; while these methods are similar, they maintain distinct differences.

To address these challenges, NLP techniques, specifically the fine-tuning of LLMs, can be applied to the cybersecurity domain. While traditionally used for linguistic tasks, these models can be adapted to identify patterns in non-textual data.

The INERCIA project proposes the adaptation of the pre-trained T5 transformer architecture to process network flow statistics. By representing these statistics as an abstract artificial language, the model is configured to perform multiclass classification of cyberattacks. This approach aims to achieve precise identification of threats through the fine-tuning of LLMs within the cybersecurity domain. This INERCIA project has been funded by Plan Propio de Investigación of the University of Cádiz, Spain, from April 2025 to October 2026.

The remainder of this paper is organised as follows. Section 2 reviews related works in the field of network security. Section 3 provides an INERCIA project overview, while Section 4 presents the datasets analysis. The selection of pre-trained models is discussed in Section 5, followed by the system design in Section 6 and its subsequent implementation in Section 7. Preliminary results and validation are detailed in Section 8, and finally, Section 9 offers conclusions and future work.

2. Related works

Transfer learning [8] is a technique focused on modifying part of an existing, pre-trained neural network architecture. This involves replacing the final layers with new ones relevant to the current problem and subsequently retraining only the altered layers to adapt the model to a new domain.

Although the application of artificial intelligence to the detection of cyberattacks offers enormous potential for overcoming current research obstacles, this field is still in its infancy. Existing studies vary so widely in terms of their data sources and training environments that it remains difficult to draw definitive conclusions. In this context, the INERCIA project aims to adapt the architecture of the pre-trained T5 model. Rather than treating network traffic as raw numbers, our method involves a translation into a tailor-made abstract artificial language. In this way, we enable the model to achieve a deep understanding of the data, allowing it to perform multiclass classification of cyberattacks with a high degree of accuracy.

The overall aim of the INERCIA project is to develop a cyberattack detection system using a pre-trained deep learning model designed for natural language processing. Through a process of fine-tuning, the system should be able to interpret network traffic statistics and adapt to a new task: the detection of cyberattacks.

On the other hand, fine-tuning [9] is an approach used to adjust the weights of an existing neural network without altering its architecture, aiming to solve a specific problem different from its original training task. The starting point is a pre-trained model which is transformed into a new one following a further training phase using new data. Fine-tuning techniques are typically used to address a more specific problem within the original scope for which a model was trained. For instance, if a model is required to identify types of flowers, one might select a model already trained to identify objects (including flowers in general). A fine-tuning process is then carried out to change the model's focus towards identifying specific flower types rather than general objects. However, in our case, we are radically changing the model's original purpose. When applying this technique, all layers of a model require a degree of learning. Regardless, this remains faster than training a model from scratch [8].

Nowadays, the majority of studies found utilise techniques based on deep learning [10, 11, 12, 13, 14, 15, 16, 17, 18, 19]. LeCun et al. [20] define deep learning as computational models based on neural networks composed of multiple layers, which learn from data with various levels of abstraction. Furthermore, this technique has drastically improved systems based on natural language processing, object recognition, speech recognition, object detection, and many others.

3. INERCIA project overview

The core objective of INERCIA is to adapt the pre-trained T5 transformer architecture to comprehend network flow statistics. By translating these statistics into an abstract artificial language, we enable the model to perform multiclass classification of cyberattacks with high precision.

The research hypotheses are:

- **H1:** Deep Learning techniques and high-quality datasets significantly improve the capacity to detect modern cyberattacks compared to traditional methods.
- **H2:** Fine-tuning pre-trained NLP models for cybersecurity tasks is a valid technique that yields higher precision and generalisation capacity than models trained from scratch.
- **H3:** The proposed system will demonstrate promising effectiveness and a reduction in false positives (FPR) in imbalanced environments where specific attack types are underrepresented.

The project objectives are:

- **O1: Dataset and technique study:** It performs an exhaustive analysis of existing datasets (CIC-IDS-2017, CSE-CIC-IDS2018, ...) and identifies suitable pre-trained models.
- **O2: Design and fine-tuning evaluation:** It develops the “Feature-to-Text” preprocessing pipeline, optimises hyperparameters to prevent overfitting, and executes the fine-tuning process.
- **O3: System validation:** It validates the system using unseen data and performs a comparison against existing literature.

In the following sections, a detailed analysis of the most relevant datasets used in cybersecurity, the selection of pre-trained models, the system design as well as its implementation and the preliminary results of the project can be found.

4. Datasets analysis

As a result of Objective O1, it was observed that the availability of public datasets within the field of cyberattack detection systems remains significantly limited. The most relevant datasets identified during this study are presented below, highlighting their significance and core characteristics.

Table 1 provides a comparative analysis of these datasets. While NSL-KDD [21] includes a wide variety of attack types, it contains the lowest number of records. Conversely, AWID2 [22] possesses the largest volume of records, yet its scope is restricted to 802.11 networks and only encompasses three attack categories. The CIC-IDS-2017 [23, 24] and CSE-CIC-IDS2018 [25, 24] datasets successfully combine a diverse range of attack types with a substantial volume of records, the latter being particularly extensive. These are considered the most up-to-date and representative of modern cyber threats. Both were generated using the CICFlowMeter [26] tool; consequently, their structure and feature count (85) are identical.

Table 2 illustrates the specific types of attacks included in each dataset. Specifically, it allows for an immediate identification of which datasets contain DoS, DDoS, Brute Force, Web Attack, Bot, Infiltration, or other categories of threats.

5. Selection of pre-trained models

In accordance with Objective O1, an extensive comparative analysis of state-of-the-art LLM architectures was conducted to identify the most suitable candidate for network traffic characterisation. This study evaluated the performance and structural advantages of *encoder-only*, *decoder-only* and *encoder-decoder* paradigms.

The analysis revealed that while Encoder-only models (such as BERT) are highly efficient at capturing bidirectional context for discriminative tasks, they lack the generative flexibility required to handle

Table 1

Cyberattack detection datasets comparison.

Dataset	Attacks	Columns	Rows	Tools
KDD Cup 1999	4	41	4,898,431	Tcpdump
NSL-KDD	4	41	125,973	Weka
UNSW-NB15	9	49	2,540,044	Tcpdump, Argus and Bro-IDS
CIC-IDS-2017	14	85	2,830,743	CICFlowMeter
CSE-CIC-IDS2018	14	85	16,232,943	CICFlowMeter
CIC-DDoS2019	12	85	12,794,627	CICFlowMeter
AWID2	3	155	42,388,298	Tshark
AWID3	13	253	15,574,911	Tshark
BCCC-CIC-IDS-2017	13	122	2,438,052	NTLFlowLyzer

Table 2

Types of attacks included in each dataset.

DATASET	DoS	DDoS	BRUTE FORCE	WEB ATTACK	BOT	INFIL.	OTHERS
KDD Cup 1999	✓		✓				✓
NSL-KDD	✓		✓				✓
UNSW-NB15	✓						✓
CIC-IDS-2017	✓	✓	✓	✓	✓	✓	✓
CSE-CIC-IDS2018	✓	✓	✓	✓	✓	✓	
CIC-DDoS-2019		✓					✓
BCCC-CIC-IDS-2017	✓	✓	✓	✓	✓	✓	✓
AWID2 / AWID3	✓		✓				✓

the translation of complex artificial languages into structured labels. Conversely, decoder-only models (such as GPT-2) excel in open-ended generation but are often less efficient in strictly mapping structured input features to specific categorical outputs in a constrained NIDS environment.

Based on these findings, we selected the T5 encoder-decoder architecture. Unlike encoder-only models, such as BERT, which excel at classification, or decoder-only models, such as GPT, which excel at generation, T5's text-to-text framework is uniquely suited for translating our abstract artificial language into specific attack labels. This versatility allows for more robust sequence-to-sequence prediction in multiclass environments.

Furthermore, leveraging a pre-trained T5 checkpoint ensures that the model benefits from vast prior linguistic knowledge, which, when fine-tuned, results in the exceptional accuracy and low detection latency (0.032s – 0.037s) achieved by the INERCIA project.

6. System design

Regarding Objective O2, Figure 1 presents the architecture of the proposed LLM-based cyberattack detection system. The framework is divided into two primary stages: the ETL (Extract, Transform, Load) Module and the Detection Module. Initially, raw network traffic is processed to extract statistical features, which are then converted from numerical values into a textual format to create a compatible LLM input. This structured data is subsequently fed into a fine-tuned LLM. If the model identifies the patterns as an anomaly or a known threat, an 'Attack' state is triggered, resulting in the generation of a system alarm.

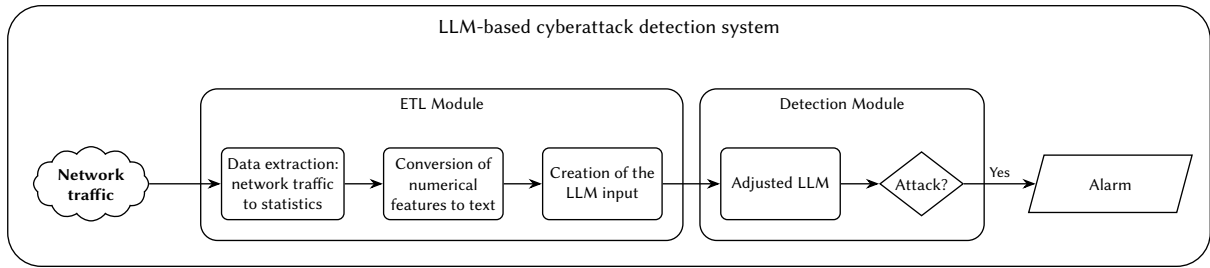


Figure 1: Operational workflow of the LLM-based detection system

6.1. Data acquisition

The system utilises CICFlowMeter to generate network flow statistics. This tool translates raw network packets into 84 distinct traffic characteristics (features), providing the high-dimensional data necessary for deep learning analysis.

6.2. The ETL Module

The ETL module performs a “Feature-to-Text” conversion. Since LLMs are designed for textual input, this module transforms numerical characteristics into string representations, enabling the transformer to interpret non-linguistic network data.

6.3. The abstract language

The sequential concatenator joins these converted strings in a fixed order. This process effectively translates a network flow into a structured textual sequence. Figure 2 illustrates the data preprocessing workflow, where raw network flow statistics from the dataset are transformed into a specialised format. The numerical features—such as destination port, flow duration, and packet counts—are extracted and combined through a concatenation process. Numerical features are joined using the vertical bar (|) separator to create a unique sequence. This results in a unified sequence that serves as the input for the model, represented here as an abstract artificial language.

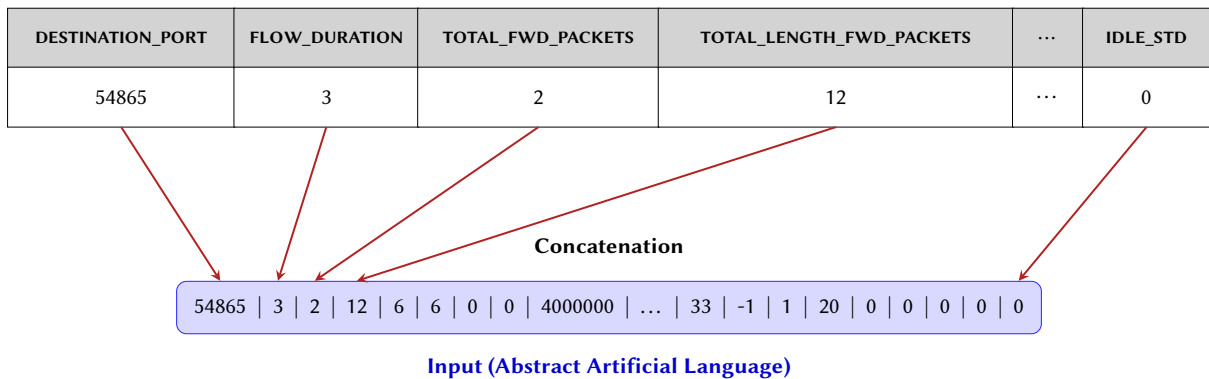


Figure 2: Abstract artificial language.

7. System implementation

The implementation utilises the SimpleT5 [27] wrapper, built upon PyTorch 2.0.1. This framework facilitates the adjustment of all model weights, ensuring that the transformer’s internal mechanisms are fully adapted to the nuances of cybersecurity traffic.

To ensure model stability and mitigate the risk of overfitting, the hyperparameters were tuned according to the configurations presented in Table 3. The `source_max_token_len` of 150 was selected based on empirical testing. While the strings averaged 189 characters, 150 tokens provided the optimal balance between metric stability and training time. Increasing this value beyond 150 resulted in longer training durations without statistically significant gains in accuracy or F-score.

Table 3

Hyperparameter settings for T5 fine-tuning.

Hyperparameter	Value
<code>max_epochs</code>	10
<code>batch_size</code>	16
<code>precision</code>	32
<code>source_max_token_len</code>	150

The experiments were performed in an environment with software that is commonly used for data science. We selected Python 3.8.17 to develop the experiments. All experiments were performed on the supercomputer of the University of Cádiz. It contains several NVIDIA A100 GPUs with 40GB of GPU memory and 256GB of RAM, each node with 2 CPUs and each with 64 cores. Moreover, it contains a number of nodes with more memory, more concretely with 1TB each node.

8. Preliminary results and validation

According to Objective O3, preliminary results and the comprehensive validation of the system are published in [28]. The system was validated using three benchmark datasets: CIC-IDS-2017, CSE-CIC-IDS-2018, and the augmented BCCC-CIC-IDS-2017. The structural characteristics of the selected CIC datasets are summarised in Table 4. These benchmarks were prioritised due to their comprehensive nature and high dimensionality; specifically, the CIC-IDS-2017 and CSE-CIC-IDS-2018 datasets feature 85 features (columns), while the BCCC-CIC-IDS-2017 provides an extended set of 122. Together, these benchmarks capture a diverse range of modern attack vectors, varying from 12 to 14 distinct categories. Notably, the datasets provide a substantial volume of data for model training and validation, with the CSE-CIC-IDS2018 and CIC-DDoS2019 versions exceeding 12 million records each. All three datasets were generated using the CICFlowMeter tool, ensuring a consistent feature set across different temporal captures, which is essential for evaluating the generalisation capabilities of the proposed detection system. To address class imbalance, we performed random undersampling of the benign class. For example, in the CSE-CIC-IDS-2018 dataset, this resulted in a final distribution of 53.16% benign and 46.84% malignant instances. Rigorous validation was ensured through stratified fivefold cross-validation, maintaining consistent label proportions across all folds.

Table 4

Characteristics of the CIC-IDS benchmark datasets.

Dataset	Attacks	Columns	Rows	Tools
CIC-IDS-2017	14	85	2,830,743	CICFlowMeter
CSE-CIC-IDS2018	14	85	16,232,943	CICFlowMeter
BCCC-CIC-IDS-2017	13	122	2,438,052	NTLFlowLyzer

8.1. Performance metrics

The performance metrics of the proposed system across three benchmark datasets are summarised in Table 5. The results demonstrate exceptional robustness, with accuracy and F-score values consistently

exceeding 99.8% in all evaluated scenarios. Notably, the model achieved its highest accuracy on the CIC-IDS-2017 dataset at 99.94%. Furthermore, the system maintained remarkably low error rates, particularly regarding the False Positive Rate (FPR) and False Negative Rate (FNR), which remained below 0.3% and 0.05% respectively. This high level of precision and recall suggests that the fine-tuning approach effectively captures the nuances of diverse cyberattack patterns while minimising misclassifications.

Table 5

Performance metrics across benchmark datasets.

Dataset	Accuracy	Precision	Recall	F-score	FPR	FNR
CIC-IDS-2017	99.94%	99.94%	99.94%	99.94%	0.01%	0.05%
CSE-CIC-IDS-2018	99.84%	99.84%	99.84%	99.84%	0.27%	0.02%
BCCC-CIC-IDS-2017	>99.90%	99.90%	99.90%	99.90%	0.11%	0.04%

8.2. Transparency and Minority Class Analysis

Despite high aggregate metrics, granular analysis of the confusion matrices revealed limitations in detecting minority classes. SQL Injection attacks, representing a very low proportion of instances, resulted in a 0% detection rate in BCCC-CIC-IDS-2017 and only 23.53% in CSE-CIC-IDS-2018. This indicates that the primary challenge for LLM-based NIDS remains the severe data imbalance at the sub-category level, where limited training samples hinder the model’s capacity to classify specific attack variants accurately.

8.3. Efficiency and generalisation

The average detection latency was recorded between 0.032s and 0.037s, supporting real-time implementation. The inference was carried out on a computer equipped with two Intel Xeon-Platinum 8358 CPUs (2.6GHz, 32 cores each), providing a total of 64 computing units and 256 GB of RAM. The convergence of training and validation loss curves by the second epoch indicates robust generalisation and the absence of overfitting.

9. Conclusions and future work

The INERCIA project confirms that fine-tuning the T5 encoder-decoder model is a promising strategy for NIDS, outperforming BERT, GPT-2, and traditional machine learning approaches. Our contributions include the development of a viable abstract artificial language for network statistics and the achievement of near-perfect detection rates on standard traffic without synthetic data.

INERCIA project aimed to identify different directions for future research to improve the effectiveness of our approach. A primary focus will be the mitigation of minority class failures; by implementing advanced data augmentation techniques such as the Synthetic Minority Oversampling TEchnique (SMOTE) or ADaptive SYNthetic (ADASYN), we aim to provide a more robust representation for under-detected threats like SQL Injection and Brute Force-Web attacks. Building on this foundation, we propose the development of a hierarchical classification framework. This would involve a cascaded architecture where an initial LLM handles rapid binary detection—distinguishing benign from malignant traffic—before passing the data to a more specialised model for granular multiclass categorisation. Finally, to ensure these systems are practical for real-world security analysts, we intend to integrate explainability and interpretability tools. By applying frameworks like Local Interpretable Model-Agnostic Explanations (LIME) or SHapley Additive exPlanations (SHAP) to the Transformer’s internal attention mechanisms, we can move beyond black-box AI, providing clear insights into the specific features that drive each classification decision.

Acknowledgments

This publication is part of the I+D+i grant PR2024-033 (INERCIA project) funded by Plan Propio de Investigación of the University of Cádiz.

The authors also thank the Systems Unit of the Information Systems Area of the University of Cádiz for computer resources and technical support.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

- [1] W. S. Admass, Y. Y. Munaye, A. A. Diro, Cyber security: State of the art, challenges and future directions, *Cyber Security and Applications* 2 (2024) 100031. doi:10.1016/j.csa.2023.100031.
- [2] INCIBE, ¿qué son y para qué sirven los SIEM, IDS e IPS?, 2020. URL: <https://www.incibe.es/protege-tu-empresa/blog/son-y-sirven-los-siem-ids-e-ips>.
- [3] M. Vielberth, G. Pernul, A security information and event management pattern, in: *Proceedings of the 12th Latin American Conference on Pattern Languages of Programs (SugarLoafPLOP 2018)*, ACM, Valparaíso, Chile, 2018. doi:10.5283/epub.41139.
- [4] A. S. Ashoor, S. Gore, Difference between intrusion detection system (IDS) and intrusion prevention system (IPS), in: *Advances in Network Security and Applications. CNSA 2011. Communications in Computer and Information Science*, volume 196, Springer, 2011, pp. 497–501. doi:10.1007/978-3-642-22540-6_48.
- [5] A. S. Ashoor, S. Gore, Importance of intrusion detection system (IDS), *International Journal of Scientific and Engineering Research* 2 (2011) 1–4. URL: <https://api.semanticscholar.org/CorpusID:7504589>.
- [6] M. Macas, C. Wu, W. Fuertes, A survey on deep learning for cybersecurity: Progress, challenges, and opportunities, *Computer Networks* 212 (2022) 109032. doi:10.1016/j.comnet.2022.109032.
- [7] A. Burkov, *The Hundred-Page Machine Learning Book*, Andriy Burkov, Quebec City, Canada, 2019.
- [8] E. C. Too, L. Yujian, S. Njuki, L. Yingchun, A comparative study of fine-tuning deep learning models for plant disease identification, *Computers and Electronics in Agriculture* 161 (2019) 272–279. doi:10.1016/j.compag.2018.03.032.
- [9] L. Alzubaidi, J. Bai, A. Al-Sabaawi, J. Santamaria, A. Albahri, et al., A survey on deep learning tools dealing with data scarcity: definitions, challenges, solutions, tips, and applications, *Journal of Big Data* 10 (2023) 46. doi:10.1186/s40537-023-00727-2.
- [10] M. Abdel-Basset, H. Hawash, R. K. Chakraborty, M. J. Ryan, Semi-supervised spatiotemporal deep learning for intrusions detection in IoT networks, *IEEE Internet of Things Journal* 8 (2021) 12251–12265. doi:10.1109/JIOT.2021.3060878.
- [11] L. Nie, Y. Wu, X. Wang, L. Guo, G. Wang, X. Gao, S. Li, Intrusion detection for secure social internet of things based on collaborative edge computing: A generative adversarial network-based approach, *IEEE Transactions on Computational Social Systems* 9 (2022) 134–145. doi:10.1109/TCSS.2021.3063538.
- [12] Y. Kunang, S. Nurmaini, D. Stiawan, B. Suprpto, An end-to-end intrusion detection system with IoT dataset using deep learning with unsupervised feature extraction, *International Journal of Information Security* 23 (2024) 1171–1186. doi:10.1007/s10207-023-00807-7.
- [13] S. Bhardwaj, M. Dave, Enhanced neural network-based attack investigation framework for network forensics: Identification, detection, and analysis of the attack, *Computers & Security* 135 (2023) 103521. doi:10.1016/j.cose.2023.103521.

- [14] Z. Wang, F. A. Ghaleb, An attention-based convolutional neural network for intrusion detection model, *IEEE Access* 11 (2023) 43116–43127. doi:10.1109/ACCESS.2023.3271408.
- [15] S. M. S. I. Tapu, S. S. Shopnil, R. Tamanna, M. A. M. Dewan, M. S. Alam, Malicious data classification in packet data network through hybrid meta deep learning, *IEEE Access* 11 (2023) 140609–140625. doi:10.1109/ACCESS.2023.3341911.
- [16] Y. Li, T. Qin, Y. Huang, J. Lan, Z. Liang, T. Geng, HDFEF: A hierarchical and dynamic feature extraction framework for intrusion detection systems, *Computers & Security* 121 (2022) 102842. doi:10.1016/j.cose.2022.102842.
- [17] M. Yilmaz, B. Bardak, An explainable anomaly detection benchmark of gradient boosting algorithms for network intrusion detection systems, in: *Proceedings of the 2022 Innovations in Intelligent Systems and Applications Conference (ASYU)*, IEEE, 2022, pp. 1–6. doi:10.1109/ASYU56188.2022.9925451.
- [18] M. Hammad, N. M. Hewahi, W. Elmedany, MMM-RF: A novel high accuracy multinomial mixture model for network intrusion detection systems, *Computers & Security* 120 (2022) 102705. doi:10.1016/j.cose.2022.102777.
- [19] L. Yu, J. Dong, L. Chen, M. Li, B. Xu, Z. Li, L. Qiao, L. Liu, B. Zhao, C. Zhang, PBCNN: Packet bytes-based convolutional neural network for network intrusion detection, *Computer Networks* 194 (2021) 108117. doi:10.1016/j.comnet.2021.108117.
- [20] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, *Nature* 521 (2015) 436–444. doi:10.1038/nature14539.
- [21] UNB, Nsl-kdd dataset, 2009. URL: <https://www.unb.ca/cic/datasets/nsl.html>.
- [22] U. of the Aegean, The awid2 dataset, 2016. URL: <https://icsdweb.aegean.gr/awid/awid2>.
- [23] UNB, Intrusion detection evaluation dataset (cic-ids2017), 2017. URL: <https://www.unb.ca/cic/datasets/ids-2017.html>.
- [24] I. Sharafaldin, A. Habibi Lashkari, A. A. Ghorbani, Toward generating a new intrusion detection dataset and intrusion traffic characterization, in: *Proceedings of the 4th International Conference on Information Systems Security and Privacy - ICISSP, INSTICC, SciTePress*, 2018, pp. 108–116. doi:10.5220/0006639801080116.
- [25] UNB, Ips/ids dataset on aws (cse-cic-ids2018), 2018. URL: <https://www.unb.ca/cic/datasets/ids-2018.html>.
- [26] UNB, Cicflowmeter, 2016. URL: <https://www.unb.ca/cic/research/applications.html#CICFlowMeter>.
- [27] S. Roy, simplet5, 2022. URL: <https://github.com/Shivanandroy/simpleT5>.
- [28] L. Gutiérrez-Galeano, J.-J. Dominguez-Jiménez, J. Schäfer, I. Medina-Bulo, LLM-based cyberattack detection using network flow statistics, *Applied Sciences* 15 (2025). doi:10.3390/app15126529.