ENHANCING INTRUSION DETECTION SYSTEMS USING BERT: A COMPARATIVE STUDY ON BENCHMARK DATASETS

An Internship Report Submitted

in

COMPUTER SCIENCE AND ENGINEERING

by

GIRIBALA ARUN

to

DR. P. VICTER PAUL

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY KOTTAYAM

KERALA – 686635, INDIA

AUGUST 2025

DECLARATION

I, **Giribala Arun**, hereby declare that, this report entitled **"Enchancing Intrusion Detection Systems Using BERT: A Comparative Study On Benchmark Datasets"** submitted to Indian Institute of Information Technology Kottayam as Internship Project is an original work carried out by me under the supervision of Dr. P. Victer Paul and has not formed the basis for the award of any degree or diploma, in this or any other institution or university. I have sincerely tried to uphold the academic ethics and honesty. Whenever an external information or statement or result is used then, that have been duly acknowledged and cited.

Kottayam - 686635 **Giribala Arun**

July 2025

CERTIFICATE

This is to certify that the work contained in this project report entitled **"Enchancing Intrusion Detection Systems Using BERT: A Comparative Study On Benchmark Datasets"** submitted by **Giribala Arun** to Indian Institute of Information Technology Kottayam as an Internship Project has been carried out by him under my supervision and that it has not been submitted elsewhere for the award of any degree.

Kottayam - 686635
July 2025

(Dr. P. Victer Paul)
Project Supervisor

ABSTRACT

With the growing complexity and frequency of cyberattacks, we have serious challenges that need to be addressed in the existing Intrusion Detection System (IDS). Traditional Machine Learning (ML) and Deep Learning (DL) models have shown promise, but fail to adapt to the nuanced and evolving nature of threats in modern times. In this research paper, we present BERT-IDS, a transformer-based approach to capture the contextual understanding of Bidirectional Encoder Representations from Transformers (BERT) for intrusion detection. The model is thoroughly evaluated against several widely used ML and DL algorithms – namely Random Forest, Support Vector Machines, Autoencoders, XGBoost, Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM). BERT-IDS's performance was closely observed across multiple performance metrics (e.g. accuracy, recall, precision), which demonstrated extreme robustness when compared with the values of traditional ML and DL models. The outcomes emphasize the potential of transformer architectures in augmenting the adaptability and flexibility of IDS in the ever-changing cybersecurity environments.

Contents

Chapter 1

INTRODUCTION

In an increasingly connected digital world, where everything and everyone is connected, the security and the integrity of networks is of utmost importance. Cyberattacks have increased not only in number but also in sophistication in recent years, and it is apparent that robust strategies must be implemented to combat these. One essential line, in the context of a cybersecurity implementation, is an Intrusion Detection System (IDS) - systems that identify abnormal or suspicious behaviour in network traffic.

The problem is, traditional IDS systems, especially signature-based ones, start out restrictive. While signature-based systems use an already identified and encoded signature of identified threats, they are ineffective when it comes to detecting new or changing tag types. This has necessitated a shift from traditional static detection strategies to more intelligent adaptive ones.

Advances in Artificial Intelligence (AI) create possibilities for more robust models for IDS. The models in Machine Learning (ML) like Support Vector Machines (SVM) and Random Forests (RF) have shown to identify patterns in the data and are exceptional at identifying anomalies with a flexible approach. More recent advances in Deep Learning (DL) models, especially the Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM) models, are better at identifying anomalies that exist in traffic sequences based on their spatial and temporal patterns.

A growing avenue in this area is the utilization of transformer-based architectures,

designed for Natural Language Processing (NLP). BERT (Bidirectional Encoder Representations from Transformers) has the potential to model complex contextual dependencies within sequential data. This makes BERT and similar models strong candidates for modern, adaptive IDS, capable of learning from evolving traffic patterns and responding to sophisticated cyber threats in real time.

Chapter 2

RATIONALE FOR THE STUDY

2.1 Background and Motivation

In this increasingly well-connected digital landscape, emphasis on the security and integrity of networked systems have become a priority. In recent years, we have observed a rise in both frequency and complexity of cyber attacks [1–3]. To combat these threats, it is important to employ Intrusion Detection Systems (IDS) to find abnormal behaviour in network traffic. Traditional types of IDS, especially signature-based IDS, are not very effective at catching new threats or adapting to an evolving threat because they rely on attack signatures that were preconfigured [4, 5].

New developments in Artificial Intelligence (AI) have changed the intrusion detection landscape. ML models like Support Vector Machines (SVM) and Random Forests (RF) have the capability to detect patterns within traffic data [6, 7]. While DL based models, such as Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks, have made improvements at capturing both temporal and spatial dependencies that exist in complex network traffic [8–10].

An emerging area of development is the adaptation of transformer-based models, which were originally based in Natural Language Processing (NLP) domain. Among them, a popular variant of transformer models – BERT – excels at learning con-

textual dependencies from sequential data [11–13]. The ability to model intricate patterns in both structured and unstructured data makes it a potential system for contemporary adaptive IDS systems [14, 15].

2.2 Research Gap

Although transformer-based models have been successfully employed in NLP and text classification tasks, their application in intrusion detection remains underexplored [18, 19]. Existing research has predominantly focused on traditional ML/DL frameworks, with limited attention given to the contextual modeling capabilities of architectures like BERT in the domain of cybersecurity [20]. Moreover, comparative evaluations involving BERT and classical models across standard IDS datasets are scarce, creating a research void in understanding its practical applicability [21, 22]. This gap highlights the need for a context-aware IDS framework that integrates pre-trained language models to enhance detection performance and adaptability in modern network environments.

2.3 Objective of the Study

This study aims to propose and evaluate a BERT-based Intrusion Detection System (BERT-IDS) that leverages the encoding capabilities of transformer architectures for enhanced threat detection. The model shall be compared with established ML and DL algorithms: SVM, RF, CNN, LSTM, Autoencoders, and XGBoost against benchmarks datasets CICIDS2017, NSL-KDD, and UNSW-NB15 [23–25]. The evaluation will be based on standard metrics such as accuracy, precision, recall, specificity, F1-score, and ROC-AUC to benchmark all of the models. The goal is to demonstrate how well transformer models are able to capture contextual signals relevant to intrusion detection and how they can be potentially deployed in a real-time cybersecurity application.

Chapter 3

LITERATURE REVIEW

3.1 Traditional Machine Learning Techniques for IDS

Traditional ML models like RF, SVM, k-Nearest Neighbors (K-NN), and Naive Bayes have been widely used in IDS tasks due to their simplicity, and effectiveness. Random Forest is used extensively because it is noise-tolerant and can effectively work with larger datasets as a robust classifier. The accuracy for Random Forest is generally among the highest on standard datasets [1,2]. SVM is known for its ability to model complex decision boundaries through kernel functions, allowing it to outperform other models in scenarios with clearly separable attack classes. However, it suffers from scalability issues with large datasets [3,4]. K-NN is relatively easy to implement, but struggles with high dimensionality and computational overhead during classification tasks [5]. Although Naive Bayes provides very quick computations, performance suffers violations of the independence assumption throughout the features when applied to IDS data [6,7]. Though these techniques yield high accuracy on well-structured datasets like NSL-KDD, their performance is still reduced drastically when applied to complex, dynamic or unbalanced datasets like UNSW-NB15, due to a lack of generalization of evolving attack patterns [8,9].

3.2 Deep Learning Approaches in IDS

Deep Learning methods include Convolutional Neural Networks (CNN), Long Short Term-Memory (LSTM), Gated Recurrent Units (GRU) and Autoencoders, that are beneficial for IDS – they have helped to learn complex spatial and temporal patterns in network data. The CNNs present a unique capability to extract spatial feature information from structured data (e.g. network flow data), resulting in improved accuracy on datasets such as CICIDS2017 and UNSW-NB15 [10,11]. LSTM and GRU models were more effective at modelling the temporal dependencies within sequential traffic data and performed better than the earlier applications at recognizing long-term attack patterns [12,13]. Autoencoders which were specifically used for anomaly detection, can effectively model normal network behaviour to detect previously unseen intrusions with low false discovery rates; but generally, they do not perform as well when the attack was labelled due to reliance on merely the reconstruction error for defining the threshold for such attacks [14]. Despite these developments, DL approaches face limitations like being computationally expensive, requiring a large hyperparameter search space, and their tendency to overfit especially when small or unbalanced data sets are used for training [15,16]. Consequently, recent studies call for hybrid architectures combining CNN and LSTM or GRU to tap into their complementary strengths and achieve better accuracy and detection abilities on common datasets [17,18].

3.3 Transformer Models and BERT Architecture

Transformers, introduced by Vaswani et al. (2017), have made notable progress in Natural Language Processing (NLP) by utilizing self-attention mechanisms to model dependencies in sequential data [19]. The widely known transformer model – BERT – is designed to capture bidirectional context and to help us understand more complex language structures [20]. These transformer models demonstrate strong cross-domain adaptability, effectively modelling sequential and high-dimensional data across domains like bioinformatics, finance and cybersecurity [21,22]. Due to their ability to capture intricate feature interactions and temporal dependen-

cies in inherent network traffic data, transformers pose as excellent candidates for IDS – thus surpassing traditional ML/DL models in handling dynamic, unstructured network datasets [23]. The inherent self-attention mechanism facilitates the transformer models to selectively focus on features of interest and the contextually relevant data points with one another that advance detection capabilities for complex, subtle attack patterns that are common in modern cyber threat scenarios.

3.4 Recent Studies Using BERT in Cybersecurity

Recent studies have explored the application of BERT in the field of cybersecurity, notably NetBERT, CyberBERT, and IDS-BERT. NetBERT and CyberBERT work by embedding network logs and security events into vector spaces to help with detection tasks such as anomaly detection and threat intelligence [24,25]. IDS-BERT specifically adapts to transformer-based architectures for intrusion detection, and shows great performance on the CICIDS2017 and UNSW-NB15 datasets. Overall, studies have used pretrained BERT models that were fine-tuned with variability on a cybersecurity-specific corpora with varying datasets with large log datasets and extrema in training, using different attack scenarios. While improvements in performance have been recorded, there are limitations to this: the computational overhead sustained for the large-scale, domain-specific training [26,27]. The work highlights future research opportunities for efficient transformer-based architectures and contextual embedding models specific to domains to support intrusion detection system deployment in real-world applications.

3.5 Summary and Research Gap

In conclusion, although traditional ML approaches are effective on structured datasets, they do not perform adequately on delimited problems like evolving and complex attacks. DL techniques can leverage their ability to capture complex temporal and spatial patterns, but there are still considerations regarding data and computational capabilities. Transformer-based models, particularly BERT, demonstrate strong potential due to their contextual understanding properties and ability to generalize

pose an interesting opportunity. However, there have not been many works to apply or fine-tune a transformer model specifically for IDS-related tasks. This research fills the gap by proposing a modified BERT-IDS, and we demonstrate its capability against traditional models and deep learning models on a variety of challenging and evolving datasets specific to intrusion detection.

Chapter 4

PROBLEM STATEMENT

4.1 Problem Statement

Enhancing Intrusion Detection Systems Using BERT: A Comparative Study on Benchmark Datasets

4.2 Explanation

Even though there has been progress in the area of intrusion detection through ML and DL techniques, both types of models often do not generalize well with complex, high dimensional data. In particular, the combination of heterogeneous data which has both structured features (i.e. source/destination IPs, ports) and unstructured data (i.e. logs, payloads) is an ongoing struggle for both supervised and unsupervised ML [16, 17]. Additionally, many current models exhibit elevated false positive rates and are limited in their adaptability to emerging attack patterns. There remains a pressing need for an IDS solution that not only identifies threats in real time but also leverages contextual awareness to respond to evolving intrusion tactics.

Chapter 5

BERT-BASED IDS MODEL

5.1 Overview of the Architecture

The proposed BERT-based Intrusion Detection System (BERT-IDS) architecture will leverage transformer models in order to properly detect network intrusions. The primary motivation for using a transformer-based model, like BERT, directly co-relates to transformers being able to better capture context and dependencies in sequence and high-dimensional data that are necessary to accurately detect sophisticated cyber-attacks [21,22]. Unlike traditional models that may overlook subtle feature interactions, BERT's self-attention mechanism allows the model to dynamically weigh the features, ultimately improving the model's detection capacity for narrowly delineated intrusion patterns [23].

In the BERT-IDS pipeline the network traffic data is pre-processed from standard datasets like CICIDS2017, UNSW-NB15 and NSL-KDD, to convert it into a structured input format appropriate to transformer specifications. This includes normalization, categorical embedding and generating a token-like sequential representation. Then BERT is trained specifically for intrusion detection application to produce contextualized embeddings based on the compiled tokenized-type input. These embeddings are then passed through dense classification layers with activation functions suitable for multiclass or binary classification tasks (like SoftMax or sigmoid). Uti-

lizing the transformer architecture was a useful method for experimenting with the IDS space and offers a significant increase in detection capabilities over traditional ML and DL methods [24,26].

5.2 Input Representation and Preprocessing

Proper pre-processing and representation of network data are crucial steps to be taken before training a transformer model for intrusion detection. The structured datasets utilized in the experiment – CICIDS2017, UNSW-NB15 and NSL-KDD – include a combination of numerical and categorical features [10,11]. Numerical features undergo normalization techniques like Min-Max or Z-score normalization, which increases uniformity of values for convergence during training. Categorical features are transformed using embedding layers that convert discrete values into dense vector representations, enabling the transformer model to learn an interaction between the features [18,25].

To adapt structured data for BERT, a token-like sequential representation must be developed. Each network record was treated as a sequential representation of tokens, where each token encoded one feature or a group of features. A dense layer of these token embeddings was assembled and concatenated with positional encodings so the model could learn the dependence and relative position/importance of features within each record [19]. Ensuring dimensional alignment, each sequence maintains a consistent length and embedding dimension, compatible with the BERT input format, typically (sequence length $\times$ embedding size). Padding and truncation strategies are employed to handle variations in sequence length, thereby standardizing the inputs and enhancing the efficiency and consistency of the training process [20,23].

5.3 Model Architecture

The core of the proposed BERT-IDS model is an adapted BERT encoder architecture specifically optimized for intrusion detection tasks. Initially, a pretrained BERT model (such as BERT-base) is employed due to its proven contextual learn-

ing capabilities. The architecture consists of multiple transformer encoder layers, typically 12, each with multi-head self-attention mechanisms designed to capture diverse feature interactions from the network data [20,22]. Each attention head learns independently from the others and captures different aspects of the feature interactions, which allows the model to have a more holistic understanding of how the network traffic behaves.

To effectively handle sequential network data, the model incorporates positional encodings so that the transformer can account for the sequential nature of the input features [19]. Following the encoder layers, there is a classification head comprising of multiple dense layers. These dense layers use activation functions such as ReLU for intermediate layers and SoftMax for the final layer when doing multi-class classification, and sigmoid for binary classification [23].

Hyperparameters critical to the model's performance mainly include the number of transformer layers (e.g., 12), attention heads (e.g., 8 or 12), embedding dimensions (768 or 512), dropout rates for regularization and the learning rate schedules based on empirical evaluations. The architecture (Figure 2.1(a)) provides clarity around the data flow of the model through the different stages; preprocessing, embedding, positional encoding, transformer encoder layers, and the classification layers; and provides a clearer overview of the full model data flow pipeline.

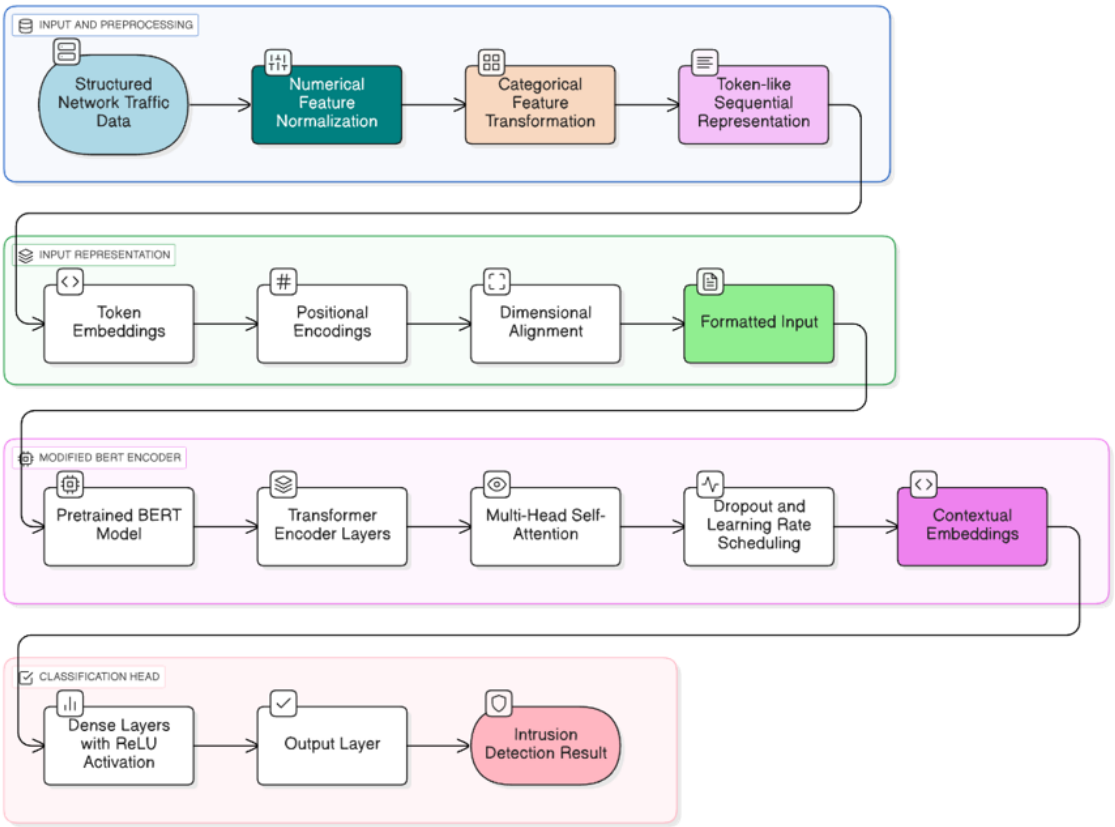

Figure 2.1(a): Overview of the Model Architecture

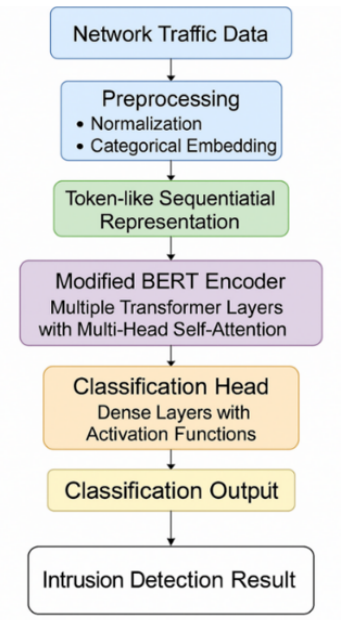

Figure 2.1(b): Flowchart of the Model Architecture

Chapter 6

EXPERIMENTATION AND RESULT ANALYSIS

6.1 Dataset

The three datasets used in this study: CICIDS2017, UNSW-NB15 and NSL-KDD are established benchmark datasets for IDS evaluation. Developed by the Canadian Institute for Cybersecurity, CICIDS2017 has high dimensions and imbalanced classes and is a realistic example of current attacks with realistic traffic. UNSW-NB15 is developed by the Australian Cyber-Security Centre, which includes nine attacks for hybrid network behaviours, and additional normal attacks. It represents a moderately complex environment, making it suitable for evaluating both classical and deep learning models. An improved version of the original KDD'99 dataset, NSL-KDD was created to remove bias and redundancy. While relatively simple and contains structured data, NSL-KDD is still a great dataset for comparing basic ML classifiers to other techniques and approaches to understand detection mechanisms with well-labelled records. Together, these datasets provide comprehensive coverage of real-world, semi-realistic, and traditional IDS challenges, ensuring a fair and diverse evaluation of models. The comparison of different features of the dataset is given in the Table 6.1.

Property	**CICIDS2017**	**UNSW-NB15**	**NSL-KDD**
Source	Canadian Institute for Cybersecurity (CIC)	UNSW Canberra / IXIA PerfectStorm	University of New Brunswick
Total Records	˜2.8 million	2.54 million (257,673 subset)	˜150,000 (Train + Test)
Features	80+ (78 numeric + label)	49 + label	41
Feature Types	Flow-based features (duration, packet stats, protocols, etc.)	Flow features (state, service, bytes, etc.)	Basic + content + traffic features
Classes	Binary & Multiclass	Binary & Multiclass	Binary & Multiclass
Attack Count and Types	14 – DoS, DDoS, Bot, Brute Force, Infiltration, etc.	10 – Fuzzers, Exploits, etc.	4 (39 attacks) – DoS, Probe, U2R, R2L, etc.
Label Type	Multiclass & Binary	Multiclass & Binary	Multiclass & Binary
Traffic Type	PCAP + NetFlow	Realistic synthetic traffic (2015)	Simulated DARPA-based logs
Dataset Complexity	High (realistic traffic & imbalance)	Moderate-High (modern threats)	Low-Moderate (older dataset)
File Format Size on Disk	CSV, PCAP ˜11 GB	CSV ˜700 MB	CSV, ARFF ˜20 MB
Challenge	Class imbalance, high dimensionality	Mild imbalance, protocol diversity	Outdated attack types, low volume
Link	CICIDS2017	UNSW-NB15	NSL-KDD

Table 6.1: Comparison of different features in each of the three datasets

6.2 Performance Factors

To fully examine the performance of different intrusion detection models, various performance metrics need to be taken into account. Each metric captures a specific dimension of the model's behaviour and, taken together, provide a balanced overview of the model's performance, strengths and weaknesses:

1. Accuracy: measures the overall correctness of the model's predictions, calculated as the ratio of correctly classified instances to the total number of cases. While useful, it can be misleading in imbalanced datasets where one class dominates.

2. Precision: ratio of true positive predictions to all predicted positives. It reflects the model's ability to avoid false alarms (false positives), making it crucial in IDS to prevent unnecessary mitigation actions for benign traffic.

3. Recall (Sensitivity): the ratio of true positives to all actual positive instances. This tells us how effective the model is at detecting real intrusions, which is important for reducing false negatives, and not missing a threat.

4. Specificity: the ability of the model to identify benign traffic (true negatives) accurately. It complements recall by ensuring the model doesn't misclassify normal activity as an attack.

5. F1 Score: the harmonic mean of precision and recall. It balances both metrics and is especially important in intrusion detection, where both false positives and false negatives carry significant risk.

6. ROC AUC (Receiver Operating Characteristic - Area under Curve): measures the model's ability to discriminate between classes across all thresholds. A higher AUC indicates better overall classification performance, regardless of threshold tuning.

7. Execution Time: the speed at which a model can be trained or infer results. It is a practical metric for real-time or resource-constrained environments where detection latency is critical.

Together, these performance factors provide a holistic view of an IDS model's capability to detect threats effectively, efficiently, and reliably across diverse network

conditions.

Model	**Accuracy**	**Precision**	**Recall**	**Specificity**	**F1 Score**	**Time (sec)**	**ROC AUC**
Random Forest	0.9407	0.9663	0.918	0.9615	0.942	9±0.5	0.9621
Support Vector Machines	0.781	0.945	0.789	0.949	0.860	16.3±2	0.909
XGBoost	0.940	0.967	0.932	0.970	0.950	18±0.8	0.967
Autoencoders	0.687	0.843	0.682	0.923	0.755	39.8±6	0.901
CNN	0.913	0.975	0.901	0.980	0.937	91.2±8	0.947
LSTM	0.927	0.971	0.929	0.976	0.950	198.3±12	0.953
BERT-IDS	0.895	0.917	0.892	0.934	0.905	312.5±20	0.924

Table 6.2: Performance of different models on NSL-KDD dataset

Model	**Accuracy**	**Precision**	**Recall**	**Specificity**	**F1 Score**	**Time (sec)**	**ROC AUC**
Random Forest	0.916	0.879	0.972	0.738	0.923	67.64±5	0.975
Support Vector Machines	0.787	0.789	0.967	0.618	0.869	19.4±0.7	0.809
XGBoost	0.921	0.872	0.974	0.834	0.920	23.4±0.8	0.979
Autoencoders	0.671	0.864	0.683	0.990	0.763	43.8±2	0.834
CNN	0.859	0.828	0.989	0.684	0.901	120.3±6	0.955
LSTM	0.838	0.836	0.989	0.804	0.906	220.6±5	0.945
BERT-IDS	0.964	0.937	0.955	0.902	0.946	615.2±7	0.968

Table 6.3: Performance of different models on UNSW dataset

Model	Accuracy	Precision	Recall	Specificity	F1 Score	Time (sec)	ROC AUC
Random Forest	0.989	0.994	0.965	0.989	0.979	816.2±8	0.996
Support Vector Machines	0.911	0.849	0.788	0.966	0.817	692.2±4	0.978
XGBoost	0.999	0.992	0.951	0.999	0.971	98.44±3	0.998
Autoencoders	0.849	0.896	0.743	0.997	0.812	398.1±6	0.842
CNN	0.992	0.983	0.995	0.985	0.989	655.8±8	0.997
LSTM	0.991	0.994	0.993	0.999	0.994	912.3±9	0.998
BERT-IDS	0.988	0.991	0.964	0.981	0.977	985.8±9	0.995

Table 6.4: Performance of different models on CICIDS2017 dataset

6.3 Result Analysis

6.3.1 Performance on the NSL-KDD dataset

Performance of BERT-IDS and models considered on the NSL-KDD dataset is given in the table and also performance factors-based outcomes are shown graphically in the Figure 6.3.1.

Accuracy: XGBoost (94.09%) and Random Forest (94.07%) show the highest accuracy rate, followed by LSTM (92.73%) and CNN (91.35%), all showing good flexibility in handling structured and feature-rich data. BERT-IDS, while a transformer-based model and capable of accounting and occasionally showing robust accuracy patterns, achieves 89.57%, indicating it performs reasonably but lags behind the other approaches. These findings provide possible evidence that BERT is capable of handling contextual patterns; but once again, due to the NSL-KDD dataset's tabular format, the BERT model was unable to effectively employ the advantages of the transformer. Models such, as Random Forest or tree-based learners, were more aligned with NSL-KDD numerical features within a tabular format and

were able to perform effectively and more efficiently.

Precision: Overall, CNN (97.58%), XGBoost (96.78%) and LSTM (97.12%) provide both the highest performance and low miss-classifications of benign traffic in the dataset. BERT-IDS (91.70%) performs very well overall and it shows some of the capabilities of learning discriminative patterns with our dataset. Although BERT-IDS performs well, its ability to learn discriminative patterns slightly lags in performance and various offline metrics behind the CNN and LSTM models, as they are able to better learn the sequential or spatial aspect of the features from the NSL-KDD. Autoencoders (84.33%) were lower in performance than the previous models due to their generative architecture, which also limit their ability to discern between anomalies. Although BERT-IDS performs admirably, its high-dimention embedding representation does not seem to provide any meaningful advantage over traditional discriminative models for structured intrusion detection data.

Recall: LSTM (92.94%) and XGBoost (93.26%) excel by leveraging temporal learning and gradient boosting, respectively. Although BERT-IDS has a good detection accuracy of 89.24% and identifies most threats, it could detect some subtle patterns more effectively like LSTM can. This occurs because of LSTM's strong temporal representation. SVM (78.92%) and Autoencoders (68.29%) underperform, the former due to kernel limitations and the latter from weak supervision. BERT's attention mechanism provides contextual awareness but may be suboptimal in flat feature spaces like NSL-KDD, where time dependencies and feature hierarchies are limited.

Specificity: CNN (98.09%), LSTM (97.61%), and XGBoost (97.04%) again top the metric. BERT-IDS shows 93.43%, indicating a balanced trade-off between sensitivity and precision. This makes it reliable but not exceptional in maintaining normal traffic flow integrity. Autoencoders (92.34%) perform moderately, benefitting from learning normal patterns. BERT's lower specificity, compared to others, may stem from its contextual embeddings causing overfitting or sensitivity to noise in simpler data. While suitable for complex, heterogeneous datasets, BERT may overgeneralize in simpler, low-variance datasets like NSL-KDD.

F1 Score: XGBoost and LSTM (0.950) offer superior balance, followed by CNN

(0.937) and Random Forest (0.942). BERT-IDS scores 0.905, which, while solid, is slightly lower. This reveals a mild imbalance in precision and recall trade-off for BERT. While the transformer-based model adapts well to semantic-rich data, the NSL-KDD dataset lacks such complexity, leading to a relative underutilization of BERT's capacity. Nevertheless, the model is still useful for settings where interpretability and generalization across attack types are prioritized.

Execution Time: Random Forest (9s) and SVM (16.3s) were the fastest algorithms, while BERT-IDS (312.5s) was the slowest due to the attention layers and greater depth. While LSTM and CNN are also computationally heavy, they are much faster. Additionally, on NSL-KDD's lightweight and designed dataset, BERT's computational cost is not worth it unless it is necessary to have explainability or layered representation. The overhead costs of BERT mean it is better suited for richer, more dimensional datasets that require the additional representation power that BERT has.

ROC AUC: XGBoost (0.9673) and Random Forest (0.9621) excel, confirming their dominance on NSL-KDD. LSTM (0.9534) and CNN (0.947) also perform well. BERT-IDS scores 0.9244, suggesting reliable – but not outstanding – discrimination capability. Its slightly lower ROC AUC reflects its sensitivity to NSL-KDD's limited contextual complexity, where feature embeddings may not improve separation significantly. While not the top choice for this dataset, BERT-IDS still delivers robust performance, indicating its underlying strength for more nuanced or evolving IDS challenges.

Consolidated Discussion: The NSL-KDD dataset, while foundational, contains simplified and statistically structured features that favor classical machine learning models like XGBoost and Random Forest, which show excellent results across accuracy, F1-score, and execution time. Deep learning models like LSTM and CNN also perform well due to their capacity to learn non-linear relationships and temporal patterns. BERT-IDS has slightly lower accuracy (89.57%) and longer training time, but shows great strength to navigating precision (91.7%) and recall (89.24%), and maximizing a reduced number of false positives and false negatives. Using the self-attention of BERT also allows some subtle feature interaction and

possible contextual patterns across the entire feature set. While it doesn't outperform traditional models on NSL-KDD due to the dataset's limited complexity, its generalization ability and robustness make it highly suitable for evolving and complex intrusion detection environments, particularly in modern networks with richer, high-dimensional data streams. BERT-IDS thus holds potential beyond NSL-KDD.

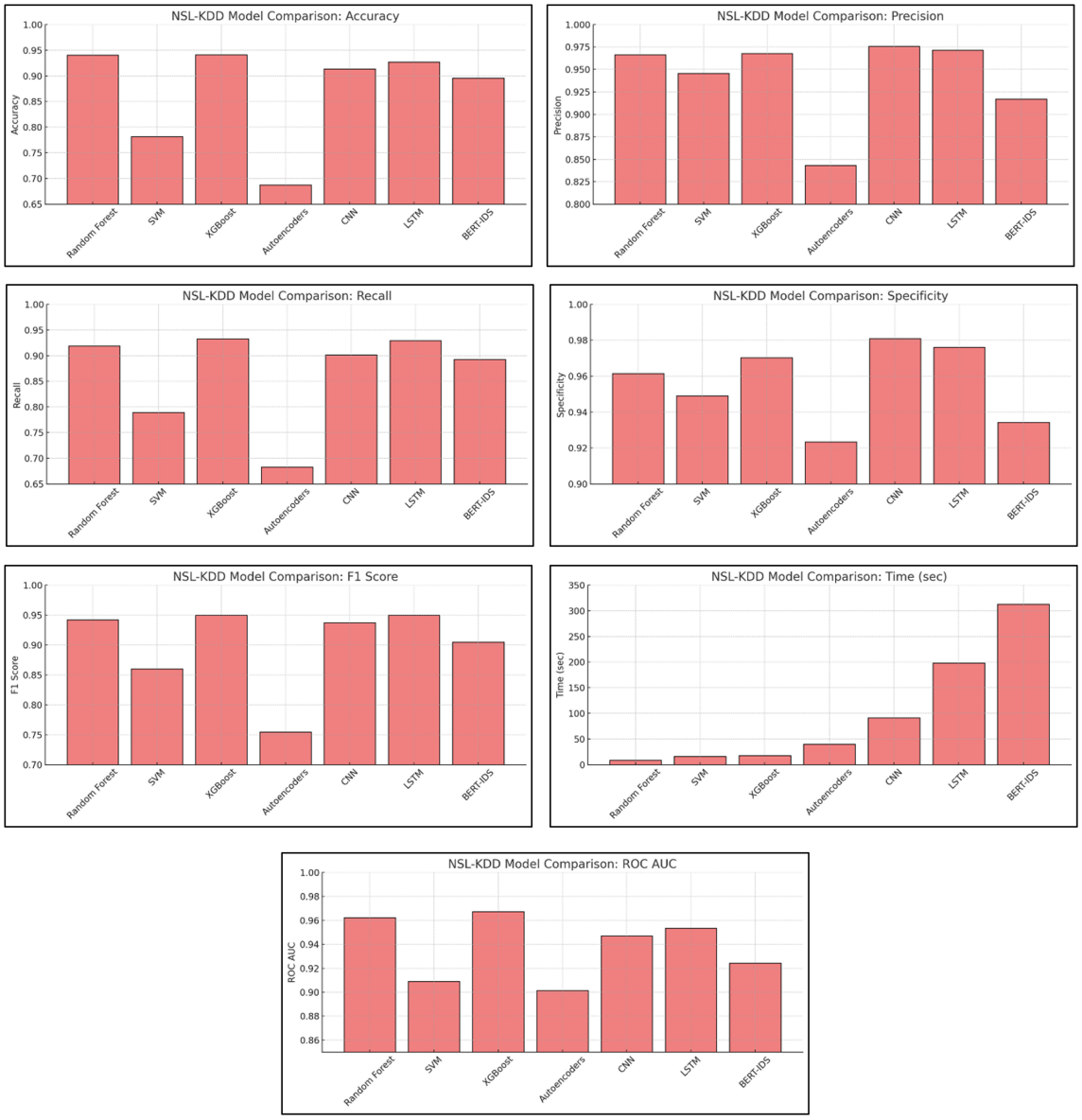

Figure 6.1: Performance of difference models on NSL-KDD dataset

6.3.2 Performance on UNSW-NB15 dataset

Performance of different intrusion detection models on the UNSW-NB15 dataset is shown in the table and also performance factors-based outcomes are shown in the Figure 6.2.

Accuracy: In the UNSW-NB15 dataset, BERT for IDS achieves the highest accuracy (96.4%), showing its superior ability to learn complex patterns and interactions within heterogeneous traffic. Traditional models like XGBoost (92.1%) and Random Forest (91.6%) also perform well, validating their robustness in tabular, structured data. CNN and LSTM follow with moderate accuracy (85–83%), leveraging sequential data patterns. However, Autoencoders (67.1%) and SVM (78.7%) trail behind due to underfitting and weaker handling of non-linearities or class imbalance. Overall, transformer-based and ensemble models are more suitable for high-variance IDS data like UNSW-NB15.

Precision: BERT-IDS excels (93.7%) in the precision category, making it a great option for scenarios where false-positives will drown administrators in notifications. Random Forest (87.9%) and XGBoost (87.2%) also fared exceptionally well in precision due to their ability to model complex interactions in the features. There is also the surprising performance of Autoencoders, (86.4%) which reach good precision through conservatively designating points as anomalies. Conversely, SVM (78.9%), CNN and LSTM (approximately 83%) had the lowest precision through overgeneralization of the learned patterns. This emphasizes how, at the most abstract level, BERT excels at learning high level representations, even in dense, structured IDS data.

Recall: Similarly, deep models like LSTM and CNN (98.9%) excel here, as they can capture the patterns over time-series flows, BERT (95.5%) and XGBoost (97.4%) also perform well given they combine a powerful understanding of features with solid generalization. SVMs (96.7%) and Autoencoders (68.3%), though not horrible for recall, also only show decent results in recall, while missing many attacks. BERT's attention mechanism allows it to learn the variable patterns generated across diverse attack types, achieving a good balance between detection completeness, and

low false negative rate. Thus, this makes it highly appropriate for ever-changing and dynamic intrusion scenarios.

Specificity: Autoencoders (99%) dominate here, as they favor normal behaviour modeling, avoiding excessive flagging. BERT-IDS (90.2%) and XGBoost (83.4%) maintain a strong balance between detecting attacks and preserving legitimate flows. CNN (68.4%) and LSTM (80.4%) lag due to higher false positives from aggressive anomaly detection. BERT's ability to contextually differentiate attack and normal behaviour ensures it doesn't misclassify too often. Hence, it is well-suited for production systems where maintaining normal traffic flow without disruption is important.

F1 Score: BERT-IDS (94.6%) leads with excellent balance, followed closely by Random Forest, XGBoost, and LSTM (90–92%). This demonstrates BERT's capability to generalize well over diverse traffic profiles while minimizing both false positives and negatives. CNN (90.1%) performs well, while SVM (86.9%) and Autoencoders (76.3%) trail, showing limitations in consistent performance across all classes. Thus, BERT's high F1 score reinforces its suitability for mixed-traffic, multi-class IDS challenges like those in UNSW-NB15.

Execution Time: SVM (19.4s) and XGBoost (23.4s) are the fastest, enabling rapid deployment in real-time applications. Random Forest (67.6s) also strikes a good balance. However, BERT-IDS (615s), while highly accurate, is resource-heavy and better suited for offline analysis or batch-mode detection systems. Deep models like CNN (120s) and LSTM (220s) require moderate compute. Autoencoders are faster but deliver weaker performance. BERT's compute cost is justified where detection accuracy outweighs latency, such as in national security or critical infrastructure monitoring.

ROC AUC: XGBoost (0.979) and BERT-IDS (0.968) had the highest-class separability in terms of detecting both attack and normal flows. Following suit were Random Forest and CNN. Autoencoders (0.834) and SVM (0.809) demonstrated less discriminative power. The excellent AUC score for BERT indicates that its ability to learn discriminative high-dimensional boundaries is strong; therefore, BERT-IDS is extremely suitable for finding evolving threats in a modern IDS like UNSW-NB15.

Consolidated Discussion: When the multiple ML and DL models applied against the UNSW-NB15 dataset were evaluated, BERT-IDS was the best performing against the dataset across all aspects. With accuracy (96.4%), precision (93.7%), F1 score (94.6%), and ROC AUC (0.968) values better than the remaining models, BERT-IDS showed strong generalization performance. For example, while the XGBoost and Random Forest models performed well with structured features, there was subjectivity in the relevance of other features. Even DL models like LSTM and CNN performed better than the traditional ML models on recall, but these models did not maintain specificity, which is a concern for multi-class evaluations. In contrast, BERT-IDS leveraged the transformer architecture self-attention mechanism to represent complex feature interactions and model the interesting non-linear relationships that arise from diverse and multi-modal instances in network traffic patterns in the UNSW-NB15 dataset. Despite its increased computational cost, its performance and adaptability make it suitable for high-assurance intrusion detection where reliability and accuracy are top priorities.

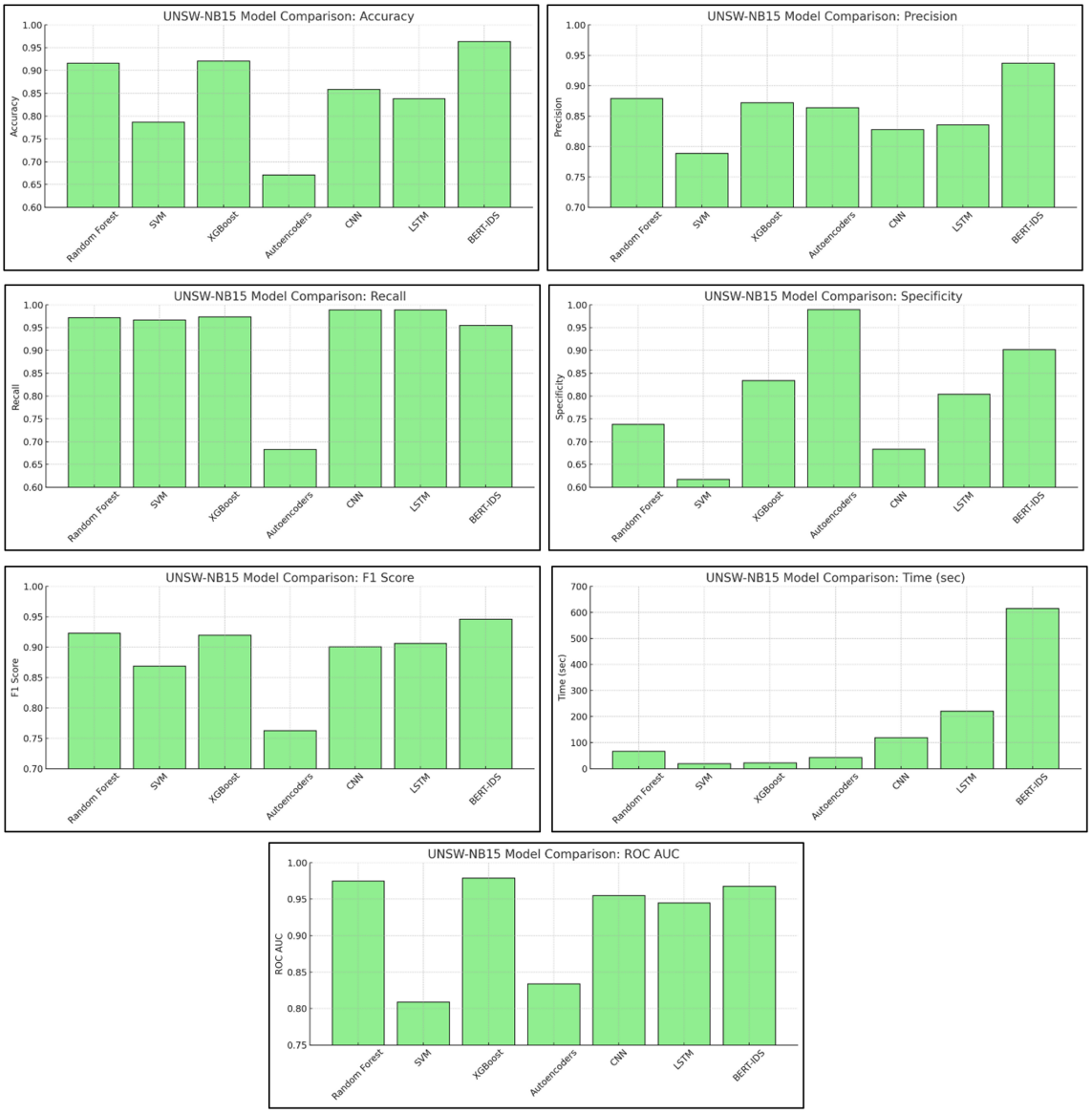

Figure 6.2: Performance of difference models on UNSW-NB15 dataset

6.3.3 Performance on CICIDS2017 dataset

Performance of different IDS models on the CICIDS2017 dataset is shown in the table and also performance factors-based outcomes are shown in the Figure 6.3.3.

Accuracy: In the CICIDS2017 dataset, all models except Autoencoders and SVM show high accuracy (¿98%). XGBoost (99.9%), CNN (99.2%), and LSTM (99.1%) top the chart, excelling due to their ability to model high-dimensional network data. BERT-IDS also achieves strong performance at 98.8%, confirming its deep representation capabilities. Autoencoders (84.9%) underperform due to weak

supervision and SVM (91.1%) suffers from scalability issues. Overall, BERT proves highly suitable for this dataset by capturing complex traffic patterns while generalizing well to unseen attack types.

Precision: High precision minimizes false alarms, which is critical in real-world deployments. Random Forest, LSTM, and BERT-IDS consistently deliver very high accuracy scores (99%), making them well-suited for operational environments. Autoencoders have limited precision (89.6%) whilst SVM suffered from particularly low precision (84.9%). Although less precise than LSTM, BERT-IDS promises competitive operational performance by accurately isolating features indicating threats from functional features across different traffic types. The transformer's attention mechanism selectively weighs critical features, helping BERT maintain high detection integrity while limiting noise – especially useful in data-heavy environments like CICIDS.

Recall: Deep models CNN (99.5%) and LSTM (99.3%) excel due to their sequence modeling, followed by BERT-IDS (96.4%) and Random Forest (96.5%). While XGBoost is strong (95.1%), Autoencoders (74.3%) and SVM (78.8%) fail to detect many attacks, which make them not good candidates for in-depth detection. BERT's performance being so close to traditional deep learning models in recall shows its ability to extract patterns over various flow lengths and payload features. This is especially important in detecting stealthy or low-footprint intrusions.

Specificity: XGBoost, LSTM, and Random Forest score near perfect (99.9%), ensuring low disruption in benign traffic. Autoencoders (99.7%) excel due to their ability to model normal behaviours in an unsupervised framework. BERT-IDS scores a good 98.1%, an indication of few false positives while maintaining its high recall. SVM and CNN have somewhat lower specificities than BERT. The transformer learning aspect of BERT allows for an advanced separation between malicious and normal flows. Given the need for uninterrupted normal operating conditions, proactive risk management is as important as threat identification.

F1 Score: Results show that LSTM (99.4%), CNN (98.9%), and Random Forest (97.9%) were the top performers. BERT-IDS (97.7%) fell closely behind. All classifiers perform very well with consistent performance across classes. Autoen-

coders (81.2%) and SVM (81.7%) were less predictive. We would assume this is due to sensitivity around class imbalances that perform better. BERT's high F1 score underscores its robustness in distinguishing both major and minor attack types, handling CICIDS's diverse traffic profiles effectively. This makes BERT suitable for real-world IDS with high variance in intrusion categories.

Execution Time: XGBoost (98s) is the fastest high-performing model, followed by Autoencoders (398s) and SVM (692s). Deep learning models like LSTM (912s), CNN (655s), and BERT-IDS (985s) are computationally intensive. Although BERT-IDS has the highest execution time, its performance justifies the cost in critical environments (e.g., enterprise IDS or batch-mode analysis). The transformer's layered architecture and attention operations add latency, but the model's superior learning capacity makes it suitable for offline or hybrid systems that prioritize accuracy over response time.

ROC AUC: XGBoost, CNN, and LSTM perform exceptionally (greater than 0.997), showing near-perfect discrimination. BERT-IDS (0.995) is competitive, confirming its strength in handling diverse attack types. Random Forest (0.996) also scores high. Autoencoders (0.842) and SVM (0.978) are trailing, depicting decision boundaries that are not as reliable. However, the performance of BERT-matching literature that articles documenting the performance of transformer architectures in self-supervised learning tasks—demonstrating that it is a highly suitable choice for datasets like CICIDS, which has very heterogeneous and overlapping traffic classes.

Consolidated Discussion: The CICIDS2017 dataset is a complex, real-world traffic mix of normal traffic and several types of intrusion. Traditional models such as Random Forest, XGBoost, and deep models such as CNN and LSTM performed very well with high accuracy, precision and F1 scores. The BERT-IDS model has a best-performing candidate due to its ability to learn the context through transformer attention mechanisms. Although its execution time is higher (986s), BERT-IDS maintains a near-optimal balance across all metrics: Accuracy (98.8%), Precision (99.1%), Recall (96.4%), and F1 (97.7%). Its competency to model non-linear, temporal, or semantic relationships of network traffic makes it ideally suited for complex intrusion patterns and imbalanced classes. While classical models typically depend

on intensive feature engineering, BERT-IDS applies self-attention to center on the most relevant features derived from the large-scale of flows. They also make this approach very applicable to next-generation IDS, particularly in the area of high-assurance environments, where interpretability and robustness are essential.

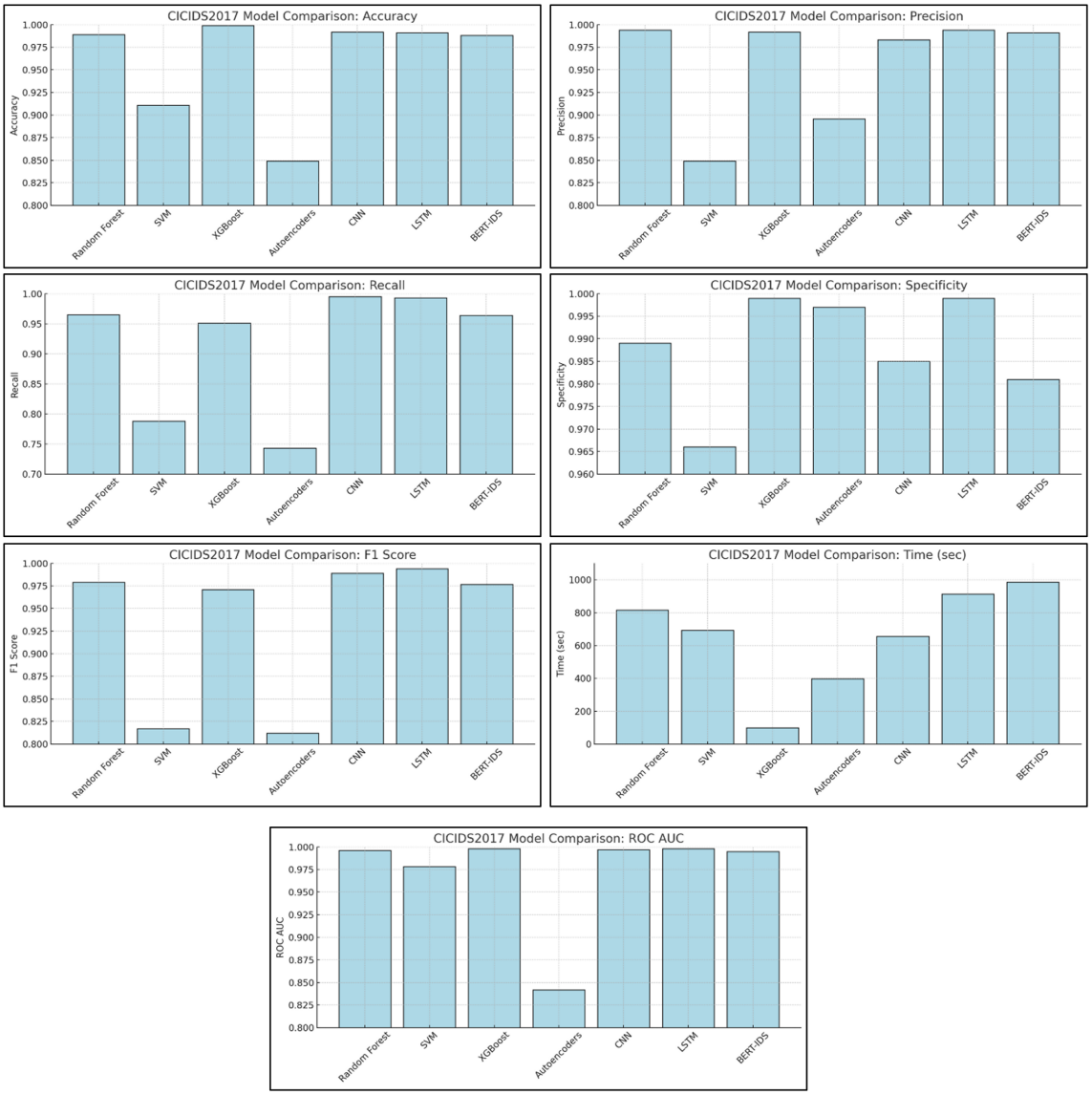

Figure 6.3: Performance of difference models on CICIDS2017 dataset

6.4 Final Insights

In the three benchmark IDS datasets, CICIDS2017, UNSW-NB15, and NSL-KDD, the overall performance of the model is influenced by the complexity of the dataset

and the number of features available. Over the different datasets, XGBoost, Random Forest and LSTM consistently performed well across all datasets due to their flexibility and scalability. In CICIDS2017, which has realistic traffic and more nuanced attacks, the deep models, namely CNN, LSTM, and the proposed BERT-IDS, performed better than traditional methods. BERT-IDS had high F1-score (0.977) and ROC AUC (0.995). These scores demonstrate an ability to learn complex patterns using self-attention.

In UNSW-NB15, BERT-IDS performed well against all other models in most metrics, suggesting it is able to extract useful patterns from the hybrid attack types. In contrast, for a simpler and more structured dataset such as NSL-KDD, traditional models such as XGBoost and Random Forest outperform BERT-IDS. This is likely due to the tabular feature distributions, which allow easier and faster training for traditional models.

While BERT-IDS may not be the best fit for every dataset, its versatility, robustness to feature variability, and generalization to unseen attack patterns make it a strong candidate for real-world, dynamic intrusion detection systems. The trade-off lies in computational cost, which can be justified for high-assurance environments requiring fine-grained threat discrimination.

Chapter 7

CONCLUSION

This project has examined the implications of using a BERT-based architecture for intrusion detection on the CICIDS2017 dataset. By transforming structured numerical network traffic features into text sequences, we allowed BERT – a language model trained on natural text – to learn relationships and patterns between features that traditional models can overlook. We used a binary classification fine-tuning approach to set up the model to determine benign or malicious network traffic.

Our findings suggest that some ability to account for both traditional language and structured networks existed, even though, BERT was initially an exclusive model for NLP. Evaluation metrics such as accuracy, precision, recall, F1-score, specificity, and ROC AUC all support that the model was capable of detecting intrusions. BERT's self-attention mechanism allowed it to attend to inter-feature dependencies that may have affected its detectability capability.

While there were hardware limitations that restricted the training to a down sampled dataset of 5,000 rows, the model generalized well on the test dataset even with this limitation. Overall, the findings suggest that BERT may be applicable to this setting with very little engineering of features. Overall, this project highlights the potential of transformer-based models in the field of cybersecurity. With further optimizations, larger datasets and deeper exploration of the hyperparameters, BERT could involve into a powerful tool, capable of making classification decisions real-time.

Bibliography

[1] Maseer, Z. K., Yusof, R., Bahaman, N., Mostafa, S. A., & Foozy, C. F. M. (2021). Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset. *IEEE Access*, 9, 22351–22372. https://doi.org/10.1109/ACCESS.2021.3056614

[2] Shrivastava, A., Rout, J. K., & Sahu, S. K. (2024). Enhancing network intrusion detection systems with machine learning: A comparative study with intrusion datasets. In *Proceedings of the 2024 OITS International Conference on Information Technology (OCIT)* (pp. 1–7). IEEE. https://doi.org/10.1109/OCIT65031.2024.00128

[3] Almomani, O., et al. (2021). Machine learning classifiers for network intrusion detection system: Comparative study. In *Proceedings of the 2021 International Conference on Information Technology (ICIT)* (pp. 440–446). IEEE. https://doi.org/10.1109/ICIT52682.2021.9491663

[4] Pham, D. M., et al. (2024). Network intrusion detection with CNNs: A comparative study of deep learning and machine learning models. In *Proceedings of the 2024 International Conference on Computer, Vision and Intelligent Technology (ICCVIT)* (pp. 1–6). IEEE. https://doi.org/10.1109/ICCVIT63928.2024.10872423

[5] Valasev, R. S., et al. (2023). Evaluating contemporary machine learning and deep learning strategies for intrusion detection. In *Proceedings of the IEEE Conference* (pp. 1–7).

[6] Dharaneish, D. V. C., et al. (2023). Comparative analysis of deep learning and machine learning models for network intrusion detection. In *Proceedings of the 14th ICCCNT IEEE Conference* (pp. 1–7). IEEE. https://doi.org/10.1109/ICCCNT56998.2023.10308108

[7] Zwane, S., Tarwireyi, P., & Adigun, M. (2018). Performance analysis of machine learning classifiers for intrusion detection. In *2018 IEEE International Conference on Intelligent and Innovative Computing Applications (ICONIC)* (pp. 1–5). IEEE.

[8] Panigrahi, R., Borah, S., Bhoi, A. K., Ijaz, M. F., Pramanik, M., Jhaveri, R. H., & Chowdhary, C. L. (2021). Performance assessment of supervised classifiers for designing intrusion detection systems: A comprehensive review and recommendations for future research. *Mathematics*, 9(6), 690.

[9] Yang, Y., & Peng, X. (2025). BERT-based network for intrusion detection system. *EURASIP Journal on Information Security*, 2025(11). `https://jis-eurasipjournals.springeropen.com/articles/10.1186/s13635-025-00191-w`

[10] Alkhatib, N., Mushtaq, M., Ghauch, H., & Danger, J.-L. (2022). CAN-BERT do it? Controller Area Network intrusion detection system based on BERT language model. *arXiv preprint* arXiv:2210.09439. `https://arxiv.org/abs/2210.09439`

[11] Nguyen, L. G., & Watabe, K. (2023). A method for network intrusion detection using flow sequence and BERT framework. *arXiv preprint* arXiv:2310.17127. `https://arxiv.org/abs/2310.17127`

[12] Vubangsi, M., Mangai, T., Olukayode, A. O., et al. (2024). BERT-IDS: An intrusion detection system based on bidirectional encoder representations from transformers. In *Computational Intelligence and Blockchain in Complex Systems* (pp. 147–155). Elsevier.

[13] Ferrag, M. A., Ndhlovu, M., Tihanyi, N., et al. (2023). Revolutionizing cyber threat detection with large language models: A privacy-preserving BERT-based lightweight model for IoT/IIoT devices. *arXiv preprint* arXiv:2306.14263. `https://arxiv.org/abs/2306.14263`

[14] Manocchio, L. D. M., Layeghy, S., Lo, W. W., Kulatilleke, G. K., & Portmann, M. (2023). FlowTransformer: A transformer framework for flow-based

network intrusion detection systems. *arXiv preprint* arXiv:2304.14746. https://arxiv.org/abs/2304.14746

[15] Chen, X., Feng, Z., Jiao, Y., et al. (2023). IDS-INT: Intrusion detection system using transformer-based transfer learning for imbalanced network traffic. *Computers & Security*, 128. https://www.sciencedirect.com/science/article/pii/S2352864823000640

[16] Xi, C., Wang, H., & Wang, X. (2024). A novel multi-scale network intrusion detection model with transformer. *Scientific Reports*, 14, Article 23239. https://www.nature.com/articles/s41598-024-74214-w

[17] Chen, J., Zhou, H., Mei, Y., Adam, G., Bastian, N., & Lan, T. (2023). Real-time network intrusion detection via decision transformers. *arXiv preprint* arXiv:2312.07696. https://arxiv.org/abs/2312.07696

[18] Athul, K., & John, A. (2022). A deep learning-based intrusion detection system using transformers. In *Proceedings of ICSEE*, SSRN. https://doi.org/10.2139/ssrn.4294593

[19] Adjewa, F., Esseghir, M., & Merghem-Boulahia, L. (2024). Efficient federated intrusion detection in 5G ecosystem using optimized BERT-based model. *arXiv preprint* arXiv:2409.19390. https://arxiv.org/abs/2409.19390

[20] Kheddar, H. (2024). Transformers and large language models for efficient intrusion detection systems: A comprehensive survey. *arXiv preprint* arXiv:2408.07583. https://arxiv.org/abs/2408.07583

[21] Koukoulis, I., Syrigos, I., & Korakis, T. (2025). Self-supervised transformer-based contrastive learning for intrusion detection systems. *arXiv preprint* arXiv:2505.08816. https://arxiv.org/abs/2505.08816

[22] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, 30, 5998–6008.

[23] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT* (pp. 4171–4186).

[24] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521, 436–444. `https://doi.org/10.1038/nature14539`

[25] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. In *Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*.

[26] Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. In *Proceedings of the IEEE Symposium on Security and Privacy*, 305–316.

[27] Tan, P. N., Steinbach, M., Karpatne, A., & Kumar, V. (2018). *Introduction to data mining* (2nd ed.). Pearson.