National College of Ireland

An Enhanced Deep Learning Approach to Detect Cyberattacks and Malware in Cybertext

MSc Research Project
MSc. in Cybersecurity

Aayush Sharma
Student ID: 23422211

School of Computing
National College of Ireland

Supervisor: Arghir Nicolae Moldovan

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Aayush Sharma		
Student ID:	23422211		
Programme:	MSc. in Cybersecurity	**Year:**	2025
Module:	Research Practicum		
Supervisor:	Arghir Nicolae Moldovan		
Submission Due Date:	28-01-2026		
Project Title:	Final Report		
Word Count:	8281	**Page Count:**	24

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Aayush Sharma

Date: 28-01-2026

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

An Enhanced Deep Learning Approach to Detect Cyberattacks and Malware in Cybertext

Aayush Sharma
X23422211

Abstract

Cyberattacks increasingly appear within textual communication such as threat reports, phishing content, incident alerts, and malware descriptions, creating a need for automated systems capable of classifying complex cyber-related language with high reliability. Traditional detection methods and many earlier analytical models often depend on shallow representations or handcrafted features, which can struggle to capture contextual semantics, subtle linguistic cues, and evolving attacker terminology. This study addresses these challenges through a hybrid **ALBERT+LSTM** framework for multiclass cybertext threat classification, where ALBERT provides compact contextual embeddings and an LSTM layer learns sequential dependencies within the encoded text. The overall workflow follows the KDD process (data selection, preprocessing, transformation, data mining, and evaluation) and includes text cleaning and normalization, transformation into model-ready representations, model training, and consistent metric-based comparison against established machine learning and deep learning baselines. Experimental results show that the **ALBERT+LSTM** model achieves **0.90 accuracy** and an **0.90 F1-score**, exceeding the performance of Random Forest, XGBoost, BiLSTM, and GRU under the same evaluation protocol. These findings demonstrate that combining lightweight contextual modelling with sequence learning improves threat-category separation in cybertext and supports robust automated cybersecurity analytics.

1 Introduction

1.1 Problem Background

The use of textual means of communication has been a growing trend and sophistication of cyberattacks through the use of threat reports, phishing emails, malware descriptions, and social-engineering messages to break into digital systems. The dynamic nature of the malicious techniques, and the large amount of text related to cyber-crime that is produced every day, has heightened the necessity of automated systems that would be able to correctly detect and categorise threats. Older methods of detection, especially the signature-based methods are hard to recognise new or obfuscated threats, and rule-based methods cannot generalise outside of the set attack patterns. Consequently, cyber defence settings are becoming more reliant on machine learning and natural language processing methodologies to derive useful patterns out of unstructured cybertext.

1.2 Problem Statement (Motivation)

The available classification systems, including the traditional machine learning models as well as the deep learning systems, have shown different degrees of effectiveness in detecting cybertext threats. Machine learning models are not usually based on handcrafted features and

do not have the semantic depth needed to understand complex threat stories. Recurrent neural networks e.g. LSTM, Bi-LSTM, GRU are more effective at sequence modelling, but are sensitive to the quality of initial embeddings and have problems with long-range contextual dependencies. Transformer-based models, including BERT, provide superior contextual representation but have a high computational cost, which restricts their use in a wider operational context. These difficulties demonstrate the necessity of a method that will be semantically rich, computationally efficient, and sequentially comprehending.

On these grounds, the study examines the efficacy of an improved hybrid architecture-ALBERT with LSTM to detect a cyberattack and malware in textual data. The research questions are to evaluate several baseline models, create contextual embeddings with the help of ALBERT, and apply sequential learning based on LSTM layers and to decide whether the hybrid architecture is superior to standalone models in terms of the predetermined evaluation metrics.

1.3 Research Question

The research question guiding this study is as follows:

"How effectively can an ALBERT+LSTM hybrid model detect and classify cyberattacks and malware within cyber-related text compared to traditional machine learning and deep learning approaches?"

Justification:

This question is motivated by the need to capture both contextual meaning and sequential patterns in cybertext while keeping computation feasible. Classical ML relies on handcrafted features, recurrent models may miss deeper context, and full transformer systems can be costly. A lightweight transformer (ALBERT) combined with LSTM may provide a practical balance for accurate cybertext threat classification.

1.4 Research Objectives

This project aims to:

- Build a cybertext classification pipeline for detecting and categorizing cyberattacks/malware-related entities in text.
- Implement and evaluate baseline **machine learning** models (Random Forest, XGBoost) for comparison.
- Implement and evaluate **deep learning** sequence models (BiLSTM, GRU) for comparison.
- Design and implement the **ALBERT + LSTM hybrid** model to combine contextual embeddings with sequential learning.

- Compare all models using consistent evaluation metrics (accuracy, precision, recall, F1-score, confusion matrix).
- Identify whether the hybrid model improves detection reliability across threat categories.

1.5 Contributions

The main contribution of this paper is the creation and testing of a hybrid ALBERT+LSTM classifier based on the strengths of contextualisation of transformer-based embeddings and serial learning abilities of the recurrent neural networks. It has been shown that contextual models and deep recurrent architectures are effective in context-sensitive classification of texts related to cybercrime, including Security BERT to identify IoT threats (Ferrag et al., 2024), BiGRU-based phishing detection (Benavides-Astudillo et al., 2023), and LSTM-based intrusion analysis (Bibi et al., 2023). This study presents and evaluates a hybrid contextual–sequential architecture (ALBERT+LSTM) for cybertext threat classification, aiming to balance transformer-level semantic richness with efficient sequence modelling.

2 Related Work

The literature review demonstrates the shortcomings of the traditional machine learning, independent deep learning and heavy transformer models in processing complicated text related to cyber-related text. The gaps create the necessity of a more efficient and context sensitive architecture, which forms the foundation of the methodology in the following section.

2.1 Machine Learning Models for Cyberattack and Malware Detection

Classical machine learning (ML) has long dominated automated cyber-text analysis and cyberattack detection due to their relative simplicity to implement, do not require much time to train, and frequently obtain competitive results when combined with appropriate feature representations. Popular classification models are Support Vector Machines (SVM), Naive Bayes, Logistic Regression, K-Nearest Neighbours (KNN) and Decision Trees. Indicatively, Akhtar & Feng, (2022) compared different ML models, such as SVM, J48, Naive Bayes, and Random Forest, in malware detection and found that Decision Trees and SVM gave high detection accuracy of over 99 percent. Their results show that ML is able to identify patterns of polymorphic malware based on statistical properties, and also it faces problems with high-dimensional textual features and adversarial obfuscation. Equally, Alsufyani & Alzahrani (2021) have constructed a phishing detection system based on Word2Vec-CBOW embeddings and ML classifiers including KNN, Multinomial Naive Bayes, Decision Trees and AdaBoost, demonstrating that ML is still useful in lightweight cybertext classification.

In addition to the threat detection field, the use of ML has been extended to other text-security fields, including cyberbullying and SMS spam detection. Ramamoorthy et al. (2024) have shown that TF-IDF with SVM can be useful in capturing lingo-cues of cyberbullying in cross-demographic settings, which supports the usefulness of simple vectorisation in doing

feasible classification problems. According to Ahmadi et al. (2025), TF-IDF combined with Naive Bayes or SVM is capable of reaching over 96% accuracy in spam detection in SMS, and thus, it can be confirmed that classical ML remains relevant in security-related text analytics. Nevertheless, their research also points out one important sensitivity: performance can be highly different across different choice of feature extraction method, which means that it is not always the classifier itself but more often the choice of representation that determines the success of the model.

Although the classical ML methods are performing well on baseline, they have significant limitations in the contemporary cybertext setting. Most of the techniques are based on sparse and high-dimensional representations, which can decrease robustness and lead to performance that greatly depends on pre-processing and manually designed features. The simple Logistic Regression can also have problems with nonlinear trends and complex threat behaviours especially when dimensionality is high (Brightwood & Anthomy, 2024). XGBoost and other tree-ensemble models might offer a better recall in high-risk detection scenarios (Atoum et al., 2025), but are highly limited by engineered features and do not necessarily reflect additional contextual sense. In a more general sense, classic ML has low contextual semantic learning, long-range dependencies, and changing adversarial language patterns, which is problematic when attackers alter wording, add obfuscation, or change countermeasures over time.

2.2 Deep Learning Models for Cybertext Threat Detection

Deep Learning (DL) has made a tremendous leap in cyberattack detection by surpassing the use of handcrafted features used by ML and providing more detailed semantic extraction. CNNs and RNNs have come to dominate the scene of text-related cybersecurity. According to , CNNs achieved high malware detection accuracy (98.76%), which is attributed to the ability of DL to learn features automatically when applied to security datasets. Coyac-Torres et al. (2023) also introduced a CNN-based system of cyberattack detection that recognizes the phishing, malware, spam, and bot attacks in social-network messages with 0.91 detection accuracy. They focused on source independent analysis, which confirmed the capability of CNNs to cross platform.

Recurrent models like LSTM, BiLSTM and GRU have also performed even better because they can capture sequential semantic dependencies. Bibi et al. (2023) showed that LSTM models are superior to multiple DL and ML classifiers in intrusion detection on such datasets as NSL-KDD and CICIDS2017 with high accuracy rates of 99%. Similarly, Benavides-Astudillo et al. (2023) have applied GloVe embeddings to different RNN-based models and identified BiGRU as the most effective model in detecting phishing-text (97.39%). All these findings point to the fact that DL is better at processing context-rich cybertext.

Hybrid DL architectures have also been introduced to further increase the ability. Albraikan et al. (2022) built a model of ODL-CDC that integrates BiGRNN with the hyperparameter optimization system based on Search and Rescue Optimization (SRO) with the accuracy of

92.45 in cyberbullying detection. Saheed and Arowolo (2021) used Deep Recurrent Neural Networks (DRNN) and bio-inspired optimization of features to reach 99.76 percent accuracy in detecting IoMT cyberattack. These works demonstrate the superiority of DL over the classical ML, particularly in the context of learning sophisticated behavioural and linguistic patterns of cyberattacks. Nevertheless, the existing versions of DL models continue to experience computational overhead, memory inefficiencies, and scalability challenges, particularly when the systems are deployed under resource limited real-life conditions.

2.3 Transformer-Based and Other High-Capacity Models

More sophisticated architectures like Inception-V4-XGBoost hybrids, DRNNs, and Transformer-based are also being used to analyze cyber-text. As an example, (Raghunath et al., 2022) presented an XGBoost Regression Classifier with Inception V4 to detect cyberattacks with a detection accuracy of over 97 percent in a number of test scenarios. Although effective, such models are computationally expensive as they have huge feature representations and complicated network topologies. Equally, the SEAM framework suggested by Ramoliya et al. (2023) utilized LSTM, GRU, and RNN variations to classify secure messages in the autonomous EV communication. Though successful, these models are computationally intensive and are not as well suited to be deployed to lightweight inference environments.

The latest development in cyber-threat detection is transformer-based models, specifically the BERT versions. Ferrag et al. (2024) presented a BERT-based IoT threat detection architecture named Security BERT that results in an 98.2% accuracy with privacy-preserving encoding. BERT-based models have been known to be memory intensive, computationally expensive, and slow when performing real time inferences despite their performance. Security BERT still consumes 16.7 MB with compression, which is a lightweight but still heavy in low power devices. In addition, the costs of training transformers are expensive, and large numbers of their parameters make their use on edge devices difficult, particularly in comparison with smaller language models.

Despite the accuracy of high-capacity models being better than traditional DL and ML, their inefficiency, large memory footprint, and dependence on large training corpora make high-capacity models impractical. Such limitations underscore the importance of lightweight, efficient architectures that do not affect contextual learning capabilities, which promotes the use of ALBERT-based hybrid models.

Table 1. Comparative Review of Other Approaches

No.	Author(s) & Year	Dataset Used	Model Used	Accuracies	Contributions	Limitations
1	Ramoliya et al., 2023	Autonomous EV communication data	LSTM, GRU, RNN (SEAM framework)	85 %	Deep learning-based secure message exchange for autonomous EVs.	Not benchmarked on standard IDS datasets; lacks attack-specific metrics.
2	Alsufyani & Alzahrani, 2021	6,224 email samples	KNN, DT, AdaBoost, MNB	95%	NLP-based phishing detection using Word2Vec features.	Limited dataset; imbalance affects performance; less effective for complex phishing.
3	Raj et al., 2021	2 real-world cyberbullying datasets	Bi-GRU, Bi-LSTM, Attention + ML baselines	95% accuracy, 98% F1 Score	Compared 11 classifiers and 7 feature methods; Bi-directional models excel.	Domain-specific focus; struggles with multimodal data; high training cost.
4	Raghunath et al., 2022	CSE-CIC-IDS2018, EMBER, UHN	XGBoost Regression + Inception V4	97%	Hybrid model reduces error rates and supports multi-attack classification.	Heavy computational load; reduced accuracy on complex attack types.
5	Coyac-Torres et al., 2023	Social network messages	CNN + NLP	0.91 (detection), 0.82 (classification)	Detects and classifies 4 types of cyberattacks from text messages.	Accuracy lower than deep hybrid models; limited to textual features.
6	Benavides-Astudillo et al., 2023	Phishing website text	LSTM, BiLSTM, GRU, BiGRU	Up to 97.39%	NLP-based phishing detection using GloVe embeddings.	Computationally heavy; dataset-specific; lacks zero-day evaluation.
7	Srivastava et al., 2021	Cyber risk forecasting data	Predictive Analytics (ML) + Random Forest, LSTM, XGBoost, SVM	Random Forest: 89% SVM: 79%	Explores feature engineering, interpretability, and data quality issues.	Assumes linearity; poor with non-linear data; less accurate than DL.
8	Ramamoorthy et al., 2024	Social media text data	TF-IDF + SVM	94%	TF-IDF converts text for SVM classification in cyberbullying detection.	Limited semantic understanding; lower accuracy; sensitive to pre-processing.

9	Ferrag et al., 2024	Edge-IIoTset	Privacy-preserving BERT (SecurityBERT)	98.2%	Lightweight BERT surpasses ML/DL models; efficient for IoT devices.	Higher memory needs; limited field validation; dataset dependence.
10	Brightwood & Anthomy, 2024	Cyber threat logs	Logistic Regression	94%	Discusses practical deployment challenges and feature selection.	Limited for evolving threats; struggles with high-dimensional data.

Table 1 provides a comparative overview of the previous research including the models used, their contribution, and the limitation found in the current methods. This summary shows the weaknesses in performance and methodological limitations of the existing research in analysis of cybertext, further supporting the necessity of a much more efficient and context-sensitive hybrid algorithm like ALBERT+LSTM.

2.4 Research Gaps

Although promising results have been obtained by ML, DL, and Transformer-based methods, various research gaps are still apparent in the literature. ML models are based on manual features, do not include more semantically profound dependencies, and suffer when faced with changing adversarial text (Ferrag et al., 2024). However, traditional DL models such as CNNs and LSTMs are effective but have been found to be characterized by high computational costs, slow training, and scalability when handling large-scale or real-time streams of cybertext (Bibi et al., 2023; Coyac-Torres et al., 2023). Transformer-based models such as BERT and Security BERT, although leading in terms of performance, use large memory, high hardware requirements and long inference times, which makes them inappropriate to deploy on limited or high throughput systems (Ferrag et al., 2024). Also, there is a lack of research on malware detection in cybertext where subtle contextual hints are critical but commonly neglected in previous architectures.

The other acute gap is that it has not sufficiently explored hybrid lightweight models that can effectively extract global contextual and sequential patterns of a cyberattack text in the form of a text. Current models either are context-based (Transformers) or sequence-dependent (LSTM/GRU), and rarely combine the two in a computationally efficient way. Furthermore, the accuracy of optimization methods applied in previous literature is higher, but the model becomes more complex, which decreases its applicability to real-time threat detection systems.

2.5 Addressing the Gaps Through an ALBERT + LSTM Hybrid Model

To overcome those shortcomings, the current study suggests a new hybrid architecture, which is ALBERT (A Lite BERT) with LSTM, that uses the advantages of both the models and reduces the drawbacks of the existing studies. The size of the model is reduced by ALBERT

through parameter sharing, factorized embeddings, and reduction of cross-layer weights which means that the memory cost is significantly reduced, and that training ALBERT takes significantly less time than full-sized versions of BERT models-essentially a direct response to the computational overhead that was emphasized in SecurityBERT and other Transformer-based publications. By combining ALBERT with LSTM, the model improves semantic comprehension by embedding the context and at the same time retrieving long-range temporal connections critical in malware and cyberattack detection in cybertext.

This hybrid architecture helps to resolve Transformer inefficiency due to its ability to retain contextual richness with minimal model latency, which can be applied to address gaps observed in the research like Ferrag et al. (2024). Furthermore, attention-based context extraction of ALBERT combined with sequential modelling of LSTM is a more effective way to capture linguistic and behavioural patterns of cyberattacks as compared to independent CNN, GRU, or XGBoost-based methods (Benavides-Astudillo et al., 2023; Bibi et al., 2023; Raghunath et al., 2022). In general, the ALBERT + LSTM solution directly addresses identified disadvantages connected to computational inefficiency, lack of context, reliance on feature-engineering, and scalability issues, which makes it a very powerful tool to tackle the problem of cyberattack and malware detection in cybertext.

3 Research Methodology

3.1 Methodology Framework

This research is based on the Knowledge Discovery in Databases (KDD) process to organize the cybertext threat classification process. This is an automated process that starts with the selection of data, which is a text on cyber-related topics concerning the types of cyberattacks and malwares. Then, the textual data undergoes pre-processing to clean and normalize the textual content, then, transformation, which converts the processed text into model-friendly representations like vectorised features or embeddings. The data mining phase trains various classification models, both classical machine learning and neural network based, to detect threat categories that occur in cybertext. Lastly, evaluation measures model effectiveness by standard classification measures in order to provide a consistent and reliable comparison of all models.

3.2 Methodology Novelty and Contribution

In spite of the numerous current researches that are either based on classical machine learning and handcrafted features or relying on entire transformer-based frameworks, which can be expensive to compute, the approach of the research is focused on a lightweight hybrid architecture and a reproducible comparison framework. It employs a parameter-efficient transformer (ALBERT) to learn strong contextual text representations with a smaller model size and training cost, and adds an LSTM layer to learn sequential dependencies that the traditional feature-based ML methods fail to learn. Moreover, a single experimental pipeline is kept between the baseline ML models, deep sequence models, and the hybrid approach,

and the variation in the results can be ascribed to the capability of the model rather than the varying pre-processing or assessment conditions. The contribution of this methodology aids effective deployment restrictions besides enhancing resilience in diverse cybertext patterns.

3.3 Tools, Technologies, and Libraries Used

The given study is conducted in Python and relies on standard data science and deep learning libraries to facilitate every step of the workflow. Numerical operations, data manipulation and data preparation are done with NumPy and Pandas, and evaluation visualisations including confusion matrices and performance plots are created with Matplotlib. Scikit-learn is applied to create and assess the classical machine learning models (e.g., Random Forest), split the dataset, calculate metrics (accuracy, precision, recall, F1-score, confusion matrix, and classification report), and optimize the hyperparameters with the help of GridSearchCV. XGBoost is a tree model that is trained as a gradient-boosted tree to compare with other baselines. In the case of deep learning models (BiLSTM/GRU and the hybrid model), layers of neural networks, model training, and optimization are specified with the help of TensorFlow/Keras. In the case of contextual embeddings, Hugging Face Transformers is loaded and fine-tuned to load and fine-tune ALBERT, as well as to create tokenised inputs that are needed to learn with a transformer.

3.4 Dataset Description

The data in this paper is composed of cyber intelligence text entries that have different levels of annotation. In every record, there are fields like text, entities, relations, comments, id, label, start offset and end offset. Textual content is real-life cyber reports of activities associated with malware, attack patterns, campaigns, and other threat indicators. Some of the records include structured annotations defining the types of entities and their location in the text, and some records are not fully annotated, which is a natural inconsistency that is common with open-source threat intelligence data. The labels in the dataset would cover the categories like malware, attack-pattern, TIME, identity, location, threat-actor, and several technical indicators. Absent or missing labels are treated by putting them in a common category, which balances the categories. The mixed nature of annotated and semi-annotated samples in the dataset also suits it to the training of the models that need to understand the context of cyber threats in unstructured text.

3.5 Data Pre-processing and Exploratory Data Analysis (EDA)

Pre-processing of data refers to the process of changing the raw cybertext into an analysable format. The first steps are to delete the columns that are irrelevant, like Unnamed: 0 and address the situation with the missing labels, placing the undefined cases in the benign category so that all records are consistent. Text fields pass through various cleaning steps, such as lowercasing, punctuation marks, tokenisation, removing of stopwords and SpaCy lemmatisation to standardise linguistic forms. Characters and patterns that are not alpha numbers are eliminated to reduce noise in the text. The limited number of class labels,

including MD5, URL, EMAIL, FILEPATH, and others, are condensed in one category known as NEED_ATTENTION in order to deal with the imbalance in the classes. After pre-processing, there is a significant range of text length as shown by exploratory analysis, with short incident descriptions and long investigative reports. The analysis of distribution of labels indicates that such classes like benign prevail, whereas others are represented in less numbers, which affects modelling strategy. The maximum length of the token and character length tests help in establishing appropriate sequence padding length to be used in the later stages of modelling. On the whole, this step is important to make sure that the textual information is clean, standardised, and statistically interpreted and then it moves to feature extraction and model creation.

3.6 Model Selection and Rationale

The choice of the ALBERT+LSTM architecture is conditioned by the necessity to use the model that can comprehend the deep contextual meaning and sequential dependence in the text related to cyber-related issues. ALBERT is a lightweight member of the BERT family, which produces dense contextual embeddings at the cost of lower computational cost due to parameter sharing, making it appropriate to large textual datasets common in cyber threat intelligence. Transformer encoders by themselves may not however fully represent the temporal or positional flow of threat indicators within a sentence. In response to this, the ALBERT embeddings are fed to an LSTM layer so that the model can learn sequential relationships and long-range dependencies, which are typical in descriptions of cyber incidents. It is a hybrid model that has a semantic power of representations that are transformer-based and the pattern-recognition ability of recurrent neural networks, leading to a better classification performance of various types of threats. The model is thus chosen as the main architecture because of its trade-off between contextual knowledge, computational economy and better adaptability to the variable-length and complex cybertext than a standalone classical machine learning or an advanced deep learning architecture.

3.7 ALBERT + LSTM Architecture Framework

The ALBERT+LSTM model used in the current research combines transformer-based contextual encoding and recurrent sequence modelling to classify cybertext to a higher degree. The pipeline starts by the ALBERT encoder, which has been selected due to its light-weight but very expressive representation features. ALBERT employs factorised embedding parameterisation and cross-layer sharing of parameters, which bring about a great deal of reduction in the complexity of models without degrading their representational depth. Each textual instance is then encoded by the encoder into a 768-dimensional contextual embedding, which is able to capture semantic dependencies, threat-related terminology and linguistic cues of cyber intelligence stories. This embedding is an information-rich dense feature that gives a strong backbone of downstream sequence learning.

After contextual embedding, the vector representation is then inputted to a multi-layer LSTM network that aims to learn long-range dependencies, temporal token dependencies, and

sequence patterns that can commonly occur in descriptions of attack behaviour, malware evolution, and threat-actor activities. The LSTM layers use gated recurrent processes that control information flowing processes to allow the model to retain important contextual cues and to eliminate insignificant or noisy textual elements. Regularisation techniques, including dropout and batch normalisation, stabilise the optimisation procedure and reduce overfitting and guarantee better convergence. The architecture is completed with fully connected layers which map learnt sequence representations to the ultimate SoftMax classifier which generates probability distributions over the target threat categories. This hybrid architecture combines both the semantic expressiveness of ALBERT and the temporal modelling power of LSTM, producing high levels of discriminative performance on heterogeneous cyber threats classes.

3.8 Evaluation Strategy

In the evaluation plan, the performance, stability and reliability of the ALBERT+LSTM model on the various threat categories will be evaluated by tracking the epoch-wise training and validation loss and accuracy to identify convergence and any overfitting. Final performance on a test set held-out after training is evaluated on the basis of overall accuracy and categorical cross-entropy loss. It is reported to analyse the correct and incorrect predictions in all the eight threat classes and class-wise strengths and weaknesses using a confusion matrix. Also, precision, recall, and F1-score are calculated on a per-class basis and macro-averaged and weighted-average summaries are calculated, to ensure that there is a balanced evaluation of the classification amid imbalance between classes, and to facilitate an equitable comparison across datasets.

4 Design Specification

4.1 System Design Specification

It is a multi-stage analytical architecture that takes cyber-related text and classifies it in a set of predefined threat categories with the help of traditional machine learning, deep learning, and transformer-based hybrid models. The entire work process is structured into cascading modules to provide structure to the flow of the raw data ingestion to model evaluation. The design commences with the data collection phase during which a huge dataset of cyber threat intelligence that includes annotated and semi-annotated written records is obtained. This data serves as the basis of all further activities. The second module is data pre-processing that includes cleaning, lemmatisation, noise removal, exploratory statistics analysis, token-level processing, and a structured train-test split, meaning that the input text is standardised and ready to be processed through computational modelling.

The modelling aspect is structured in such a way that it has three parallel branches so that a consistent comparison could be made across methods. The former branch performs classical machine learning paradigms (Random Forest and XGBoost) based on structured feature representations, and the latter branch performs deep learning sequence paradigms (LSTM, BiLSTM, and GRU) to learn sequential text dependency. The third branch operates the

hybrid ALBERT+LSTM model, in which ALBERT produces contextual embeddings which are fed through LSTM layers to identify sequence patterns. The architecture ends with a single evaluation module that compares all models based on accuracy, precision, recall, F1-score and confusion-matrix analysis that offers a modular scheme of evaluating the performance of cybertext threat classification.

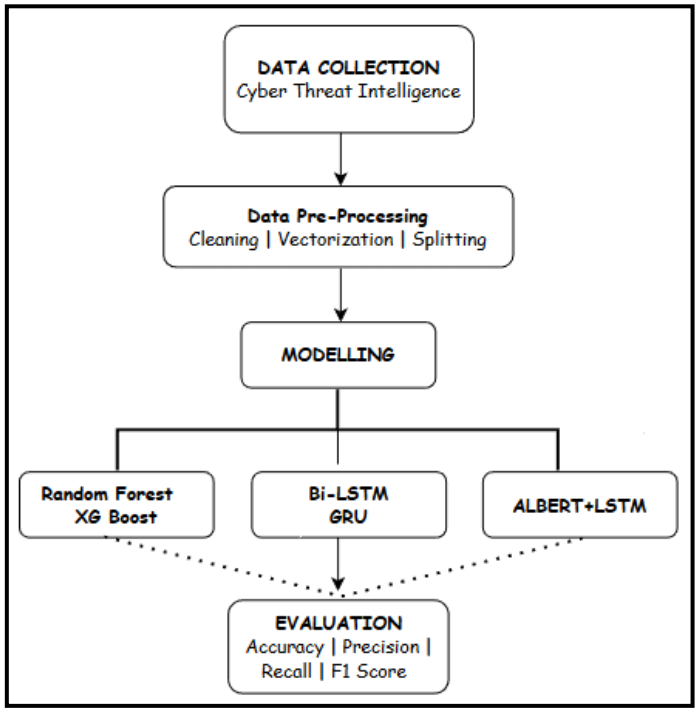

Figure 1. End-to-End Cybertext Threat Classification Workflow

Figure 1 illustrates the complete pipeline from cyber threat text collection through pre-processing (cleaning, vectorization, and train–test splitting) to model training using classical ML, deep learning, and the ALBERT+LSTM hybrid approach.

4.2 Workflow Pipeline

The system workflow is organized as a sequence of pipelines of tasks sequentially processing raw cybertext data into significant threat classifications by a sequence of analytical steps. The process starts with data receiving, during which samples of cyber threat intelligence text are obtained and sorted into a single dataset. It is then followed by the pre-processing phase in which the text is processed by cleaning, tokenisation, lemmatisation, deleting stopwords, normalising labels and performing exploratory analysis to make sure that the data is in a uniform and useful format. The results of this step are further split into training and testing divisions to facilitate objective assessment. The following level focuses on the modelling process, the framework of which is a 3-branch model. The former branch involves the use of classic machine learning models, including Random Forest and XGBoost, which allow making comparisons of the baseline using structured input features. The second branch is concerned with the deep learning sequence model, such as LSTM, Bi-LSTM, and GRU, and

they are applied to study temporal dependencies in the text. The third branch is the primary hybrid design, in which contextual embeddings produced by the ALBERT encoder are integrated with LSTM layers to extract both semantic and sequential patterns in cybertext. After training all the models, the working process shifts to the evaluation step, where the classification metrics, confusion matrices, performance curves, and comparative evaluation will be employed to ascertain system effectiveness.

5 Implementation

5.1 Machine Learning Models Implementation

The machine learning phase entails the training of two base classifiers, namely, the Random Forest and the XGBoost, on systematic hyperparameter optimization to optimise their behaviour in a manner that is suitable to cybertext classification. In the case of Random Forest, the model is set with a range of estimators, tested against the values of 30, 50, 70 and 100 and depth-related parameters are examined to determine the configuration of model complexity and generalisation. Multi-metric grid search using multi-metric scoring, which comprises of accuracy, precision, recall and weighted F1 is used to identify the most effective parameter combination. The tuning process validates the fact that the model performs best with 70 estimators and therefore, this setting is the most robust in validation folds. The XGBoost classifier is applied with the parameters of tuning being the learning rate and the depth of the tree with the range of learning rate parameters as 0.01-0.3 and the depth parameters as various levels. Multi-objective scoring guarantees that there are more criteria in which the model is evaluated in cross-validation. The best XGBoost model is obtained with the learning rate of 0.3 and larger tree depth value, which means that the deeper the boosting framework is, the more the discrimination of cybertext inputs it achieves. These tuned machine learning classifiers are used as comparative baselines to further deep learning and hybrid modelling processes.

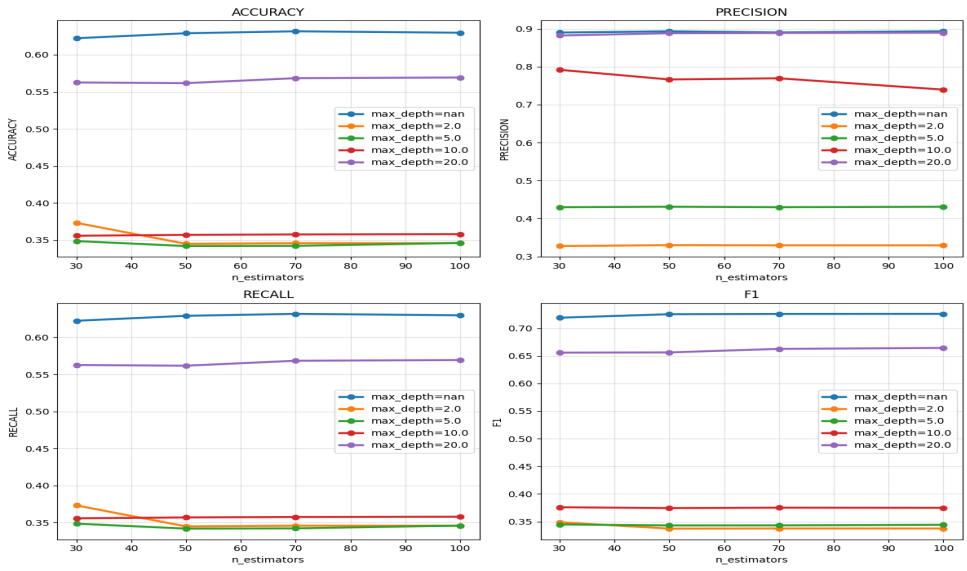

Figure 2. Performance Trends of Random Forest Across Varying Hyperparameters

Figure 2 show the impact of the variation in the number of estimators and depth settings on the accuracy, precision, recall and F1 score of the classifier. The curves indicate that increased values of estimators with unrestricted depth always gives the best and stable performance of Random Forest.

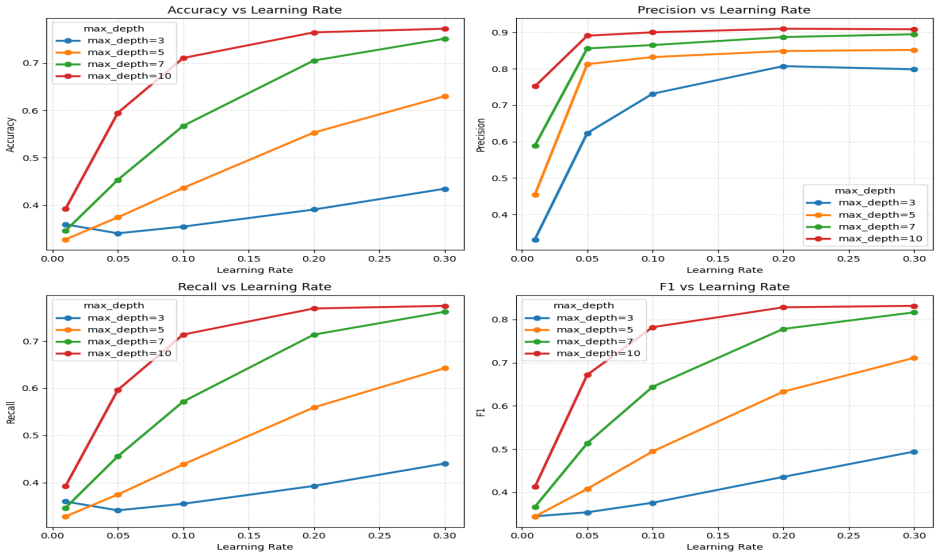

Figure 3. Performance Variation of XGBoost Across Different Learning Rates and Depth Configurations

Figure 3 depict the consistent accuracy, precision, recall, and F1 score increase with growing the learning rate, with more profound tree structures delivered better overall results. The trends show that XGBoost gains much in learning rates with increased depth values and this results in a higher degree of discrimination of the classes in cybertext classification.

Table 2. Hyperparameter Tuning Summary for Random Forest and XGBoost

Model	Hyperparameter	Values tested (Grid Search)	Best value
Random Forest	n_estimators	30, 50, 70, 100	**70**
Random Forest	max_depth	None, 2, 5, 10, 20	**None**
XGBoost	learning_rate	0.01, 0.05, 0.1, 0.2, 0.3	**0.3**
XGBoost	max_depth	3, 5, 7, 10	**10**

A summary of the ranges of hyperparameters tested in GridSearchCV with the Random Forest and XGBoost classifiers is presented in Table 2. It also records the values of the final parameters chosen in each model.

5.2 Deep Learning Models Implementation

The deep learning phase involves the application of three recurrent neural network model's LSTM, BiLSTM, and GRU to test their ability to model sequential patterns in cyber-related

text. Each of the models is constructed with an embedding layer that has the pre-existing vocabulary size and embedding dimension generated in the pre-processing stage. The LSTM variant uses one recurrent layer with 150 units, dropout regularisation and a dense layer of 50 neurons followed by the final softmax output layer with eight class nodes. The BiLSTM model is a generalization of this architecture but it uses 150 units in each direction to enable bi-directional processing so that the network can learn both forward and backward contextual dependencies in the order. Equally, the GRU architecture is also set to a recurrent layer of 150 units, dropout and recurrent dropout are added to enhance the efficiency of generalisation. The models will all employ the Adam optimiser and categorical cross-entropy loss with a batch size of 64 and a fixed number of epochs (10-15) of training controlled by validation behaviour. Early validation loss-based stopping is also added to avoid overfitting and stabilise convergence. Intermediate components consist of dense layers using nonlinear activations and dropout to optimize feature extraction and then classify. These deep learning structures all offer a systematic basis on which sequential modelling approaches can be compared before analysing the transformer-enhanced hybrid model.

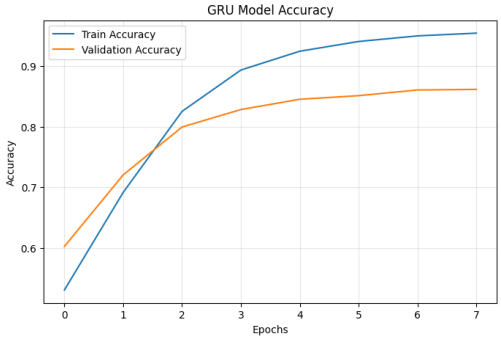

Figure 4. GRU Model Accuracy Plot

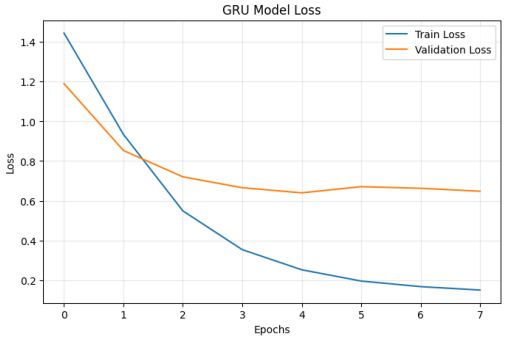

Figure 5. GRU Model Loss Plot

Figure 4 and 5 depict the training and validation accuracy and loss curves of the GRU, and the accuracy and loss curves rise and fall respectively in the epochs, which means that the convergence is stable and there is no serious overfitting.

5.3 ALBERT + LSTM Model Implementation

The hybrid ALBERT+LSTM model is carried out in two phases with a pipeline that consists of transformer-based contextual encoding and recurrent sequence modelling. It starts with the albert-base-v2 tokenizer and encoder being loaded to produce dense contextual embeddings of every cybertext example. In order to make the execution memory safe, the embeddings are generated in small batches, and each input sequence is truncated or padded to a maximum sequence length of 64 tokens. From the encoder output, only the CLS-token representations are retained, resulting in 768-dimensional embedding vectors for all samples. These vectors are then reshaped with an additional time-step dimension to form a suitable input structure for the LSTM network. The downstream classifier is constructed using a stacked LSTM configuration consisting of an initial 256-unit LSTM layer with sequence return enabled, followed by dropout and batch normalisation for regularisation, and a second LSTM layer

with 128 units for deeper temporal feature extraction. The recurrent layers are followed by a 128-unit dense layer with L2 regularisation and dropout, culminating in an output layer with softmax activation matching the number of target classes. Model training employs the Adam optimiser with a learning rate of 1e-4, sparse categorical cross-entropy loss, a batch size of 32, and up to 500 epochs. Early stopping and learning-rate reduction call backs are integrated to stabilise convergence and prevent overfitting. The final architecture thereby integrates ALBERT's semantic contextualisation with LSTM's sequential feature learning to create a robust classifier for cybertext threat categorisation.

6 Evaluation and Results

The analysis outcomes have depicted clearly the relative advantages and drawbacks of each modelling method indicating the performance difference between the traditional methods and the more complex contextual architectures.

Table 3. Results and Performance Analysis

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
Random Forest	0.864	0.900	0.864	0.873
XGBoost	0.861	0.891	0.861	0.868
Bi-LSTM	0.852	0.854	0.852	0.852
GRU	0.858	0.859	0.858	0.858
ALBERT + LSTM	**0.900**	**0.900**	**0.900**	**0.900**

Table 3 gives a comparative summary of the model performance of all the approaches that were evaluated using accuracy, precision, recall and F1-score allowing identification of the best performing classifier directly.

6.1 Random Forest

Random Forest classifier has shown equal results in all the evaluation measures with an accuracy of 0.864 and a precision score of 0.900. The fact that the precision is relatively high indicates that the model is effective in minimizing false positives, i.e. the classifier is accurate when a given threat category has been predicted. The value of recall (0.864) shows that there is a consistent capacity to distinguish between actual instances of instances per classes, however, certain misclassifications also take place. The F1 score of 0.873 indicates the existence of high harmonic balance between precision and recall, indicating that the model is good at dealing with the majority of the class distributions. This performance implies that Random Forest can detect threat patterns based on structured feature representations based on cybertext, but due to its dependence on non-contextual features, it is less able to represent

more semantic complexity, particularly relative to more complex sequential and transformer-based designs.

6.2 XGBoost

The XGBoost classifier shows consistent behaviour in all evaluation measures where the accuracy is 0.861; the precision is 0.891; the recall is 0.861; and F1 score is 0.868. The high value of its precision indicates a high level of discrimination, which minimizes the possibility of classifying innocent content as malicious. The score on recall suggests similar levels of effectiveness of retrieving correct examples of the vast majority of types of threats. The F1 score supports this balance indicating that the model is predictable in varying frequencies of classes. Nevertheless, the findings also demonstrate that performance is still marginally lower than that of the hybrid ALBERT-based model, which means that tree-based boosting fails to capture the feature interaction depth, despite the fact that it is able to capture the representational depth of textual threat analysis.

6.3 Bi-LSTM

An accuracy of 0.852 is achieved with the Bi-LSTM model with a precision, recall, and F1 value of a similar number (0.850). This consistency indicates that model learns both directional contextual relationships in the cybertext and this improves it in comprehending linguistic patterns related to threats. The equal values in the metrics indicate constant generalisation with an equal treatment of frequent and infrequent classes. Nevertheless, its general accuracy is slightly worse than that of Random Forest and XGBoost, which implies that although this model can be used to model sequential information, the embedding representation employed in this setup might not be able to represent higher-level semantic relationships completely. This further emphasizes the reliance of recurrent networks to the quality of initial embeddings, and the drawbacks of pure sequence-based learning without additional contextual encoding.

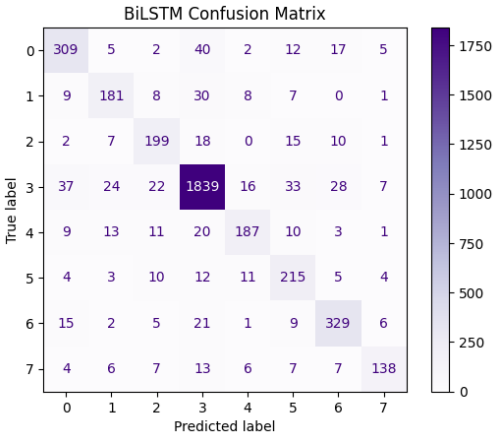

Figure 6. Confusion Matrix of LSTM Model

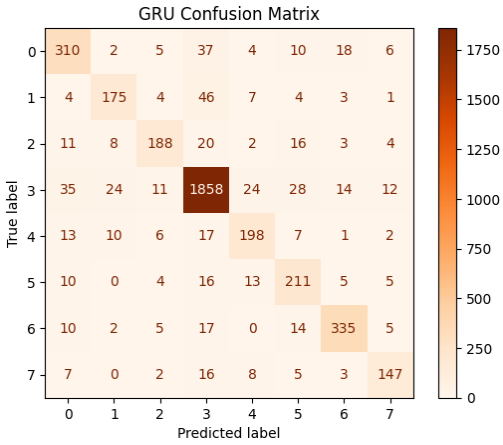

Figure 7. Confusion Matrix of GRU Model

Figure 6 shows that the confusion of BiLSTM model correctly classifies most of the samples (high values in the diagonal), where some misclassification is also observed between very few similar classes.

Figure 7 confusion matrix shows that overall (dominant diagonal) the GRU model has high correct classification and few errors are distributed between some close classes.

6.4 GRU

GRU model has an accuracy of 0.858 and the precision, recall and F1 scores that are centered around 0.858. These findings suggest that temporal information is efficiently learned with marginally quicker convergence qualities than configurations based on LSTM-based configurations. The metric uniformity indicates that the model is generalised to all of the classes without strong bias to prevailing categories. GRU is slightly better than Bi-LSTM in this context but it is not much better than the transformer-enhanced architecture. This level of performance shows that gated recurrent mechanisms can be used to obtain sequence-level patterns, but also confirms that recurrent-only methods can be constrained in their ability to process complex semantic relationships found in text in the cyber domain.

6.5 ALBERT + LSTM

The ALBERT+LSTM hybrid model has the best performance of all the evaluation metrics with an accuracy, precision, recall, and F1 score of 0.900. This homogeneous distribution of scores underscores the fact that the model has a high predictive reliability in all the categories of threats. The contextual embeddings provided by ALBERT together with the sequential processing of context provided by LSTM allow the architecture to learn not only semantic properties of the data, but also the temporal relationships between them, which is a significant benefit over either purely machine learning or purely recurrent deep learning strategies. The findings suggest that the model is effective in the explanation of fine linguistic patterns, long contextual clues, and indirect relationships among threat signals. This stability in the metrics shows that the hybrid model meets the drawbacks that have been found in the previous models that have been used, and it gives the strongest classification behaviour of all the models that have been tested.

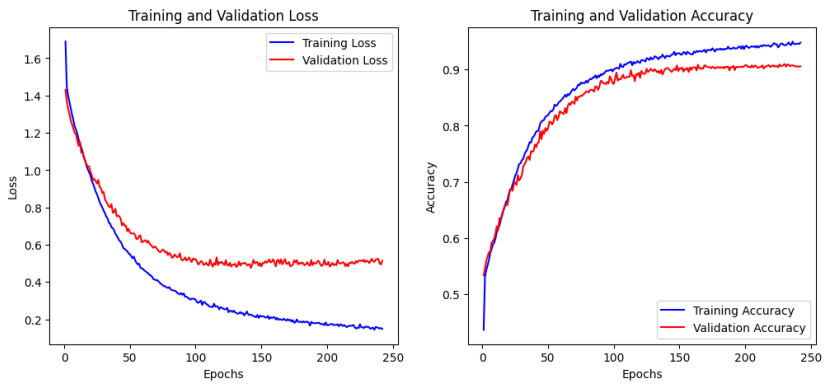

Figure 8. Training and Validation Loss–Accuracy Curves for the ALBERT+LSTM Model

Figure 8 curves depict constant convergence to better accuracy and reduced loss across epochs, which is a sign of stable learning behaviour and high generalisation to unknown cybertext.

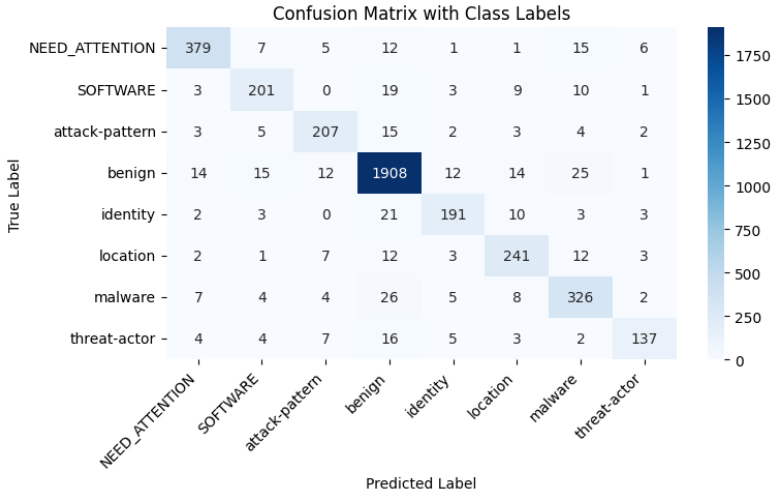

Figure 9. Confusion Matrix of ALBERT+LSTM Predictions

Figure 9 shows the percentages of correct and inaccurate classifications of the hybrid model over eight threat classes with the high values on the diagonal demonstrating the strong accuracy of prediction. The level of misclassification is also low in categories, which exhibit a balanced performance and effective discrimination between closely related types of cyber threats.

6.6 Comparison of Models

The comparative analysis of the models all gives a clear indication of the differences in the performance of machine learning, deep learning, and hybrid architectures. Random Forest and XGBoost had good baseline performance of 0.8641 and 0.8608 respectively and indicated their capability to deal with the structured text-based features but had limited capturing of contextual features. Bi-LSTM and GRU models attained 0.8518 and 0.8581 as their accuracies, which indicates that they were strong in sequential pattern modelling in cybertext but were limited by the conventional embedding representations in their overall classification. Conversely, the ALBERT+LSTM hybrid model demonstrated the best performance with 0.90 accuracy and precision, recall, and F1 score which means it has excellent generalisation and balanced predictive behaviour of all the threat categories. This can be credited to the power of contextual embedding that ALBERT has as well as the temporal learning of LSTM which allows the model to interpret fine threat signals and complicated language structure. Altogether, the comparison proves that hybrid transformer-

recurrent architectures significantly perform better in cybertext classification tasks compared to the traditional and independent deep learning models.

6.7 Comparison with Prior Studies on the Same Dataset

A number of studies have been done to classify cybertext threats using the same data, which serves as a point of reference to place the findings of this study. Ramoliya et al. applied an LSTM model and obtained an accuracy of around 85 percent on the dataset. Based on the same dataset, the ALBERT+LSTM model had a 90 percent accuracy, which is an improvement on the reported LSTM baseline. Such growth is explained by the fact that the contextual embeddings provided by ALBERT are more semantically focused on cyber-related text when compared to the traditional feature representations, and the LSTM element is able to retain sequential information that would be necessary to classify the threat categories correctly.

6.8 Superiority of the ALBERT + LSTM Model

The performance of the ALBERT+LSTM architecture is better than the other two architectures because the two parts are complementary and thus produce a more in-depth semantic understanding and high sequence pattern recognition than the prior architectures. The ALBERT offers a sophisticated contextual embedding framework which is based on sharing of parameters and factorised learning of representations which facilitates extraction of linguistic features that are rich and at the same time is computationally efficient. These embeddings are able to encode threat-domain vocabulary, subtle cyber-attack semantics and contextual cues that cannot be completely understood by traditional models and standalone recurrent networks. These contextual embeddings can be further processed by adding a stacked LSTM layer, which will model temporal connections and token dependencies, which are found in narratives of cyber incidents. This combined representation allows the classifier to get both the sense of the text and the sequence of occurrence of important indicators of threats and thus it has a structural advantage over the other classifier models that learn only through frequency-based or recurrent-only learning.

The overall efficiency of these mechanisms is seen in the evaluation measures in which the ALBERT+LSTM model shows the same high level of results in all accuracy, precision, recall, and F1 measures. The overall performance with all metrics is balanced and shows that, the model does not only identify threats with higher precision, but also provides consistency with both minority and majority classes, which the machine learning baselines demonstrate as slightly different. The architecture shows fewer false positives and negatives, which implies better classification of related threat groups. Moreover, the overall consistency in precision, recall, and F1 values indicates the presence of an optimal harmonic balance, which indicates that the model is able to deal with detection and generalisation. Compared to the results of Random Forest, XGBoost, GRU, and Bi-LSTM, the increased contextual-semantic interaction in ALBERT+LSTM provides it with the representational depth needed to

outperform all other models tested, and thus the strongest and most reliable classifier in this paper.

7 Conclusion and Future Work

7.1 Conclusion

This paper has investigated the issue of detecting cyberattacks and malware in a text-based cybercrime by comparing various machine learning, deep learning, and transformer-based architectures. The analysis also focused on the question of whether contextual semantic modelling and sequence-aware learning can enhance classification of threat categories, i.e., malware types, attack patterns, and threat actors. The research question was whether a model can out-compete the traditional methods and encompass more context and more powerful sequential patterns in cybertext. The general process of work was based on the Knowledge Discovery in Databases (KDD) process, which was supported by the systematic pre-processing, numerous modelling experiments, and the total evaluation of the results in terms of accuracy, precision, recall, F1-score, and confusion-matrix analysis. The ALBERT+LSTM hybrid model shown as the best in all the assessed approaches was the most consistent and effective at that point, as it combines the contextual embeddings with the sequence learning to identify semantic and temporal relations in cybertext, thus answering the main research question and outperforming the baseline models.

This study was able to meet its objectives. A pipeline of end-to-end cybertext classification was created that comprised of the dataset preparation, text pre-processing, transformation to model ready representations, and multiclass threat classification. A consistent experimental setup was used to implement and evaluate baseline machine learning models (Random Forest and XGBoost) and deep learning sequence models (LSTM, BiLSTM and GRU). The hybrid ALBERT+LSTM architecture was also applied and evaluated under identical conditions and allowed direct comparison of the approaches. The general analysis based on accuracy, precision, recall, F1-score, and confusion-matrix analysis revealed that the hybrid model provided the best and most consistent performance, which proves that contextual embeddings along with sequence learning may enhance cybertext threat classification and achieve the predetermined research goals.

7.2 Future Work

Future research can involve adding more multilingual cyber intelligence data to the dataset to experiment with cross-lingual resilience and flexibility. Research on domain-specific pre-training of ALBERT or other lightweight transformer models may also increase the quality of contextual embedding further. Fine-grained differentiation of the closely related types of threats can be enhanced by incorporating attention-based recurrent models or incorporating contrastive learning techniques. Moreover, application of the model in a real-time threat monitoring setting might offer an insight into the operational performance, latency behaviour, and system level integration issues. These guidelines provide an opportunity to further

develop the model in terms of its effectiveness and expand its applicability to more comprehensive cybersecurity intelligence models.

References

Ahmadi, M., Khajavi, M., Varmaghani, A., Ala, A., Danesh, K., & Javaheri, D. (2025). Leveraging large language models for cybersecurity: Enhancing SMS spam detection with robust and context-aware text classification. *Cyber-Physical Systems*, 1–27. https://doi.org/10.1080/23335777.2025.2550938

Akhtar, M. S., & Feng, T. (2022). Malware Analysis and Detection Using Machine Learning Algorithms. *Symmetry*, *14*(11), 2304. https://doi.org/10.3390/sym14112304

Albraikan, A., Hassine, S., Fati, S., Al-Wesabi, F., Hilal, A., Motwakel, A., Hamza, M., & Al Duhayyim, M. (2022). Optimal Deep Learning-based Cyberattack Detection and Classification Technique on Social Networks. *Computers, Materials & Continua*, *72*, 907–923. https://doi.org/10.32604/cmc.2022.024488

Alsufyani, A., & Alzahrani, S. (2021). Social Engineering Attack Detection Using Machine Learning: Text Phishing Attack. *Indian Journal of Computer Science and Engineering*, *12*, 743–751. https://doi.org/10.21817/indjcse/2021/v12i3/211203298

Atoum, M. S., Alarood, A. A., Alsolami, E., Abubakar, A., Hwaitat, A. K. A., & Alsmadi, I. (2025). Cybersecurity intelligence through textual data analysis: A framework using machine learning and terrorism datasets. *Future Internet*, *17*(4), 182.

Benavides-Astudillo, E., Fuertes, W., Sanchez-Gordon, S., Nuñez-Agurto, D., & Rodríguez-Galán, G. (2023). A phishing-attack-detection model using natural language processing and deep learning. *Applied Sciences*, *13*(9), 5275.

Bibi, A., Sampedro, G. A., Almadhor, A., Javed, A. R., & Kim, T. (2023). A Hypertuned Lightweight and Scalable LSTM Model for Hybrid Network Intrusion Detection. *Technologies*, *11*(5), 121. https://doi.org/10.3390/technologies11050121

Brightwood, S., & Anthomy, L. (2024). *Practical Considerations and Challenges in Applying Logistic Regression for Cyber Threat Detection*.

Coyac-Torres, J. E., Sidorov, G., Aguirre-Anaya, E., & Hernández-Oregón, G. (2023). Cyberattack detection in social network messages based on convolutional neural networks and NLP techniques. *Machine Learning and Knowledge Extraction*, *5*(3), 1132–1148.

Ferrag, M. A., Ndhlovu, M., Tihanyi, N., Cordeiro, L. C., Debbah, M., Lestable, T., & Thandi, N. S. (2024). Revolutionizing Cyber Threat Detection With Large Language Models: A Privacy-Preserving BERT-Based Lightweight Model for IoT/IIoT Devices. *IEEE Access*, *12*, 23733–23750. https://doi.org/10.1109/ACCESS.2024.3363469

Raghunath, K. K., Kumar, V. V., Venkatesan, M., Singh, K. K., Mahesh, T. R., & Singh, A. (2022). XGBoost regression classifier (XRC) model for cyber attack detection and classification using inception V4. *Journal of Web Engineering*, *21*(4), 1295–1322.

Raj, C., Agarwal, A., Bharathy, G., Narayan, B., & Prasad, M. (2021). Cyberbullying detection: Hybrid models based on machine learning and natural language processing techniques. *Electronics*, *10*(22), 2810.

Ramamoorthy, M., Kumar, B. A., Singh, A., & Karthik, B. (2024). Detecting cyberbullying in text using TF-IDF vectorization and support vector machine classification. *2024 International Conference on Advances in Computing, Communication and Materials (ICACCM*, 1–5.

Ramoliya, F., Kakkar, R., Gupta, R., Tanwar, S., & Agrawal, S. (2023). SEAM: Deep learning-based secure message exchange framework for autonomous EVs. In *2023 IEEE Globecom Workshops (GC Wkshps* (pp. 80–85). IEEE.

Srivastava, A., Tiwari, S., & Sresth, V. (2021). The Role of AI in Cyber Risk Management: Predictive Analytics for Security Incident Forecasting and Mitigation Vishal Sresth Independent Researcher. *INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS*, *8*, 945.