EAI Endorsed Transactions
on AI and Robotics
Research Article EAI.EU

Comparative Analysis of Transformer and LSTM Architectures for Cybersecurity Threat Detection Using Machine Learning

Jobanpreet Kaur[1], Mani Prabha[2,] Md Samiun[2], Syed Nazmul Hasan[1], and Rakibul Hasan[3,*], Hammed Esa[2], Md Fakhrul Hasan Bhuiyan[4], Md Abdur Rob[5], Durga Shahi[3]

[1]College of Technology & Engineering, Westcliff University, CA 92614, USA
[2]Department of Business Administration, International American University, Los Angeles, CA 90010, USA
[3]Department of Business Administration, Westcliff University, Irvine, CA 92614, USA
[4]College of Engineering and Science, Trine University, Angola, IN 46703, USA
[5]Department of Economics, Ohio University, Athens, OH 45701, USA

Abstract

The growing prevalence of advanced persistent threats (APTs), zero-day exploits, and the rapid proliferation of IoT devices have exposed limitations in traditional cybersecurity approaches. In response, this study presents a comparative analysis of deep learning models—specifically Long Short-Term Memory (LSTM) and Transformer-based architectures—for cybersecurity threat classification from textual data. Leveraging a standardized dataset and consistent preprocessing pipeline, both models are evaluated across key performance metrics, including accuracy, precision, recall, and F1-score. The results demonstrate that Transformer models significantly outperform LSTM-based approaches, exhibiting superior capacity to capture long-range dependencies, handle complex threat narratives, and generalize to previously unseen data. These findings offer valuable insights into the practical application of modern deep learning techniques in cybersecurity and provide a foundation for designing more robust and adaptive threat detection systems.

Keywords: Cybersecurity, machine learning, LSTM, transformer, threat classification, emerging threats, predictive analytics.

Received on 19 July 2025, accepted on 30 August 2025, published on 16 September 2025

doi: 10.4108/airo.9759

*Corresponding author. Email: r.hasan.179@westcliff.edu

1. Introduction

In the digital age, the imperative for robust cybersecurity measures has reached an unprecedented zenith. The rapid proliferation of internet connectivity, the ascendancy of cloud computing, and the pervasive integration of Internet of Things (IoT) devices have engendered a vast digital ecosystem. While these advancements confer myriad benefits such as enhanced operational efficiency and democratized access to information, they concurrently expose organizations to a spectrum of cyber threats.

The World Economic Forum's Global Risks Report [1] highlights the escalation of cybercrime as a significant global risk, underscoring the urgent need for effective defenses against increasingly sophisticated cyber threats. As highlighted by Symantec [2], adversaries have transitioned from rudimentary malware to intricate, targeted attacks, leveraging APTs, zero-day exploits, and IoT vulnerabilities to breach defenses.

The ramifications of such cyber threats are profound, leading to severe data breaches, substantial financial losses, and disruptions of critical infrastructure. Indeed, projections indicate that cybercrime will inflict a staggering economic toll of approximately $10.5 trillion annually by 2025 [3]. This alarming trajectory emphasizes the necessity for adaptive cybersecurity frameworks capable of evolving in tandem with emerging threats [4][5].

Recent studies have illuminated the potential of machine learning (ML) as a transformative tool in cybersecurity, offering innovative methodologies for threat detection and classification. ML techniques, particularly those leveraging deep learning architectures such as Long Short-Term Memory (LSTM) networks and transformer models, have been explored for their efficacy in discerning patterns within voluminous textual data related to cyber threats [6][8]. However, a discernible gap persists in comparative analyses of these models' performance in threat classification tasks. Specifically, there is a lack of head-to-head comparisons between LSTM and transformer architectures under similar conditions, as well as limited exploration of how contextual features may enhance model performance. Additionally, existing research often utilizes diverse datasets and metrics, complicating efforts to benchmark model effectiveness across various cyber threat scenarios.

This study aims to bridge this gap by rigorously evaluating the effectiveness of transformer-based models in classifying cybersecurity threats from textual data, juxtaposed with traditional LSTM frameworks. Specifically, we will assess the strengths and limitations of various ML techniques, culminating in the proposal of a novel classification model predicated on transformer architecture. The findings from this research will contribute to a deeper understanding of the role of ML in cybersecurity and inform the development of more resilient defenses against cyber threats. Our key contributions include: (1) a head-to-head evaluation on identical preprocessed threat data, (2) consistent benchmarking against a state-of-the-art model (SEAM), and (3) detailed analysis of generalization and loss behavior.

The subsequent sections of this paper are organized as follows: the Literature Review surveys existing research on ML applications in cybersecurity; the Methodology delineates the research methods employed, including data collection and evaluation metrics; the Results Analysis and Discussion section presents an analysis of the performance of the proposed models; and the Conclusion synthesizes key findings, addresses limitations, and proffers recommendations for future research.

2. Literature Review

2.1. Evolution of Cybersecurity Threats

The landscape of cybersecurity threats has undergone a significant transformation over the past few decades, evolving from relatively simple forms of malware to highly sophisticated and persistent attacks. The early days of computing were marked by the emergence of viruses and worms, often created more for notoriety than financial gain. As the internet became more prevalent and valuable data moved online, cybercrime evolved into a sophisticated and lucrative industry. By the early 2000s, threats had grown more complex, with the rise of more destructive malware [9]. The late 2000s saw a shift towards financially motivated attacks, including phishing schemes and ransomware. For instance, in 2007, a shift in malicious activity towards Web-based attacks, targeting individual computers through trusted websites was noticed. This shift is likely due to increased security measures on networks and the potential for greater success in targeting end-users [10]. The report also notes an increase in phishing attacks, particularly targeting ISPs and financial institutions, and a rise in banking trojan infections. In recent years, the threat landscape has further evolved to include advanced persistent threats (APTs) and state-sponsored attacks [2]. APTs are characterized by their stealth, persistence, and the involvement of well-organized groups, often state-sponsored, aiming to infiltrate networks and remain undetected for extended periods to extract valuable information or cause significant harm [11].

Alongside the proliferation of APTs, the sophistication of attack techniques has continued to grow, with cybercriminals increasingly leveraging zero-day exploits and AI-powered attacks. Zero-day exploits take advantage of previously unknown vulnerabilities and for which no patches exist. These exploits enable attackers to breach systems without detection, making them extremely challenging to defend against using traditional security measures [12]. The emerging trend of AI-powered attacks has also raised significant concerns, as malicious actors leverage artificial intelligence and machine learning to automate and scale their attacks [13]. IoT devices, often characterized by limited computational resources and weak security protocols, are particularly vulnerable to attacks, making them attractive targets for cybercriminals [14]. Similarly, the migration to cloud computing environments has introduced challenges related to data security, access control, and the potential for large-scale breaches [15].

2.2. Limitations of traditional cybersecurity approaches in addressing evolving threats

Traditional approaches to cybersecurity have proven increasingly inadequate in addressing the evolving threat landscape. As traditional intrusion detection systems (IDSs) often rely on outdated methods like pattern-based detection, they struggle to identify new or modified attacks [16]. Besides, rule-based approaches require constant knowledge updates, while heuristic-based methods are computationally expensive. These conventional methods often struggle to keep pace with the rapidly changing attack vectors, as they are primarily designed to detect known threats based on predefined patterns [17]. Firewalls and IDS have been foundational components of network security, designed to monitor and control incoming and outgoing network traffic based on predetermined security rules. Firewalls act as barriers between trusted and untrusted networks, while IDS monitor network traffic for signs of suspicious activity. Attackers have developed techniques to bypass or disable these defenses, rendering them less effective against sophisticated attacks like APTs or zero-day exploits [18]. One of the primary shortcomings of traditional cybersecurity approaches is their reactive nature. Many traditional systems rely on identifying and responding to threats after they have

already infiltrated a network, rather than proactively identifying and mitigating potential threats before they can cause harm. This limitation is particularly problematic in the context of novel and rapidly evolving threats, where time is of the essence and delays in detection can result in significant damage. As cyber threats become more sophisticated, dynamic, and difficult to detect, new strategies and technologies are required to anticipate and mitigate emerging challenges.

2.3. Emergence of machine learning (ML) as a transformative technology in cybersecurity

The integration of Machine Learning (ML) into cybersecurity represents a paradigm shift, offering new ways to enhance threat detection and response. The adoption of ML in cybersecurity has been gradual, beginning with basic applications such as spam filtering [19] and progressing to more sophisticated uses in recent years. Notable milestones include the development of IBM Watson for Cyber Security [20], which leverages ML and natural language processing to analyze vast amounts of security data, and the use of deep learning models for more accurate malware detection.

ML offers several distinct advantages over traditional cybersecurity methods. One of the key benefits is adaptability—ML models can continuously learn and evolve by analyzing new data, enabling them to stay ahead of emerging threats. This adaptability is particularly important given the constantly changing nature of cyber threats. ML techniques, such as anomaly detection, predictive analytics, and behavior-based threat detection, enable cybersecurity systems to identify and respond to previously unknown threats, reducing the time between detection and mitigation [21]. ML models can scale to analyze large datasets, making them suitable for use in environments with high volumes of network traffic or security logs. The study also suggests that predictive capabilities of ML allow for the identification of potential threats before they materialize, providing a more proactive approach to cybersecurity.

2.4. Current Applications of ML in Cybersecurity

Machine Learning has found diverse applications within the field of cybersecurity, with some of the most prominent areas being anomaly detection, malware classification, network intrusion detection, and threat intelligence. Anomaly detection involves identifying deviations from normal behavior in network traffic or user activity, which may indicate the presence of a security threat. ML models are particularly effective at this task, as they can analyze large volumes of data and detect subtle patterns that might escape traditional detection methods [22]. Malware classification and detection is another key area where ML has made significant contributions. By training models on large datasets of known malware samples, researchers have developed systems that can accurately identify new and unknown malware based on their behavior or characteristics, rather than relying solely on signatures [23]. Network intrusion detection and prevention systems (NIDS/NIPS) also benefit from ML, with models capable of analyzing network traffic in real time to identify and block potential intrusions. This study [24] presents a novel approach to intrusion detection by hierarchically combining misuse detection and anomaly detection models. The proposed method utilizes a C4.5 decision tree to initially decompose the normal training data into smaller subsets. Subsequently, a one-class support vector machine is employed to create an anomaly detection model for each of these subsets. This integration allows the anomaly detection model to indirectly leverage known attack information, leading to enhanced performance in identifying unknown attacks. Threat intelligence and predictive analytics represent the frontier of ML in cybersecurity. By extracting features from packet data and applying various machine learning models, the system can effectively predict suspicious packets [25]. The models tested in this work include neural networks, support vector machines, logistic regression, and linear regression. Experimental results demonstrate the effectiveness of the approach in detecting trojan malware and other malicious activities. This capability allows organizations to identify and mitigate threats before they can cause significant damage.

2.5. Emerging Challenges and Limitations of ML in Cybersecurity

Despite its potential, the application of Machine Learning in cybersecurity is not without challenges. One of the primary issues is the quality and availability of data. Effective ML models require large volumes of high-quality data to train on, but obtaining such data can be difficult due to privacy concerns, the sensitive nature of cybersecurity incidents, and the potential for biased or incomplete datasets. Despite its potential, the application of Machine Learning in cybersecurity is not without challenges. One of the primary issues is the quality and availability of data. Effective ML models require large volumes of high-quality data to train on, but obtaining such data can be difficult due to privacy concerns, the sensitive nature of cybersecurity incidents, and the potential for biased or incomplete datasets. This study [26] presents a comprehensive survey of existing network-based intrusion detection data sets. It analyzes the properties of these data sets, including data format, volume, recording environment, and evaluation criteria. The paper provides a detailed overview of 34 data sets, highlighting their unique characteristics and suitability for different research purposes. Adversarial machine learning poses another significant challenge. Attackers can manipulate inputs to ML models in ways that cause the models to make incorrect predictions, potentially leading to security breaches. This area of research is critical, as the effectiveness of ML-based cybersecurity solutions depends on their resilience to such attacks. [27] offers a comprehensive overview of the research landscape in adversarial machine learning, particularly within the context of cybersecurity. This study highlights the susceptibility of

machine learning models to adversarial attacks and the potential consequences of such attacks in cybersecurity, including compromised system security, false positives, and false negatives. The interpretability and explainability of ML models also remain areas of concern. Many ML models, particularly deep learning models, operate as "black boxes," making it difficult for security professionals to understand how they arrive at their decisions. This lack of transparency can hinder trust and adoption, especially in high-stakes environments where understanding the reasoning behind security decisions is essential. The work [28] presents a comprehensive overview of methods for explaining black box, highlighting the importance of defining a common formalism for explaining black boxes, measuring their comprehensibility, and addressing latent features. Moreover, the integration of ML in cybersecurity raises ethical considerations and privacy concerns. The collection and analysis of large amounts of data, especially personal data, must be done in compliance with privacy regulations and ethical standards. Additionally, there is the risk that ML models could inadvertently reinforce existing biases in data, leading to unfair or discriminatory outcomes. The study [29] finds that while algorithms can be used to address ethical concerns, they also introduce new challenges due to their complexity and opacity. It emphasizes the need for a comprehensive approach to the ethics of algorithms that considers various factors, including epistemic deficiencies, ethical residues, and the interconnectedness of ethical concerns.

The review of existing literature highlights several limitations in the application of machine learning (ML) to cybersecurity. Traditional methods have struggled to keep up with the rapidly evolving threat landscape, particularly in addressing advanced persistent threats (APTs) and zero-day exploits, which are beyond the scope of conventional, pattern-based detection systems. While ML models offer more adaptive solutions, challenges such as the quality and availability of data remain significant, as large volumes of high-quality data are often difficult to obtain due to privacy concerns, biases, and the sensitive nature of cybersecurity incidents. Furthermore, adversarial attacks pose a critical risk, with malicious actors exploiting vulnerabilities in ML models to cause incorrect predictions, potentially leading to security breaches. Additionally, the interpretability of these models is another key issue, as deep learning techniques often function as "black boxes," limiting transparency and trust in their decision-making processes. Ethical and privacy concerns also arise, particularly in relation to the use of personal data and the potential for biased or unfair outcomes. In light of these limitations, the objective of this study is to develop a machine learning model specifically designed for classifying cybersecurity threats from textual data, addressing issues related to adversarial attacks and model transparency. The study will also compare the performance of the proposed model with traditional LSTM models to assess its effectiveness in real-world cybersecurity applications.

3. Methodology

This study employs a deep learning approach to detect and classify cyber threats using textual data. The methodology encompasses data preparation, feature engineering, model construction, and evaluation. The process is detailed as follows:

3.1. Dataset

The dataset utilized in this study is obtained from Kaggle, titled Text-Based Cyber Threat Detection [30]. This dataset encompasses network traffic data, a diverse range of cyber threat-related textual content, and entity relationships, making it a valuable resource for training machine learning models in cybersecurity applications. The data includes various labels indicating different categories of threats, which are essential for the supervised learning tasks undertaken in this study. This dataset is a comprehensive collection designed for cyber threat detection, diagnosis, and mitigation. It is structured to provide a holistic view of cyber threats, including their identification, analysis, and potential solutions. This rich and multifaceted dataset has numerous potential applications in the field of cybersecurity. It can be used to train machine learning models for cyber threat detection and classification, conduct threat intelligence and analysis, develop incident response and mitigation strategies, implement real-time network security monitoring, and serve as a valuable resource for cybersecurity education and research.

3.2. Data Loading and Preprocessing

To prepare the raw textual data for training deep learning models, a structured preprocessing pipeline was applied, ensuring both consistency and preservation of semantic information critical for cyber threat classification.

- Text Cleaning: The text was initially standardized by removing URLs, punctuation, and special characters, and converting all text to lowercase. Unlike traditional NLP pipelines, stopwords were retained to preserve contextual cues often essential in threat descriptions.
- Data Loading: The cleaned dataset was imported using Python's Pandas library, which provides efficient handling of large-scale data and enables initial inspection and quality checks.
- Tokenization: Text sequences were tokenized using Keras's Tokenizer class. The vocabulary was limited to the 10,000 most frequent words to maintain computational efficiency, with an Out-of-Vocabulary (OOV) token assigned to rare or unseen terms. This step converted each document into a sequence of integers, aligning the input with model requirements.
- Sequence Padding: Deep learning models like LSTM, BiLSTM, and Transformers require

uniform input lengths. Therefore, all tokenized sequences were padded or truncated to a fixed length of 128 tokens using Keras's pad_sequences function. This ensured compatibility with batch processing during training.

- Label Encoding: The eight distinct cyber threat categories were transformed using one-hot encoding, resulting in binary vectors where each dimension corresponds to a specific threat class. This encoding format aligns with the requirements of multi-class classification tasks.

These preprocessing steps were carefully selected to balance model efficiency and classification performance, enabling the downstream models to process structured inputs while maintaining the contextual integrity of the original cybersecurity data.

3.3. Feature Engineering

Feature engineering is a critical step in the machine learning pipeline, especially in the context of cybersecurity, where the quality and relevance of the features significantly impact the model's performance. In this study, feature engineering involves two stages designed to transform raw textual data into a structured format that can be effectively processed by LSTM and BiLSTM models.

3.3.1. Sequence Preparation

The first step in feature engineering is the preparation of sequences from the tokenized text. Unlike traditional machine learning models that might treat text as a bag of words, LSTM and BiLSTM models rely on the sequential nature of the data. Therefore, maintaining the order of words in a sentence is crucial. The tokenized text is structured into sequences where each sequence represents a continuous segment of text. These sequences are designed to capture the temporal dependencies in the data, which is essential for understanding the context of cyber threat indicators. For instance, the phrase "unauthorized access attempt" carries different implications depending on its context within a broader text, and LSTM/BiLSTM models can recognize such nuances through properly structured sequences.

3.3.2. Label Binarization with One-Hot Encoding

The original labels in the dataset, which represent different categories of cyber threats, are categorical and need to be transformed into a numerical format that the model can work with. Label binarization in this study is done using one-hot encoding, where each category is converted into a binary vector. For instance, as there are eight classes of cyber threats, namely 'NEED_ATTENTION' 'SOFTWARE' 'attack-pattern' 'benign' 'identity' 'location' 'malware' 'threat-actor' each label is converted into a vector of length eight, with a '1' indicating the presence of a particular class and '0' indicating its absence. This method ensures that the model's output layer can correctly predict the probability of each class.

3.4. Model Development

In this study, we employed a multifaceted approach to classify cyber threat intelligence, integrating both recurrent neural networks and Transformer-based models. Recurrent models such as LSTM and BiLSTM were initially utilized due to their strength in capturing sequential dependencies inherent in cybersecurity data. However, recognizing the limitations of these models in handling long-range dependencies, we proposed a more advanced architecture rooted in the Transformer framework. This Transformer-based approach leverages self-attention mechanisms to extract deep contextual relationships from the text, enhancing the detection of nuanced threats. Through iterative training, optimization, and regularization, the proposed models were fine-tuned for the robust classification of cyber threats across a diverse dataset.

3.4.1. Recurrent Models: LSTM and BiLSTM

To leverage the temporal dynamics of cybersecurity text data, we initially employed two recurrent neural network architectures: LSTM and BiLSTM. These architectures have been extensively validated for their capability to capture sequential dependencies, a property crucial for detecting latent patterns in threat-related text.

LSTM Architecture: The LSTM model was configured to process input sequences through a series of layers designed to progressively extract features. Starting with a tokenized input, the model used an LSTM cell to capture long-range dependencies, enabling the model to retain critical information over extended sequences. This capability is essential in the context of cybersecurity, where threats may emerge from subtle and delayed connections between events. The cell architecture's gating mechanisms allowed for the selective memory retention and updating of the sequence information, refining the model's capacity to discern meaningful insights.

BiLSTM Architecture: To complement the unidirectional processing of LSTM, the BiLSTM model was integrated. This model introduced bidirectional layers, processing sequences in both forward and reverse order. The concatenation of these outputs provided the network with a more holistic understanding of the sequence, allowing for the identification of threat patterns that could manifest both before and after specific triggers within the text. This bidirectional approach was especially pertinent to the task of cybersecurity threat detection, where understanding both the preceding and succeeding context is critical. Figure 1 illustrates the architecture of a Bi-LSTM model, which was used to experiment on the Dataset.

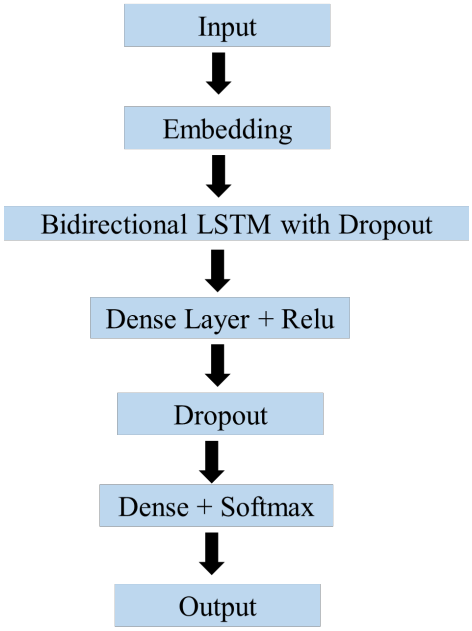

Figure 1. Architecture of Bi LSTM model

3.4.2. Transformer Model Architecture

Building on the limitations of recurrent models, we proposed a Transformer-based model rooted in the Transformer architecture, tailored for the classification of cyber threat reports. This model exploits self-attention mechanisms, offering a robust way to handle long-range dependencies and mitigate the sequential bottlenecks inherent in recurrent architectures.

Input Layer and Embedding

The model begins with an input layer designed to handle fixed-length sequences of tokenized text, derived from preprocessed cybersecurity data. Each token is embedded into a dense, continuous vector space via a learned embedding matrix. Formally, the embedding can be represented as:

$$E(x_i) = W_e \cdot x_i$$

where x_i is the i-th token in the input sequence, and W_e is the learned embedding matrix. This representation encapsulates semantic relationships between tokens, which is crucial for distinguishing nuanced threat categories in complex text corpora.

Multi-Head Self-Attention: The multi-head attention mechanism is a pivotal component of our model, enabling parallel processing of text sequences. By allowing the model to attend to multiple positions within the input simultaneously, this mechanism uncovers intricate relationships between tokens. The attention heads, operating concurrently, enhance the model's ability to contextualize individual tokens within the broader scope of the sequence, significantly improving the detection of subtle threats. Each attention head computes a self-attention score using the query (Q), key (K), and value (V) matrices as follows:

$$\text{Attention}(\text{Q}, \text{K}, \text{V}) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})\text{V}$$

where d_k is the dimension of the key vectors. This allows the model to attend to different parts of the sequence in parallel, significantly improving the detection of subtle relationships in the text.

Residual Connections and Normalization: Residual connections and layer normalization are applied to stabilize training and maintain performance across deeper layers. The output of each layer is given by:

$$x_{output} = LayerNorm(x_{input} + Layer(x_{input}))$$

This setup ensures that the model retains important information across layers, while also normalizing the output distribution, which aids in efficient convergence.

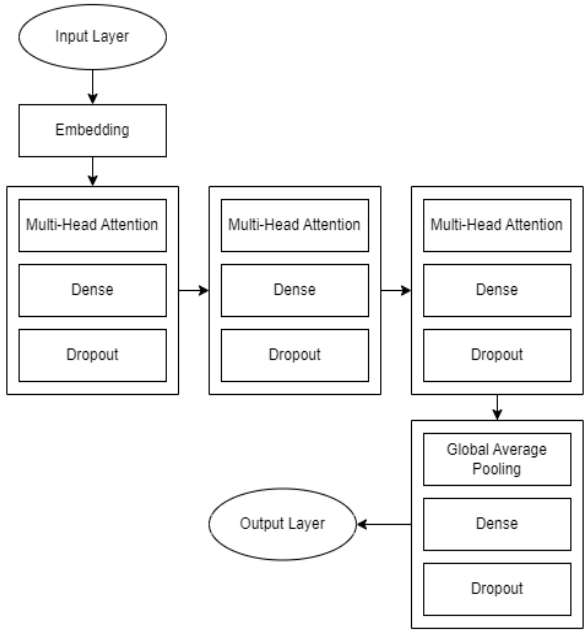

Figure 2. Architecture of the proposed transformer models.

Feedforward Layers and Stacking: To further refine the learned representations, the output from the attention layers is passed through fully connected dense layers. These layers, interspersed with dropout for regularization, are instrumental in reducing dimensionality and capturing high-level abstractions. The entire architecture is repeated multiple times, forming a deep stack of attention blocks that refines the attention maps, leading to a more precise identification of cybersecurity threats.

Global Pooling and Output: After the attention layers, global pooling ensures a fixed-size output for the classifier. The pooled output is passed through fully connected layers, and finally, a softmax activation function is applied to produce probabilities for the threat classes:

$$P(y = c \mid x) = softmax(h)$$

where h represents the pooled output, and the softmax converts it into class probabilities. This final classification layer outputs the predicted threat class, vital for subsequent threat mitigation actions.

3.4.3. Model Regularization and Optimization

To enhance model generalization and prevent overfitting, multiple regularization and optimization strategies were adopted across the evaluated architectures.

For the LSTM and BiLSTM models, dropout layers were incorporated between hidden layers to reduce co-adaptation of neurons and improve robustness. Early stopping was employed based on validation loss, enabling training to terminate once performance plateaued, thereby preventing overtraining. The Adam optimizer was chosen for its adaptive learning capabilities, allowing efficient convergence across sequence-based architectures.

For the Transformer-based model, regularization was similarly applied using dropout within encoder layers. In addition to early stopping, a learning rate scheduler was utilized to dynamically adjust the learning rate in response to validation performance, promoting stable optimization. Adam remained the optimizer of choice due to its ability to handle sparse gradients and adapt learning rates across parameters. These strategies ensured that models were not only optimized for training efficiency but also capable of maintaining generalization performance on unseen data.

3.4.4. Hyperparameter Tuning and Training Environment

Hyperparameters were tuned within predefined ranges through manual grid search, with early stopping applied to prevent overfitting. For the LSTM and BiLSTM models, hidden units between 64 and 256 and dropout rates from 0.2 to 0.5 were tested; the final choice was two layers of 128 units with a dropout of 0.3 and a learning rate of 0.001. For the Transformer, encoder layers (1–4), attention heads (4–12), and embedding dimensions (128–512) were explored, with the best configuration being two layers, eight heads, and an embedding size of 256. Dropout of 0.3 and a learning rate of 2e-4 (with a scheduler) provided the most stable training.

All experiments were conducted in Python (TensorFlow and PyTorch) on an NVIDIA RTX 3090 GPU with 24 GB VRAM. Training typically required less than one hour for LSTM/BiLSTM models and around three hours for the Transformer model.

3.5. Training and Validation

Our experiments involved splitting the dataset into training and validation sets, maintaining an 80:20 ratio. The BERT model, along with the recurrent LSTM and BiLSTM models, was trained for a predefined number of epochs, with batch sizes optimized for computational efficiency. Model performance was evaluated based on key classification metrics, ensuring robust and reliable threat detection performance across varying cybersecurity data inputs.

3.6. Model Compilation

The compiled model is configured with a loss function, optimizer, and evaluation metrics, which are critical for guiding the training process.

Loss Function: The loss function quantifies the difference between the predicted and actual class labels, guiding the model in adjusting its weights during training to minimize this difference. Categorical cross-entropy is used, which is appropriate for multi-class classification.

Optimizer: The Adam optimizer is selected for its efficiency and effectiveness in training deep learning models. It adapts the learning rate for each parameter, which accelerates convergence and helps the model escape local minima.

Evaluation Metrics: Evaluation metrics such as accuracy, precision, recall, and F1-score are used to assess the model's performance during training and validation. These metrics provide insights into the model's ability to correctly classify cyber threats.

4. Results and Discussion

The comparative evaluation of Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (BiLSTM), and Transformer-based models reveals distinct patterns of performance across training, validation, and testing phases. The analysis focuses on the convergence behaviors, generalization capabilities, and overall efficacy of these models in cyber threat classification tasks.

4.1. Model Convergence and Loss Analysis

The loss curves for all three models—LSTM, BiLSTM, and Transformer—demonstrate the distinct learning trajectories and convergence patterns over 10 epochs. The LSTM model, characterized by its unidirectional processing of sequential data, exhibited a steady decrease in training loss from 1.4 to 0.15, while the test loss declined from 1.25 to 0.69. Despite this progress, the LSTM model's test loss plateaued around 0.69, indicating challenges in minimizing generalization error.

In contrast, the BiLSTM model, which processes data in both forward and backward directions, showed slightly improved performance, with training loss reducing from 1.4 to 0.17 and test loss from 1.25 to 0.65. The reduction in test loss relative to LSTM highlights BiLSTM's superior capacity to generalize, likely due to its ability to capture bidirectional dependencies in the data.

The Transformer model, leveraging self-attention mechanisms, demonstrated the most pronounced improvements. Its training loss decreased from 1.4 to 0.12, and test loss from 1.3 to 0.55. The superior convergence rate and lower final test loss underscore the Transformer's robust ability to model complex patterns without significant overfitting. This advantage is attributable to the model's attention-based architecture, which allows for better

contextual understanding and long-range dependency modeling. The loss curves are depicted in Figure 3.

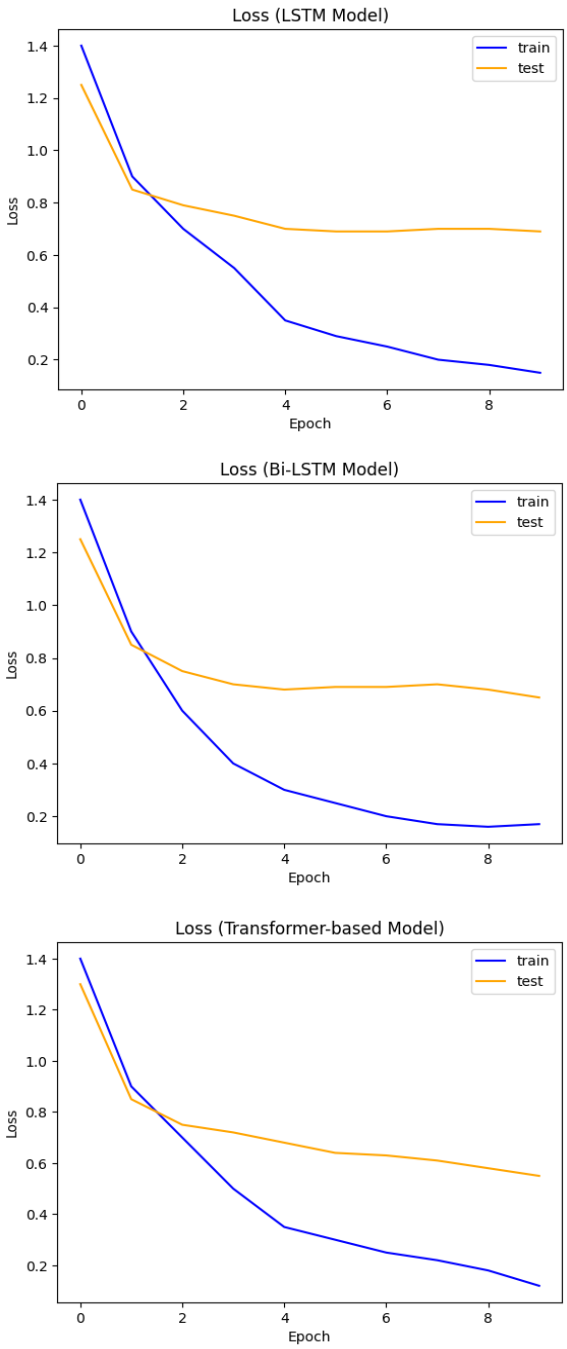

Figure 3. Loss vs Epoch Curve for train and test.

4.2. Performance Evaluation

4.2.1. Training Performances

The Transformer model outperformed both the LSTM and BiLSTM architectures across all metrics. Specifically, the training accuracy for the Transformer reached 98.22%, compared to 95.36% for LSTM and 95.95% for BiLSTM. Validation accuracy followed a similar trend, with the Transformer achieving 90.35%, while the BiLSTM and LSTM models recorded 86.24% and 85.58%, respectively.

The analysis of training loss further emphasizes the superiority of the Transformer model, which recorded a loss of 0.12, in contrast to the LSTM and BiLSTM losses of 0.15 and 0.17, respectively. Validation loss also reflected this trend, with the Transformer achieving a loss of 0.52, while the BiLSTM and LSTM displayed higher losses of 0.67 and 0.70, respectively. In Table 1, the training performances are summarized.

Table 1. Overall training performances of all three trained models.

Metric	LSTM	BiLSTM	Transformer Model
Training Accuracy (%)	95.36	95.95	**98.22**
Validation Accuracy (%)	85.58	86.24	**90.35**
Training Loss	0.15	0.17	**0.12**
Validation Loss	0.70	0.67	**0.52**

4.2.2. Test Performances and Comparison

In this study, we evaluated the performance of various models for our classification task of identifying cyber threats from textual data, comparing them against the benchmark model, SEAM (Ramoliya et al., 2023). The results, summarized in Table 1, highlight the performance metrics of each model across accuracy, precision, recall, and F1 scores. The SEAM model achieved an accuracy of 85.18%, with a precision of 85.34%, a recall of 85.18%, and an F1 score of 85.20%. Our experimental models demonstrated varied performance metrics, with the BiLSTM model performing closely to SEAM, achieving an accuracy of 86.03%, precision of 86.31%, recall of 85.80%, and an F1 score of 86.10%. The LSTM model recorded slightly lower metrics, with an accuracy of 84.97%, precision of 83.89%, recall of 84.45%, and an F1 score of 84.16%. In contrast, our Transformer model outperformed all others, achieving an impressive accuracy of 89.19%, precision of 89.89%, recall of 89.06%, and an F1 score of 89.51%.

These results indicate that while the BiLSTM model shows competitive performance relative to the SEAM benchmark, the Transformer model significantly surpasses all tested models, establishing it as a superior approach for this task. In Table 2, an overall comparison is depicted.

Table 2: Test performance of different models and comparison with the benchmark model.

Model/Metric	Accuracy	Precision	Recall	F1
SEAM [30]	85.18	85.34	85.18	85.2
LSTM	84.97	83.89	84.45	84.16
BiLSTM	86.03	86.31	85.8	86.1
Transformer Model (ours)	**89.19**	**89.89**	**89.06**	**89.51**

4.3. Discussion on Generalization and Model Robustness

The superior performance of the Transformer model in cyber threat classification stems from its architectural advantages and strong generalization capabilities. Unlike recurrent models such as LSTM and BiLSTM, which process inputs sequentially, Transformers employ a self-attention mechanism that enables simultaneous consideration of all tokens in a sequence. This parallelism enhances efficiency and facilitates the capture of long-range dependencies—critical for analyzing cybersecurity narratives that often span multiple sentences and stages.

The multi-head attention mechanism further enriches the model's contextual understanding by attending to multiple aspects of the input simultaneously, which aids in distinguishing subtle threat patterns. Additionally, Transformers avoid the vanishing gradient issues common in RNNs, contributing to faster convergence, lower loss values, and improved generalization, as evidenced by gains in test accuracy and F1-score.

Extensive pre-training on large corpora also bolsters the Transformer's ability to generalize to emerging, previously unseen threat indicators. This is particularly valuable in the dynamic cybersecurity landscape, where language and tactics evolve rapidly.

However, these advantages come with increased computational demands. The high memory and processing requirements of Transformer models may limit their deployment in resource-constrained environments, such as IoT gateways or edge devices. This suggests a need for future work on model compression, distillation, or hybrid architectures to support lightweight, real-time applications. In summary, the Transformer architecture demonstrates robust performance and adaptability in cyber threat classification, making it well-suited for centralized threat intelligence systems and a strong candidate for future research in scalable cybersecurity solutions.

5. Conclusion

This study highlights the transformative potential of machine learning (ML) in strengthening cybersecurity frameworks against increasingly complex digital threats. By conducting a comparative analysis of deep learning architectures—namely LSTM, BiLSTM, and Transformer-based models—our findings underscore the critical role of advanced sequence modeling in detecting and classifying emergent threats such as advanced persistent threats (APTs), zero-day exploits, and AI-driven attacks. The integration of Transformer architectures demonstrated notable improvements in predictive accuracy, generalization, and the timely identification of nuanced threat patterns, outperforming traditional recurrent models. These results affirm the value of attention-based mechanisms in capturing long-range dependencies and contextual nuances present in cybersecurity-related textual data.

In summary, while ML offers significant advancements for cybersecurity, its responsible and effective deployment will require continuous innovation, interdisciplinary collaboration, and rigorous governance to adapt to the ever-evolving threat landscape.

Limitations and Future Work

Despite promising results, the study highlights important limitations. The effectiveness of the models depends heavily on access to high-quality, representative datasets and robust preprocessing pipelines. Additionally, issues such as vulnerability to adversarial inputs and limited interpretability remain pressing challenges for practical deployment in real-world cybersecurity environments.

Future work should explore the development of resilient architectures that can withstand adversarial manipulation, along with explainable AI (XAI) techniques to enhance transparency and trust. Hybrid approaches that combine the interpretability of recurrent models with the performance advantages of Transformer-based architectures may provide a balanced solution. Furthermore, integrating these models into real-time threat detection pipelines, while addressing ethical considerations, data privacy, and responsible deployment, will be essential for advancing the practical application of ML in cybersecurity.

References

[1] W. E. Forum. "Global Risks Report 2023." The World Economic Forum. https://www.weforum.org/publications/global-risks-report-2023/digest/ (accessed 10 Sept, 2024).

[2] D. B. Davis. "ISTR 2019: Internet of Things Cyber Attacks Grow More Diverse." https://symantec-enterprise-blogs.security.com/expert-perspectives/istr-2019-internet-things-cyber-attacks-grow-more-diverse (accessed 30 Aug, 2024).

[3] P. Estes. "Cybercrime Costs Skyrocket to $10.5 Trillion: AI in Cybersecurity Fights Back." virtasant. https://www.virtasant.com/ai-today/cybercrime-costs-skyrocket-to-10-5-trillion-ai-in-cybersecurity-fights-back (accessed Sep 1, 2024, 2024).

[4] D. D. Jim Boehm, Charlie Lewis, Kathleen Li, Daniel Wallance. "Cybersecurity trends: Looking over the horizon."

McKinsey & Company. https://www.mckinsey.com/capabilities/risk-and-resilience/our-insights/cybersecurity/cybersecurity-trends-looking-over-the-horizon (accessed.
[5] IBM. "Cost of a Data Breach Report 2021." IBM. https://www.ibm.com/reports/data-breach (accessed 28 Aug, 2024).
[6] I. F. Kilincer, F. Ertam, and A. J. C. N. Sengur, "Machine learning methods for cyber security intrusion detection: Datasets and comparative study," vol. 188, p. 107840, 2021.
[7] B. Naik, A. Mehta, H. Yagnik, M. J. C. Shah, and I. Systems, "The impacts of artificial intelligence techniques in augmentation of cybersecurity: a comprehensive review," vol. 8, no. 2, pp. 1763-1780, 2022.
[8] H. J. a. p. a. Kheddar, "Transformers and large language models for efficient intrusion detection systems: A comprehensive survey," 2024.
[9] T. M. Chen and J.-M. Robert, "The evolution of viruses and worms," in *Statistical methods in computer security*: CRC press, 2004, pp. 289-310.
[10] Symantec. "Symantec Global Internet Security Threat Report Trends for 2008." Symantec enterprise security. https://docs.broadcom.com/doc/istr-global-09-april-volume-xiv-en (accessed 28 Aug, 2024).
[11] N. Sfetcu, *Advanced Persistent Threats in Cybersecurity–Cyber Warfare*. MultiMedia Publishing, 2024.
[12] L. Bilge and T. Dumitraş, "Before we knew it: an empirical study of zero-day attacks in the real world," in *Proceedings of the 2012 ACM conference on Computer and communications security*, 2012, pp. 833-844.
[13] M. Brundage *et al.*, "The malicious use of artificial intelligence: Forecasting, prevention, and mitigation," 2018.
[14] S. Sicari, A. Rizzardi, L. A. Grieco, and A. J. C. n. Coen-Porisini, "Security, privacy and trust in Internet of Things: The road ahead," vol. 76, pp. 146-164, 2015.
[15] K. Hashizume, D. G. Rosado, E. Fernández-Medina, E. B. J. J. o. i. s. Fernandez, and applications, "An analysis of security issues for cloud computing," vol. 4, pp. 1-13, 2013.
[16] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin, K.-Y. J. J. o. N. Tung, and C. Applications, "Intrusion detection system: A comprehensive review," vol. 36, no. 1, pp. 16-24, 2013.
[17] S. Furnell, *Computer insecurity: Risking the system*. Springer Science & Business Media, 2005.
[18] K. J. N. s. p. SCARFONE, "Guide to Intrusion Detection and Prevention Systems (IDPS)," 2007.
[19] I. Androutsopoulos, J. Koutsias, K. V. Chandrinos, G. Paliouras, and C. D. J. a. p. c. Spyropoulos, "An evaluation of naive bayesian anti-spam filtering," 2000.
[20] A. Yaqoob. "Watson for Cyber Security." IBM. https://www.ibm.com/blogs/nordic-msp/watson-cyber-security/ (accessed 2 September 2024).
[21] A. L. Buczak, E. J. I. C. s. Guven, and tutorials, "A survey of data mining and machine learning methods for cyber security intrusion detection," vol. 18, no. 2, pp. 1153-1176, 2015.
[22] M. Alabadi and Y. Celik, "Anomaly detection for cyber-security based on convolution neural network: A survey," in *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, 2020: IEEE, pp. 1-14.
[23] D. Dasgupta, Z. Akhtar, S. J. T. J. o. D. M. Sen, and Simulation, "Machine learning in cybersecurity: a comprehensive survey," vol. 19, no. 1, pp. 57-106, 2022.
[24] G. Kim, S. Lee, and S. J. E. S. w. A. Kim, "A novel hybrid intrusion detection method integrating anomaly detection with misuse detection," vol. 41, no. 4, pp. 1690-1700, 2014.
[25] V. Sowinski-Mydlarz, J. Li, K. Ouazzane, V. J. T. o. C. S. Vassilev, and C. Intelligence, "Threat intelligence using machine learning packet dissection," 2021.
[26] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, A. J. C. Hotho, and security, "A survey of network-based intrusion detection data sets," vol. 86, pp. 147-167, 2019.
[27] B. Biggio and F. Roli, "Wild patterns: Ten years after the rise of adversarial machine learning," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 2154-2156.
[28] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. J. A. c. s. Pedreschi, "A survey of methods for explaining black box models," vol. 51, no. 5, pp. 1-42, 2018.
[29] B. D. Mittelstadt, P. Allo, M. Taddeo, S. Wachter, L. J. B. D. Floridi, and Society, "The ethics of algorithms: Mapping the debate," vol. 3, no. 2, p. 2053951716679679, 2016.
[30] F. Ramoliya, R. Kakkar, R. Gupta, S. Tanwar, and S. Agrawal, "SEAM: Deep Learning-based Secure Message Exchange Framework For Autonomous EVs," in *2023 IEEE Globecom Workshops (GC Wkshps)*, 2023: IEEE, pp. 80-85.