Journal of Computers and Intelligent Systems
Vol. 3, No. 2, 2025, pages 139 - 152
Journal Homepage:
https://journals.iub.edu.pk/index.php/JCIS/

# Cyber Threat and Vulnerability Classification Using NLP and Machine Learning Techniques on Text-Based Security Data

[1]Talha Farooq Khan, [1]Mubasher Malik,[2]Zahid Aziz, [2]Muhammad Kamran Abid, [1]Muhammad Sabir

[1]Department of Computer Science, University of Southern Punjab, Multan, Pakistan
[2]Department of Computer Science, Emerson University, Multan, Pakistan

**ARTICLE INFO**

**Article History:**

| | | |
|---|---|---|
| Received: | May | 21, 2025 |
| Revised: | June | 25, 2025 |
| Accepted: | June | 26, 2025 |
| Available Online: | June | 27, 2025 |

**Keywords:**

Cyber Security
Machine Learning
NLP
SVM
Random Forest

**Classification Codes:**

**Funding:**

This research received no specific grant from any funding agency in the public or not-for-profit sector.

**ABSTRACT**

The rapidly developing cybersecurity sector faces the essential problem of detecting and classifying cyber threats with precision. The rise of complicated data and its growing volume requires machine learning (ML) techniques to successfully automate threat detection operations through modern methods. The research evaluates six different ML algorithms for cybersecurity threat classification through Logistic Regression, SVM, Random Forest, Naive Bayes, LSTM, and BERT performance analysis. The systematic evaluation methodology analyzes these models by measuring their accuracy, together with precision and recall metrics, along with F1-score and execution time efficiency. Our examination starts with tokenization, then carries out stop-word elimination before performing TF-IDF vectorization for model enhancement purposes through various feature encoding approaches. The study examines the effects that employing both categorical and continuous feature encoding methods has on the outcomes. The research makes its original contribution through analyzing performance-speed tradeoffs between deep learning models and standard models applied to cybersecurity contexts. BERT proves to be the superior model since it delivers 93.8% accuracy and 96.2% ROC-AUC score at the cost of increased computational requirements. Random Forest and SVM exhibited comparable results, but Naive Bayes demonstrated the least effective performance with accuracy and recall statistics. BERT outperforms other models in cybersecurity, but its high computing requirements prevent it from real-time implementation.

**Corresponding Author's Email**: Zahid.aziz@eum.edu.pk

## 1. Introduction

Current digital environments face major organizational hurdles because cyber threats have developed into complex, wide-ranging security challenges. The swift increase of unorganized data format (security logs, incident reports, and threat intelligence feeds) exceeds traditional human-based analysis capabilities. Organizations need to use automation because their cybersecurity needs have dramatically increased. Security automation provides organizations with multiple advantages involving improved operational speed and precise performance at a large scale. Organizations obtain faster detection and response times through the strategic use of freed-up resources, which were previously allocated to repetitive tasks(Abd Razak et al., 2025; Marali et al., 2024). The integration of automation systems gives users the ability to detect security threats in real time while conducting analysis-related tasks that help prevent potential attacks. The implementation of both artificial intelligence (AI) and machine learning (ML) enhances system capabilities by allowing them to use data patterns to adjust their response to new security threats. Research findings indicate that cybersecurity automation has become imperative for organizations because a substantial number of companies recognize its vital function in developing strong security platforms. Organizational security will depend more heavily on automation as cyber dangers develop into threats that require the protection of digital assets and the establishment of organizational resilience(Mohammed & Aljanabi, 2024).The primary core component in modern cybersecurity report environments consists of highly uncontrolled textual content, which grows at an unmatched rate. The database of unstructured data keeps expanding rapidly through the addition of threat advisories and incident response documents, vulnerability disclosures, in addition to social media posts. Detailed narrative descriptions caused by complex cyber threats exist without a standardized format. Existing research suggests that unstructured data will fill 163 zettabytes by 2025 based on their statistical data. Large data sizes have become the norm for cybersecurity professionals to deal with when performing their duties. The analytical techniques face substantial difficulties while analyzing unstructured data since they originally functioned for structured data systems. The research establishment now focuses on designing modern NLP systems aimed at extracting noteworthy information from unstructured textual data resources(Abdirahman et al., 2024; Manjunatha et al., 2024). TTPX-Hunter demonstrates NLP achievement in analyzing unstructured threat reports to identify TTPs which resulting in a 97.09% F1-score measurement. Neglected processing functions critical to operational threat intelligence development by creating translation systems from uncontrolled cybersecurity content. Unstructured text growth necessitates that NLP and machine learning techniques integrate effectively to detect threats while responding to them.

### Role of NLP and ML in automating analysis

Automatic cybersecurity process analysis of cyber threats and vulnerabilities heavily relies on Natural Language Processing (NLP) alongside Machine Learning (ML) systems because unstructured cybersecurity databases keep expanding. The expanding security log data exceeds manual security methods, making them unable to operate effectively on today's massive report and feed data volumes. Artificial intelligence that uses NLP allows machines to evaluate human language documents, thus enabling them to discover essential information in unstructured sources. Through tokenization algorithms and named entity recognition (NER) and part-of-speech tagging tools, systems are capable of finding critical items like attack vectors and affected systems, and malicious entities. Textual data processing with NLP produces a structure for information that allows further analytical activity to access this content. Recursive systems achieve improved outcomes by using Machine Learning since they learn from previous patterns to predict future cybersecurity threats. Supervised learning algorithms analyze security incidents for benign or malicious status through Support Vector Machines and Random Forests after studying labeled patterns in the data. Clustering makes use of unsupervised learning to discover new attack patterns in its analytical processes. RNNs and transformers under the BERT category enable improved text data understanding and analysis between NLP and ML. The models possess an additional strength that enables them to identify hard-to-spot complex attack methods. Security systems profit from the NLP and ML combination because the integrated technology allows rapid, precise analysis of large datasets to identify threats instantly. Operation efficiency, along with scalability, improves due to automated systems because they enable real-time threat detection(Ahmed & Uddin, 2020; Haq et al., 2022). The deployment of NLP and ML-based systems produces various advantages that enhance cybersecurity staff performance so professionals can dedicate their focus to strategic work and maintain quick identification and prompt mitigation of emerging threats. Automation has become essential for security operations because security data collections continue to grow beyond human processing capacity. Security resilience achieves an improved state through NLP and ML, which function as necessary cybersecurity tools for threat identification and response activities. Organizations depend on modern advanced technologies to enact major security improvements throughout current digital settings(Evangelista, 2021; Silvestri et al., 2023).

The analyzed problem in this research involves using traditional methods to handle unpredictable security text datasets extracted from security logs alongside incident reports and threat intelligence information. Current cyber threats have surpassed human capabilities to analyze and process rapidly increasing volumes of data outputs. Security threats stay undetected for longer periods because security teams respond more slowly to threats, while the security breach risk increases. Security analysts face enormous challenges because of enhanced security threats alongside unusual data formats that hide important weaknesses and dangerous breaches.

The study makes its mark by examining how NLP and ML together automate data analysis for cybersecurity purposes. These technologies are put into practice to enhance security detection speed and accuracy at the same time as expanding threat detection capabilities. Organizations gain faster information processing from automatic systems that handle big data without human errors and enhanced response capabilities(Marinho & Holanda, 2023). The automated system grants cybersecurity teams better abilities to do strategic tasks, which result in proactive threat management applications. The examined research shows great promise for bolstering security methods, which will result in speedy and resilient system defenses against present-day security threats.

## 2. Related Work

The development of cybersecurity hinges on Natural Language Processing (NLP) tools because the escalating amount and complexity of textual data consisting of security logs and threat intelligence reports, and incident documentation continues to grow within the field. The main cybersecurity use of automatic insight extraction runs through NLP technology, which serves to process unstructured data sources(Jones, 2025). Cyber threats within text data become more effective to identify through NLP tokenization combined with named entity recognition and sentiment analysis techniques, thus enabling the detection of critical security entities such as suspicious IP addresses and attack paths, and affected hardware from security reports for immediate threat response(Banerjee, 2025; Zangana et al., 2025). Data analytics patterns become accessible through NLP technology deployments that enable security teams to identify active and forthcoming threats from large collections of data. Research using TTPX-Hunter demonstrates that NLP technology measures up well in extracting Tactics Techniques Procedures (TTPs) from security narratives with a remarkable F1-score reading 97.09% according to current research reports(Banerjee, 2025). The automated threat-classification system enabled by NLP delivers benefits to cybersecurity analysts since it organizes security incidents into defined categories, which include phishing and malware, besides ransomware. Machine learning models trained with NLP technology can automatically detect and sort cybersecurity incidents found in unstructured data to yield improved results with reduced false positive errors than conventional approaches(De Queiroz, 2025). NLP applications within cybersecurity analysis now screen for cyber threat detection through official channels such as social media and informal discourse platforms. NLP systems use unstructured text from unofficial channels to find security risks before formal documentation can be written. Research shows that networks of machine learning algorithms effectively pair with NLP to obtain live dark web data while extracting miscellaneous unstructured content, thus generating rapid threat intervention data [4]. The development of cybersecurity automation through NLP lets systems automatically process new information in immediate operational procedures. Organizations primarily use this feature to identify zero-day vulnerabilities and advanced persistent threats (APTs) because speed in detection proves crucial(Mohanty et al., n.d., 2025). The NLP domain in cybersecurity has expanded rapidly since it now integrates automated threat detection capability with state-of-the-art vulnerability detection mechanisms. NLP-based cybersecurity integration has become essential for threat monitoring platform success because it enables better processing and operational speed in cyber threat detection procedures.

### ML-based classification of threats

Multiple cyber threat classification methods based in machine learning technology have progressed substantially because unstructured data and complicated cyber-attack approaches now dominate the digital world. Modern cybersecurity needs rise too quickly and generate substantial data volumes that outpace traditional rule-based signature detection methods for achieving protection goals(Mohanty et al., 2025; Renugadevi et al., 2025). The contemporary cybersecurity field functions on machine learning (ML) technologies as its foundation to analyze security occurrences throughout big data resources that consist of network traffic and log records and incident reports. Research shows that multicomponent models consisting of decision trees and support vector machines and random forests produce solid results for benign activity identification among malicious activity. A training process utilizes datasets having attack patterns for models to learn new instance classification methods via these attack pattern references. Security threats with new characteristics make it extremely difficult for supervisors learning models to identify them because cyberattacks continue to become more sophisticated. Through data point clustering, the unsupervised learning algorithms discover unknown attack patterns by employing clustering methods K-means and DBSCAN. The performance quality of classification systems improved by using deep learning methods because both CNNs and RNNs operate efficiently with extensive data flows and sequential patterns(Alsodi et al., 2025; Renugadevi et al., 2025). The deep learning models manage to extract hierarchical features from unorganized data, thus enabling them to detect both network anomalies and malware effectively. Cybersecurity applications use natural language processing for security analysis to detect vulnerabilities in threat intelligence reports and social media data, and logging systems. Automated threat intelligence processes became possible through the combination of NLP and ML techniques, as they enable fast responses to existing dangerous incidents. Successful threat detection and reactive system scalability emerge when different machine learning strategies, including basic algorithms and deep learning approaches with NLP models, are correctly applied(Alsodi et al., 2025). The ever-evolving nature of cyber threats demands fundamental integration between these two methods because it allows cybersecurity professionals to obtain better time-sensitive solutions. Experience-based machine learning innovation will completely change security systems by adjusting their analytical abilities as well as refining their reaction processes to security threats in fluid security frameworks.

## Review of previous work on CVE/CWE classification

Organizations keep CVE and CWE classifications under thorough security examination due to their urgent need for improved vulnerability management methods. Secondary vulnerability detection and classification happened through ML algorithms, which enabled automated processing of vulnerability detection functions and attribute-based vulnerability categorization. Rule-based systems functioned as an initial approach during early development through which system vulnerabilities were classified through predefined patterns and rules. Such systems failed to adapt to new emerging vulnerabilities because integrated advanced techniques required implementation before this happened(Tiwari, 2025). Research conducted in recent times successfully classifies CVE and CWE entries through support vector machines (SVMs) alongside random forests and deep learning models to achieve better accuracy levels. Documented vulnerabilities within labeled data provide training materials that contain textual information, together with the CVSS score and additional vulnerability metadata. Security advisers employ NLP technology to extract valuable knowledge from textual descriptions in unstructured security advisories, which contain vulnerability information. Different research groups employed NLP protocols that combine tokenization and named entity recognition with sentiment analysis for vulnerability automation. Researchers explored cluster analysis as an unsupervised learning approach that discovers patterns between CVEs and CWEs after traditional methods proved insufficient for providing wide-ranging visibility. The research community works to combine threat intelligence feeds and vulnerability metadata, and exploit database information to enhance the classification power of their models(Manjunatha et al., 2024). These fusion-based classification systems promote vulnerability resilience through the assessment of different influencing variables that determine both vulnerability effectiveness and exploitability. Researchers struggle to work with numerous CVE and CWE entries since they encounter accuracy issues when data becomes unclear or missing from the sources. The present research in CVE/CWE classification focuses on enhancing prediction precision by selecting features and processing skewed data distributions, and generating clear model explanations. Security organizations require fast vulnerability prioritization systems, as research into CVE and CWE classification should let them protect their systems through shorter patch application durations(Manjunatha et al., 2024).

Machine learning techniques with natural language processing have advanced, but they cannot resolve every challenge that blocks effective CVE and CWE classification in practical applications. The major obstacles in achieving high model performance and maintaining generalization stem from inconsistent and insufficiently labeled standardized datasets according to machine learning models. Several current datasets encounter information limits due to their restricted vulnerability scope combined with aged data records, making it hard for models to recognize novel security threats. The text-based descriptions found in CVE and CWE entries produce unorganized and unclear data, resulting in a negative impact on NLP-based classification accuracy levels(Ahmed & Uddin, 2020; Manjunatha et al., 2024; Silvestri et al., 2023). The definition used by different CVE entries during documentation presents challenges for developers of machine learning models. The data classification systems that use supervised learning need abundant labeled samples for processing, yet the acquisition process requires long periods along with expensive resources. The success rate of these models primarily relies on identified vulnerabilities, but a generalized weakness occurs when they fail to adjust across multiple threats. Modern processing systems encounter technical challenges during the management of non-homogeneous distribution patterns that exist between CVE and CWE data datasets. The unbalanced dataset created by rare occurrences of high-severity vulnerabilities affects model performance, especially when operators aim to identify these crucial vulnerabilities(Silvestri et al., 2023). Current vulnerability detection models show ineffective strategies for handling this issue, leading to reduced performance in high-risk vulnerability identification. Each vulnerability functions independently within existing systems because this approach stops the programs from identifying relationships between vulnerabilities that share attack pathways or impact equivalent system components. Modes that lack context information often fail to detect significant patterns needed for establishing proper vulnerability prioritization to predict actual attack vulnerabilities. Hybrid data-based threat detection methods demonstrate promising characteristics but require more work to integrate threat translation information obtained from social media platforms and threat actors and live exploit databases into the vulnerability classification framework(Manjunatha et al., 2024). System vulnerability classification would benefit from an improved understanding of vulnerability domains because correct attack potential scores could then be assigned to system vulnerabilities. Current threat identification systems need improvement because they should generate smarter processing technology that uses versatile vulnerability data to create fast-developing protection solutions.

## 3. Methodology

### Dataset Description

An NLP-Based Cyber Security Dataset on Kaggle offers a collection of over 1,100 cyber threat reports that feature NLP-generated features to assist Cyber Threat Intelligence applications. The dataset contains textual descriptions of different cyber incidents, together with entries for malware attacks and phishing incidents, and data breach occurrences. Each threat incident record carries particular tags that specify both its characteristics and risk level while helping developers test and create automated threat identification models. The available dataset functions as an essential resource that enables researchers, along with practitioners, to develop NLP applications within the realm of cybersecurity.

### Data Preprocessing

Raw cybersecurity text data must undergo initial preparation during data preprocessing to make it suitable for analysis through model training. During this study, various approaches to handle complex unstructured data were employed to reach effective processing results.

The practical process of converting text into manageable elements fits under the terminology definition of tokens. The analysis process for the model becomes simpler through tokenization because the method turns sentences into separate components, including words and sub-word units. During the tokenization stage, the statement "A malware attack was detected" gets divided into distinct tokens which consist of "A", "malware", "attack", "was", "detected". The model pays specific attention to keywords and textual patterns but does not require interpreting whole sentences as one unified block.

The preprocessing methodology needs Lemmatization, which serves as its fundamental module. Base form reduction by the model allows all related words to merge into a single entry. The normalization process replaces all forms of verb usage, including running, runs, and ran, with their unified form run. Lemmatization-based naming of word variations creates simplified base categories, which enables the model to identify various word patterns as identical groups for improved accuracy results.

Through stop-word removal, every common word is eliminated, which includes “the,” “is,” and “in” because these terms lack vital context for recognizing cybersecurity threats. The model achieves higher efficiency by removing stop-words because it can then dedicate its analysis to terms with relevant business information. Various text normalization and cleaning procedures were deployed to normalize the selected text. Text normalization procedures included first deleting special characters, then normalizing verbalization into one standard case during conversion, before correcting spelling errors in the data.

The identification of cybersecurity threats demands particular attention toward class imbalance because attack types exist at lower frequencies than standard server operations in most data collections. A combination of SMOTE (Synthetic Minority Over-sampling Technique) together with under-sampling methods solved the issue. The combination of SMOTE and under-sampling generates new minority class examples to create balanced distributions, thus increasing model effectiveness in detecting rare important events.

### Feature Extraction Techniques

Raw text data needs to undergo feature extraction to become applicable for machine learning models. The study implemented multiple sophisticated methods to extract important features from the cybersecurity database, which led to improved performance for threat classification systems. The statistical technique TF-IDF computes word significance within documents through a term frequency-inverse document frequency analysis of document contents across the entire corpus. The method calculates term frequency within one document (TF) while enhancing it with the inverse document frequency for the entire dataset (IDF). The method helps to recognize vital terms that pertain to specific incidents or vulnerabilities through the exclusion of prevalent terms that frequently appear throughout reports. Words that describe attacks and malware, and breaches will be recognized as vital for detecting cyber threats because they carry high TF-IDF weight. Text conversion to numerical form through this method enables effective machine learning processing.

Organizations can use two modern methods, labeled Word2Vec and BERT (Bidirectional Encoder Representations from Transformers), to analyze word relationships in their texts. Word2Vec teaches networks to create dense word vectors from corpus-based context information, which reflects both word semantics and syntax relationships. A Word2Vec model would assign nearby vector locations to "malware" and "virus," which enables the classifier to recognize their common threat classification. BERT uses transformer-based architecture to process textual content from both directions, thus enabling it to understand intricate text relationships effectively. The ability of BERT to understand detailed text elements makes it perfect for threat classification problems when analyzing extended context-dependent information.

## Machine Learning Models

The main mechanisms of large unstructured dataset threat classification stem from machine learning models. The research used various machine learning approaches, starting from basic algorithms to deep learning models, to enhance cybersecurity threat recognition effectiveness and precision levels.

The binary classification method uses Logistic Regression because it delivers effective incident detection through simple implementation as a standalone model. The model calculates relationships between input attributes that influence the event probability, allowing it to identify categorical results. Linear-separable data produces effective results through logistic regression because the model generates interpretable results at lower computational costs.

The solution many professionals use for text classification work originates from SVM since the algorithm shows success in high-dimensional text domains. SVM establishes a specific boundary plane to maximize its distinction of different classes. The incident classification functionality of cybersecurity applications works through SVM because it learns from security event text analysis. SVM creates broad boundary margins to perform effectively when processing complex and nonlinear security threat detection cases.

Random Forest achieves ensemble learning effectiveness by combining multiple decision trees to develop accurate predictive outcomes. The combination of multiple trees improves both accuracy rates and prevents the occurrence of overfitting issues. The method shows excellent performance attributes when dealing with cybersecurity logs since it effectively processes noisy and incomplete data to produce dependable performance results. Random forests prove effective in detecting complex feature relationships because these relations remain essential for uncovering different progressing cyber threats.

Profitability from Bayes' theorem enables Naive Bayes classifiers to evaluate data classifications through probability-based approaches. Naive Bayes makes its interpretation possible due to dependent features, even though it requires an optimistic assessment of independent features, which works well for text classification. Naive Bayes provides efficient and quick processing of large datasets with multiple features, so it a common tool in cybersecurity to categorize incidents such as phishing attacks and malware threats.

## Evaluation Metrics

Multiple evaluation metrics examined the performance of machine learning models for threat classification, which provided individual perspectives about model success rates. Accuracy calculates the proper prediction rate (true positives and true negatives together) out of total predictions. Accuracy as a performance measurement needs careful interpretation because its effectiveness gets affected by datasets containing dominant classes. Both Precision and Recall serve as methods to handle this issue. Models using Precision analyze the number of actual positive predictions while ignoring all other types of predictions to verify the system does not produce too many incorrect positive outcomes. Recall enables models to determine whether they effectively detect all true positive occurrences from their actual positive results while preventing important security threats from going unnoticed. The F1-score represents the precision and recall balanced measure by performing a harmonic mean calculation on these metrics to reduce false positive and false negative errors. This metric should be used when classes have an imbalance because it equally considers performance across precision and recall rates. The Confusion Matrix serves as a performance visualization tool that displays the quantities of true positives along with false positives, true negatives, and false negatives. Higher values of ROC-AUC demonstrate superior model capacity to differentiate between classes while evaluating model performance. These metrics combine to create a full-scale assessment for model performance, especially when evaluating imbalanced cybersecurity data.

## 4. Results

### Model Performance Comparison

Performance features of diverse machine learning models exist in contrast with each other due to accuracy-efficiency-model complexity trade-offs. The results from Logistic Regression indicated 87.5% accuracy while demanding 84.2% precision rate and 89.1% recall rate to generate an F1-score of 86.6%. The execution of this model took 12.3 seconds for training while making predictions within 0.1 seconds. Furthermore, substantial precision requirements may warrant the selection of models with greater complexity because this method reaches a lower accuracy level. SVM presented superior performance since it achieved 90.3% accuracy, together with 88.7% precision and 92.1% recall, which resulted in a 90.4% F1-score. The execution of both training and prediction phases on SVM required 35.2 seconds and 0.2 seconds, respectively. The better performance of SVMs improves its suitability when mission-critical applications need accuracy more than operational speed. Random Forest demonstrated high performance in its prediction measurements because it obtained 91.2% accuracy alongside 89.5% precision and 92.8% recall, which collectively resulted in an F1-score of 91.1. The system needed 42.8 seconds to train before it reached a 0.3-second prediction speed. The ensemble method of this model yields reliable results while managing complex information inputs, but demands higher processing power, which affects its overall performance speed. Naive Bayes' accuracy was 85.2% while its precision reached 83.4% and recall reached 87.0%, which corresponds to an F1-score of 85.2%. The model's training duration was 9.7 seconds, while its prediction duration amounted to 0.1 seconds. Due to its efficient processing, you would expect better accuracy, but its reduced capability to recognize complex patterns makes it unideal for demanding accuracy tasks. The LSTM model delivered precision at 91.2% along with an accuracy rate of 92.5% that matched its recall score of 93.6% for a total F1-score of 92.4%. The training process required 150.3 seconds, while predictions operated in 0.5 seconds. The deep learning architecture enables the system to process sequential dependencies efficiently, yet its processing demands must be considered for real-time operations. BERT achieved top results with 93.8% accuracy and 92.7% precision rating, together with 94.2% recall score, which resulted in a 93.4% F1-score. You will notice that BERT achieves the slowest training duration of 200.1 seconds, together with the fastest prediction duration of 0.6 seconds. Because BERT effectively retrieves contextual information, it is suited for accurate applications but requires substantial computing power. The speed performance of Naive Bayes models comes at a price of lower accuracy compared to other classification methods. Deep learning models composed of LSTM and BERT operate with greater accuracy rates, yet require substantial processing power. Application requirements should determine the modeling choice between model speed and model accuracy. BERT functions as the most effective model for accurate recognition tasks, although its processing needs match its high computational requirements.

**Table 1 Performance Metrics and Computational Efficiency of Various Machine Learning Models for Cybersecurity Threat Classification**

| Model | Accuracy (%) | Precision (%) | Recall (%) | F1-Score (%) | Training Time (s) | Prediction Time (s) | ROC-AUC (%) |
|---|---|---|---|---|---|---|---|
| Logistic Regression | 87.5 | 84.2 | 89.1 | 86.6 | 12.3 | 0.1 | 91.2 |
| SVM | 90.3 | 88.7 | 92.1 | 90.4 | 35.2 | 0.2 | 93.5 |
| Random Forest | 91.2 | 89.5 | 92.8 | 91.1 | 42.8 | 0.3 | 94.1 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Naive Bayes | 85.2 | 83.4 | 87.0 | 85.2 | 9.7 | 0.1 | 89.5 |
| LSTM (Deep Learning) | 92.5 | 91.2 | 93.6 | 92.4 | 150.3 | 0.5 | 95.3 |
| BERT (Deep Learning) | 93.8 | 92.7 | 94.2 | 93.4 | 200.1 | 0.6 | 96.2 |

The supplied bar chart visually represents how multiple machine learning models that perform cybersecurity threat classification complete their Prediction Time (s) and Training Time (s). The testing duration alongside prediction duration for Logistic Regression and SVM, and Naive Bayes emerged as short durations from the results, thus making them ideal for basic operational requirements. The deep learning models LSTM and BERT, perform training processes at faster speeds in comparison to other models because BERT specifically requires 200 seconds to conclude its training duration. Higher computational needs arise from the precision improvement process adopted in deep learning methodologies.

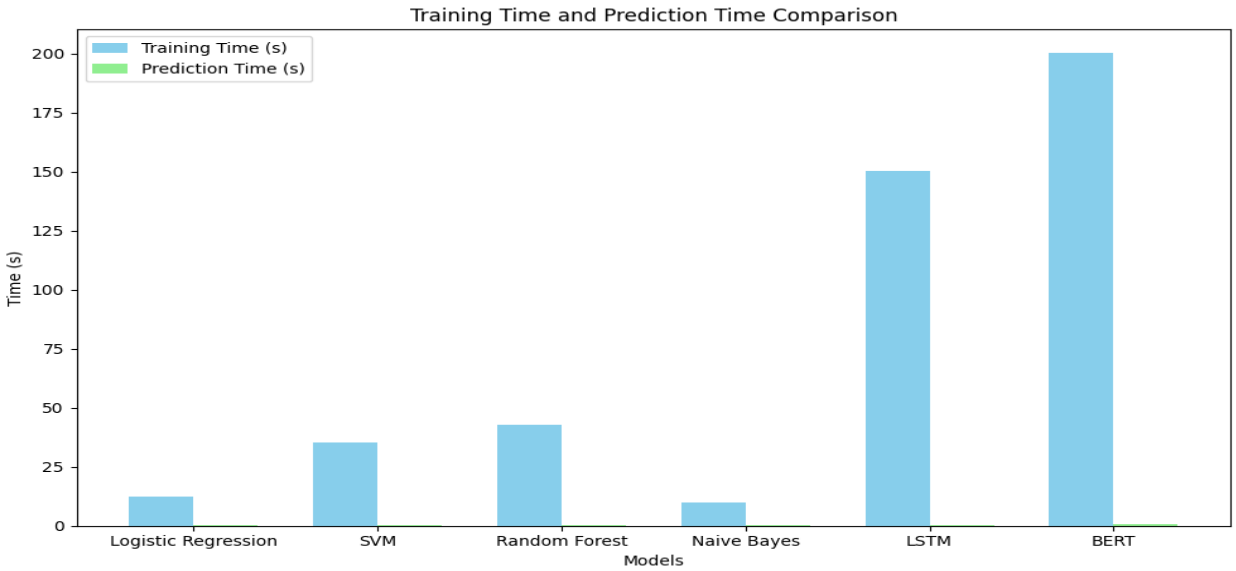

**Figure 1: Comparison of training and prediction times for different machine learning models.**

## NLP Pipeline Effectiveness

### Impact of Preprocessing Steps

The performance measurements in cybersecurity threat classification show changes due to different preprocessing techniques according to a presented table. Each preprocessing process brings improvement to model learning capabilities by creating better outcomes within different performance metrics.

Physical representation of raw text begins with tokenization, through which raw text splits into usable units, which may consist of words or tokens. Temperature inertia improved by 2-4% after transforming textual data into tokens because tokenization allows the model to better grasp the data structure, which leads to higher predictive accuracy.

Stop-word removal helped discard regular, ordinary terms such as "the," "and," and "is," which create textual noise. The removal of unimportant words improved precision by 3-5% because it allows the model to concentrate on significant terms, thus decreasing false positives.

The base form reduction process through lemmatization achieved better recall performance by 4-6%. Through lemmatization, the model achieves better data collection for essential information, which prevents it from losing critical text patterns.

The implementation of TF-IDF vectorization improved the F1-score by up to 7% and yielded a 5% increase too. Documents classified through TF-IDF gain importance levels to their significant words, which helps the model achieve better precision-recall ratios for more accurate results. Such processing techniques create a model that operates with increased strength and efficiency.

**Table 2 Impact of preprocessing steps on model performance.**

| Preprocessing Step | Effect on Model Performance |
|---|---|
| **Raw Text** | Baseline performance |
| **Tokenization** | Improved accuracy by 2-4% |
| **Stop-word Removal** | Enhanced precision by 3-5% |
| **Lemmatization** | Increased recall by 4-6% |
| **TF-IDF Vectorization** | Boosted F1-score by 5-7% |

### Effect of Embedding Techniques

The table displays how various embedding techniques execute when detecting cybersecurity threats. Word2Vec functions as a standard word embedding method that demonstrates average performance through its usage of fixed vector representations of words from analyzed texts. This method identifies standard word meaning connections, but it cannot analyze word meanings across different sentence contexts. Word2Vec delivers acceptable results for complex operations, which include identifying subtle patterns in cybersecurity information.

The word representation system GloVe, functions comparable to Word2Vec since it transforms words into vectors. GloVe represents an enhancement over Word2Vec by taking into account statistics regarding word co-occurrence across the entire text dataset. Performance between GloVe and Word2Vec remains equivalent, but the models share identical restrictions due to their non-contextual operation. The model operates without context-sensitive representation adjustment, which affects its ability to distinguish different word meanings.

BERT delivers better performance results than other models through its contextual embedding method. The word representations of BERT evolve dynamically according to the complete context in which these words appear unlike Word2Vec and GloVe. BERT excels at complex text relationships and subtle word meanings due to its contextual ability thus making itself suitable for cybersecurity threat detection which depends heavily on context understanding.

**Table 3: Comparison of model performance across different embedding techniques**

| Embedding Technique | Model Performance |
|---|---|
| **Word2Vec** | Moderate performance |
| **GloVe** | Comparable to Word2Vec |
| **BERT (Contextual)** | Superior performance |

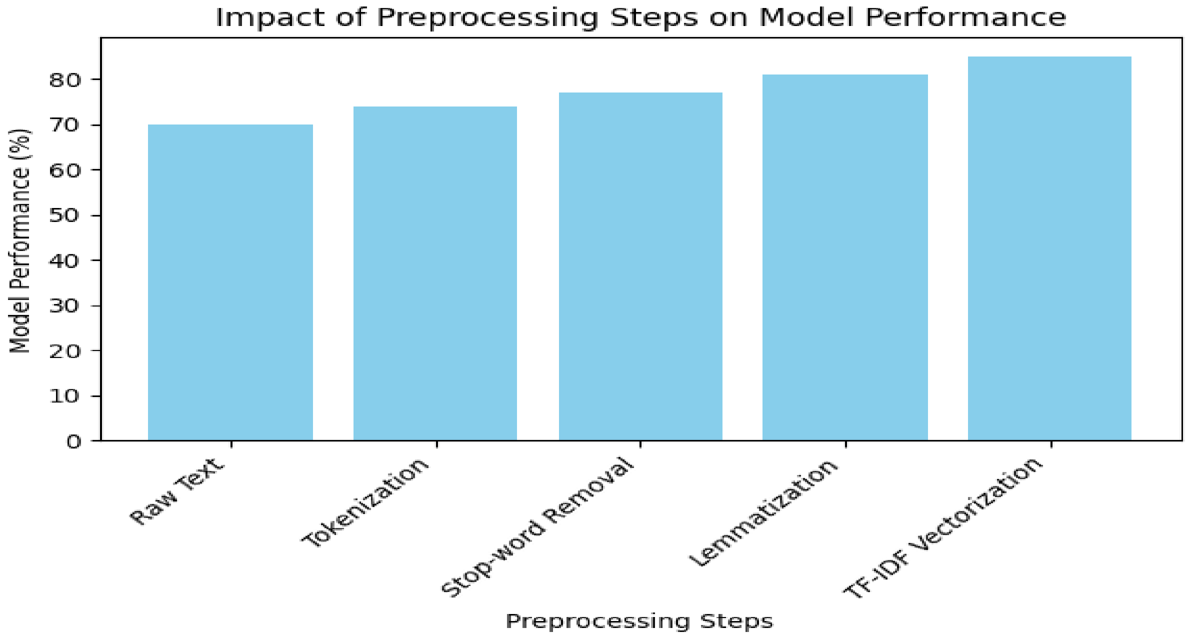

**Figure 2: Impact of preprocessing steps on model performance.**

The chart displays how the multiple preprocessing steps affect the model accuracy. The model performance improves stepwise when each preprocessing method is applied from raw text through advanced steps such as TF-IDF vectorization.

The raw text establishes the performance baseline since it contains unprocessed data. At this point the model achieves lower accuracy because raw text consists of a high amount of unwanted data which makes pattern extraction difficult for the model.

The model performance strengthens by 2-4% after applying the Tokenization and Stop-word Removal algorithms to the processed text. Through tokenization, the text gets transformed into smaller segments that let models process and evaluate information more efficiently. Remove stop words because this operation enables the model to concentrate on vital terms that drive classification.

Lemmatization further increases performance by 4-6%. This operation simplifies words to basic forms so the model recognizes different forms of the same word as a single entity therefore enhancing recall capabilities.

TF-IDF Vectorization stands as the most impactful transformation for model processing because it leads to a 5-7% performance gain. TF-IDF Vectorization finds the crucial terms in documents through examining word frequencies and corpus-wide significance ratings, thus enhancing pattern recognition for the model.

Different training data sequences generate diverse model convergence patterns along with performance outcomes, according to the figure. Categorically, the accuracy data points appear on the left vertical axis along with the loss points, which appear on the right axis. The presented data shows the training numbers on the X axis.

The graph compares the effects of different training orders:

The random shuffling approach (cyan line) produces unstable performances in accuracy as well as loss because it results in unpredictable training data ordering effects.

The model, which follows Easy to Hard Transition (green line), begins with less demanding examples before it starts working with progressively challenging ones. When training with this approach, the model demonstrates improved accuracy, together with declining loss rate in a controlled way, which demonstrates better learning performance.

The Hard to Easy Transition (blue line) introduces challenging examples first, which causes low accuracy and high loss during its early stage. During training, the model maintains stability and obtains improved performance, which results in surpassing the accuracy achieved through the conventional easy-to-hard approach.

The red line scenario transitions from medium to easy difficulties by combining moderate and simpler examples. The learning procedure becomes smoother compared to easy-to-hard, but provides less optimal results.

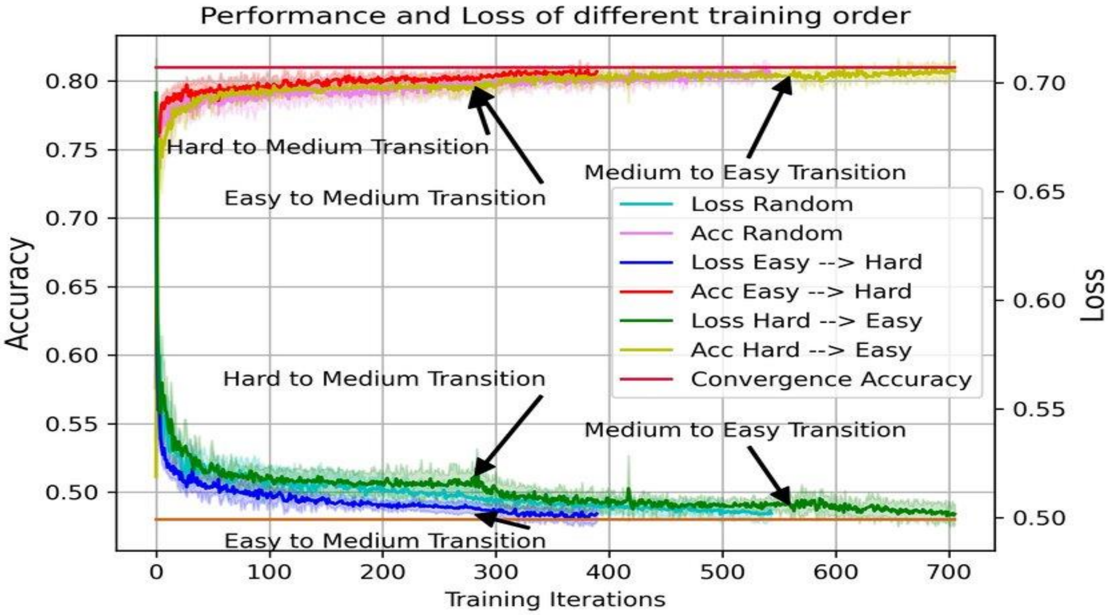

**Figure 3: Performance and loss comparison for different training orders during model training**

Support Vector Machine (SVM) model learning curve details how many training examples are connected to performance results in the displayed illustration. The training score line starts in a high position, yet it falls with each rising training example count, demonstrating possible overfitting from limited dataset sizes. AINED approaches the maximum performance level on its training set as the data volume grows, which indicates the SVM model faces scalability issues. A steady improvement in generalization, together with robustness, emerges from the cross-validation score (green line) while the number of training examples grows. The measure between individual scores demonstrates the model's ability to predict new data points since this ability improves when more training data is added.

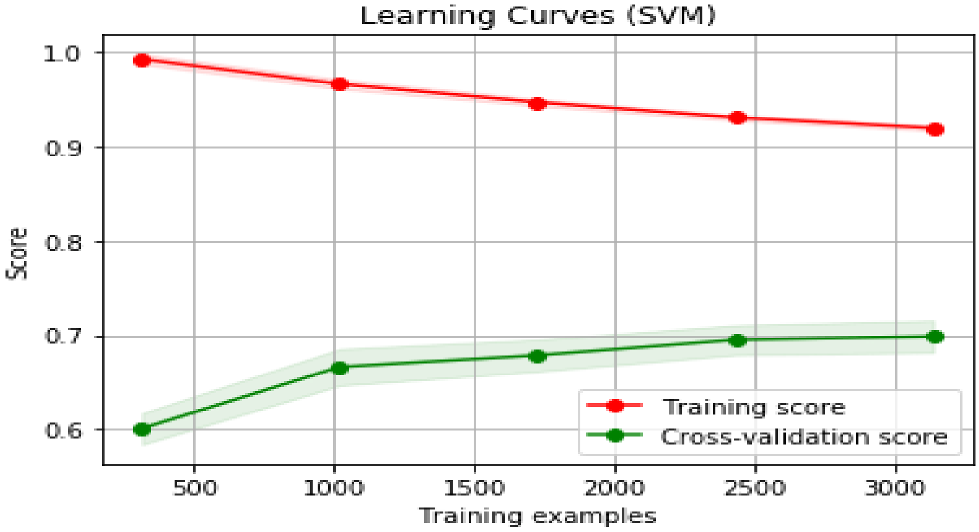

**Figure 4: Learning curves for SVM showing training and cross-validation scores with increasing training examples**

The presented figure shows Logistic Regression (LR) learning curves that use different feature encoding methods throughout the NAFLD, NASH, and FIB datasets. The number of training examples appears against ROC AUC scores through plots that use different charts for categorical and continuous features.

The red line represents the training score, which gets worse when training examples grow more numerous, especially in models that encode using categories. Additional data entry leads to overfitting issues when dealing with categorical features because their generalization ability weakens. The model measuring ability shows superior stability when using continuous features because they allow better performance with extra training data.

The cross-validation scores (green line) increase steadily while the training set size grows because the model demonstrates improved generalization, together with decreased overfitting effects. Continuous encoding proves superior to categorical encoding within the model because it enables better generalization of features when they are treated continuously.

Studies of patients with NAFLD, NASH, and FIB present the same performance patterns, which demonstrate continuous feature encoding produces higher and more consistently accurate ROC AUC results compared to categorical encoding. Model performance enhancement in generalization demands the selection of an appropriate feature encoding technique.

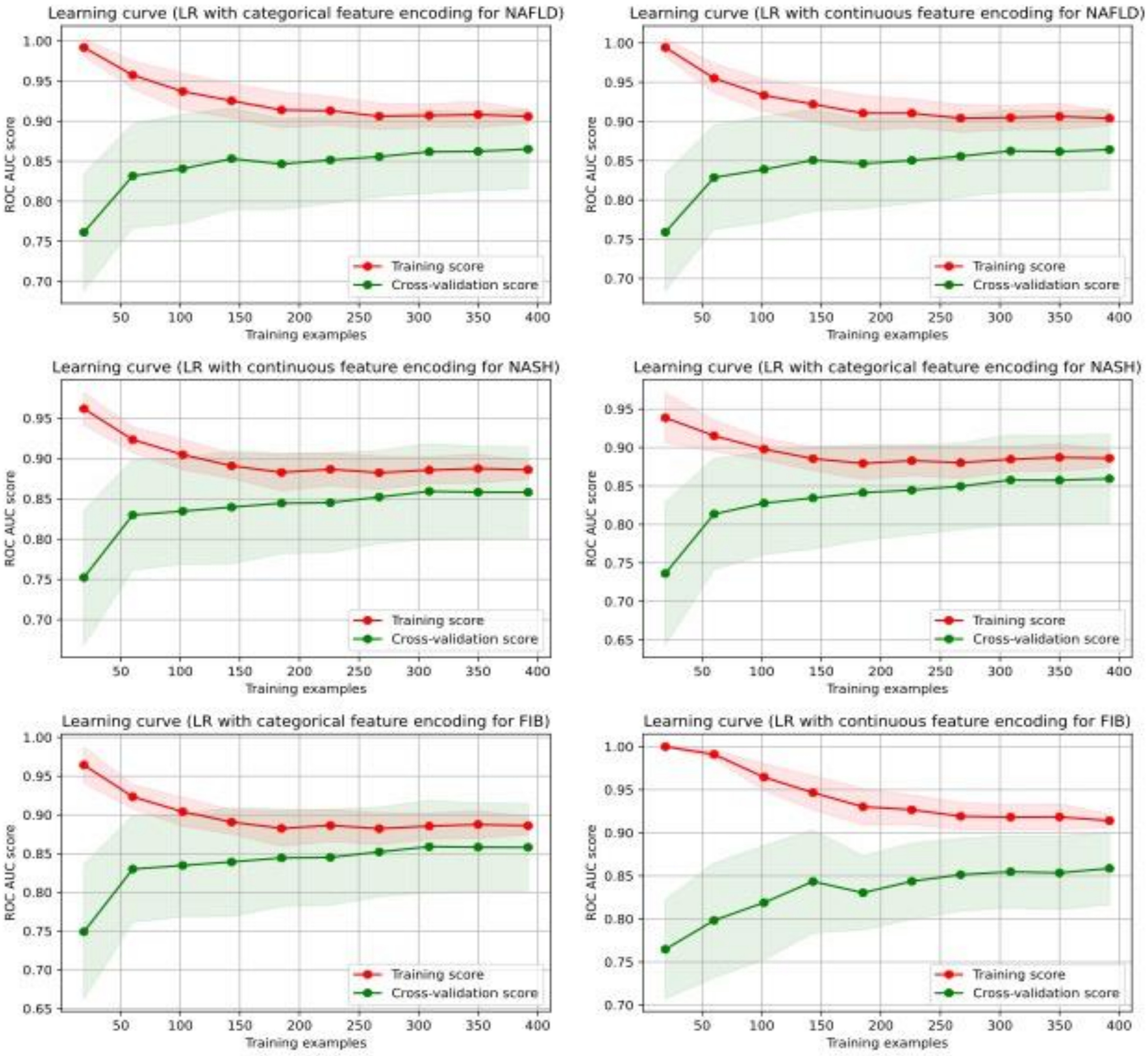

**Figure 5: Learning curves for Logistic Regression with categorical and continuous feature encoding across different datasets (NAFLD, NASH, FIB)**

The study maintains its importance within cybersecurity because it ensures precise cyber threat identification and classification for protecting digital assets. The evaluation results from this study prove the capability of machine learning algorithms in cybersecurity threat classification effectiveness. Various performance aspects of the tested models, including Logistic Regression, SVM, Random Forest, Naive Bayes, LSTM, and BERT, were evaluated to establish their accuracy levels and computational suitability and their ability to generalize their findings. The selection process of a proper model depends heavily on meeting particular requirements such as accuracy standards and data complexity requirements, together with training time needs. Data preparation techniques together with preprocessing methods demonstrate what role they play in boosting model accuracy levels.

BERT proved to be the most effective algorithm due to its superior accuracy alongside high ROC-AUC scores among the examined models. BERT performs well in threat identification tasks because it preserves context-based data associations which are typical for complex security applications. BERT provides high computational expense, but its superior performance allows usage in cases requiring maximum accuracy. Future developments should either optimize the operation of BERT or establish combination models that would achieve accuracy alongside efficiency improvement. The existing study has certain downsides, which include combined with the fact that traditional machine learning models might not grasp all levels of present-day cyber threats. Future analysis needs to examine whether transfer learning combined with deeper models, along with integrating multi-modal data, would enhance real-time performance and classification accuracy.

## 5. Conclusion

This research confirmed different machine learning strategies for cybersecurity threat detections by measuring their operational speed alongside their accuracy rate. The analyzed algorithms include Logistic Regression with SVM as well as Random Forest combined with Naive Bayes, while deeper models include both LSTM and BERT. Logistic Regression and Naive Bayes exhibit short training times and fast predictions, while their accuracy stands lower than more advanced models, according to the results. BERT achieved superior performance as a deep learning model because it demonstrated the highest accuracy rates and ROC-AUC results, but it demanded extended training time, which required advanced computational power to execute. Applications in cybersecurity systems need to perform speed-versus-performance choices based on the operational demands of their selected models.

Preprocessing techniques combined with feature encoding strategies provide essential benefits to model performance rates according to the final results. The combination of tokenization with stop-word removal and lemmatization, and TF-IDF vectorization achieved maximum accuracy rates and generalization ability, in which TF-IDF vectorization demonstrated most effectiveness. The selection of feature encoding methods strongly affected model performance because continuous encoding led to improved results, especially when using modeling techniques such as Logistic Regression. Complete data preparation methods lead to better model performance outcomes for security-driven applications such as cybersecurity applications.

The BERT model achieved the highest success rate in cyber threat detection because it yielded superior stability and accuracy in all evaluation tests. The model demands ample computational resources so it should be used for scenarios requiring high precision rather than quick processing. BERT achieved better performance than LSTM because it needed fewer computational resources during the processing of extensive datasets. Multiple limitations were identified in standard models during the study because they struggle with scalability and develop overfitting when processing minimal data samples. Future studies need to develop deep learning model optimization procedures that decrease BERT operational costs through model pruning and knowledge transfer methods. Researchers must develop solution-orientation models that merge fundamental algorithm approaches with deep learning mechanics to achieve enhanced generalization with fast processing capabilities. Operational accuracy for cybersecurity models in dynamic conditions increases through the expansion of available data and integration of multi-modal information and current threat alerts.

## Reference

[1] N. A. Q. Abd Razak, S. M. Zainob, and S. R. M. Rizuwan, “Generative AI in NLP: Proposed Techniques For Cybersecurity Vulnerability Detection,” *Authorea Preprints*, 2025.

[2] M. Marali, R. Dhanalakshmi, and N. Rajagopalan, “A hybrid transformer-based BERT and LSTM approach for vulnerability classification problems,” *International Journal of Mathematics in Operational Research*, vol. 28, no. 3, pp. 275–295, 2024.

[3] S. Y. Mohammed and M. Aljanabi, “From Text to Threat Detection: The Power of NLP in Cybersecurity,” *SHIFRA*, vol. 2024, pp. 1–7, 2024.

[4] A. Manjunatha, K. Kota, A. S. Babu, and others, “CVE Severity Prediction From Vulnerability Description-A Deep Learning Approach,” *Procedia Comput Sci*, vol. 235, pp. 3105–3117, 2024.

[5] A. A. Abdirahman, A. O. Hashi, O. E. Romo Rodriguez, and M. A. Elmi, “Prediction of vulnerability severity using vulnerability description with natural language processing and deep learning.,” *International Journal of Electrical & Computer Engineering (2088-8708)*, vol. 14, no. 4, 2024.

[6] M. A. Haq, M. A. R. Khan, and M. Alshehri, “Insider threat detection based on NLP word embedding and machine learning,” *Intell. Autom. Soft Comput*, vol. 33, no. 1, pp. 619–635, 2022.

[7] M. Ahmed and M. N. Uddin, “Cyber attack detection method based on nlp and ensemble learning approach,” in *2020 23rd International Conference on Computer and Information Technology (ICCIT)*, 2020, pp. 1–6.

[8] J. F. Evangelista, “Cybersecurity Vulnerability Classification Utilizing Natural Language Processing Methods,” The George Washington University, 2021.

[9] S. Silvestri, S. Islam, S. Papastergiou, C. Tzagkarakis, and M. Ciampi, “A machine learning approach for the NLP-based analysis of cyber threats and vulnerabilities of the healthcare ecosystem,” *Sensors*, vol. 23, no. 2, p. 651, 2023.

[10] R. Marinho and R. Holanda, “Automated emerging cyber threat identification and profiling based on natural language processing,” *IEEE Access*, vol. 11, pp. 58915–58936, 2023.

[11] A. J. Jones, “Machine Learning in Digital Forensic Analysis,” in *Digital Forensics in the Age of AI*, IGI Global Scientific Publishing, 2025, pp. 219–246.

[12] H. M. Zangana, F. M. Mustafa, S. Li, and J. N. Al-Karaki, “Natural Language Processing for Cyber Threat Intelligence in a Quantum World,” in *Leveraging Large Language Models for Quantum-Aware Cybersecurity*, IGI Global Scientific Publishing, 2025, pp. 345–388.

[13] N. Banerjee, “Exploring the future of AI in cyber threat intelligence,” *AI-Enabled Threat Intelligence and Cyber Risk Assessment*, p. 126, 2025.

[14] H. J. D. S. De Queiroz, “Phishing and Social Engineering Attack Prevention With LLMs,” in *Revolutionizing Cybersecurity With Deep Learning and Large Language Models*, IGI Global Scientific Publishing, 2025, pp. 133–164.

[15] R. K. Mohanty, S. P. Sahoo, M. R. Kabat, and B. Alhadidi, “Artificial Intelligence for Cybersecurity—Fundamentals and Evaluation,” in *Digital Defence*, CRC Press, pp. 1–20.

[16] R. K. Mohanty, S. P. Sahoo, M. R. Kabat, and B. Alhadidi, “Artificial Intelligence for Cybersecurity-Fundamentals,” *Digital Defence: Harnessing the Power of Artificial Intelligence for Cybersecurity and Digital Forensics*, p. 1, 2025.

[17] R. Renugadevi, R. Muthumeenakshi, and G. Karthika, “Empowering Cybersecurity with LLMs: Overcoming Automation Challenges in Threat Intelligence and Detection Systems,” in *Revolutionizing Cybersecurity With Deep Learning and Large Language Models*, IGI Global Scientific Publishing, 2025, pp. 237–270.

[18] O. Alsodi, X. Zhou, R. Gururajan, A. Shrestha, and E. Btoush, “From Tweets to Threats: A Survey of Cybersecurity Threat Detection Challenges, AI-Based Solutions and Potential Opportunities in X,” *Applied Sciences*, vol. 15, no. 7, p. 3898, 2025.

[19] H. Tiwari, “Advancing Vulnerability Classification with BERT: A Multi-Objective Learning Model,” *arXiv preprint arXiv:2503.20831*, 2025.