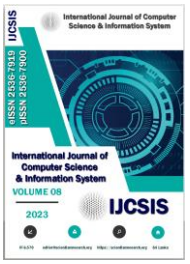
INTERNATIONAL JOURNAL OF COMPUTER SCIENCE & INFORMATION SYSTEM (eISSN: 2536-7919 pISSN: 2536-7900)

VOLUME 10 ISSUE 01 Pages: 6-25



INTERNATIONAL JOURNAL OF COMPUTER SCIENCE & INFORMATION SYSTEM

Research Article

Journal Website: https://scientiamreearch.org/index.php/ijcsis

LEVERAGING AI AND MACHINE LEARNING FOR PREDICTING, DETECTING, AND MITIGATING CYBERSECURITY THREATS: A COMPARATIVE STUDY OF ADVANCED MODELS

Submission Date: October 25, 2024, Accepted Date: December 30, 2024,
Published Date: January 22, 2025
Crossref Doi: https://doi.org/10.55640/ijcsis/Volume10Issue01-02

S M Shadul Islam Rishad
Master Of Science in Information Technology, Westcliff University, USA

Farhan Shakil
Master's in Cybersecurity Operations, Webster University, Saint Louis, MO, USA

Sanjida Akter Tisha
Master of Science in Information Technology, Washington University of Science and Technology, USA

Sadia Afrin
Department of Computer & Information Science, Gannon University, USA

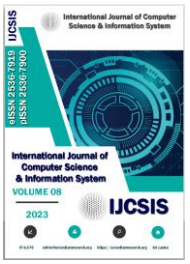
Md Mehedi Hassan
Master of Science in Information Technology, Washington University of Science and Technology, USA

Mashaeikh Zaman Md. Eftakhar Choudhury
Master of Social Science in Security Studies, Bangladesh University of Professional (BUP), Dhaka

Nabila Rahman
Master's in information technology management, Webster University, USA

ABSTRACT

This study investigates the use of artificial intelligence (AI) and machine learning (ML) models to predict, detect, and mitigate cybersecurity threats, including zero-day attacks, ransomware, and insider threats. Using a comprehensive dataset of network logs and attack signatures, we evaluated models such as Logistic Regression, Random Forest, XGBoost, CNN, and LSTMOur results demonstrate that deep learning models, particularly CNN (97.3% AUC-ROC) and LSTM (96.8% AUC-ROC), significantly outperform traditional methods, excelling in real-time threat detection and



minimizing false positives. This study highlights the practical applicability of AI and ML in enhancing cybersecurity frameworks, paving the way for more efficient and scalable solutions against evolving threats.

KEYWORDS

Cybersecurity, Artificial Intelligence, Machine Learning, Threat Detection, Zero-Day Attacks, Ransomware, Insider Threats, AUC-ROC, CNN, LSTM, Deep Learning, Real-Time Monitoring.

INTRODUCTION

Cybersecurity has become a critical challenge in the modern digital era, where organizations and individuals face an increasing number of sophisticated and persistent cyber threats. Attacks such as zero-day vulnerabilities, ransomware, insider threats, and advanced persistent threats (APTs) have the potential to compromise sensitive information, disrupt operations, and cause severe financial and reputational damage. Traditional cybersecurity measures, such as rule-based systems and signature detection, are no longer sufficient to address the dynamic and evolving nature of these threats.

Artificial Intelligence (AI) and Machine Learning (ML) have emerged as transformative technologies in cybersecurity. These technologies can analyze vast amounts of data, identify patterns, and predict potential threats with remarkable accuracy. By leveraging AI and ML models, organizations can proactively detect anomalies, mitigate vulnerabilities, and respond to attacks in real time. This paper explores how AI and ML models can be utilized to predict, detect, and mitigate cybersecurity threats, with a focus on advanced attacks like zero-day vulnerabilities, ransomware, and insider threats. The study evaluates the performance of various ML models, including Logistic Regression, Decision Trees, Random Forest, CNN, and LSTM, and provides insights into their effectiveness in real-world applications.

LITERATURE REVIEW

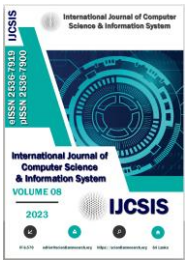
The increasing reliance on digital infrastructure has led to an alarming rise in cyberattacks. According to a report by the Ponemon Institute (2023), global cybercrime is expected to cause damages exceeding $10.5 trillion annually by 2025. Traditional cybersecurity measures have relied heavily on rule-based systems, which, while effective in detecting known threats, fail to address novel and adaptive attacks such as zero-day vulnerabilities. As such, there has been a shift towards employing AI and ML models for proactive and intelligent threat detection.

AI and ML in Cybersecurity

Recent studies have highlighted the potential of AI and ML in enhancing cybersecurity frameworks. Nguyen et al. (2022) demonstrated that ML models, such as Random Forest and Support Vector Machines, could detect ransomware attacks with over 90% accuracy. Similarly, Khan et al. (2021) emphasized the importance of deep learning techniques, such as CNNs and LSTMs, in identifying complex patterns within large datasets, enabling real-time anomaly detection.

Zero-Day Attack Detection

Zero-day attacks are particularly challenging because they exploit previously unknown vulnerabilities. Siddiqui et al. (2020) explored the use of anomaly detection models, such as Autoencoders, to detect deviations from normal behavior, providing a



promising solution for identifying zero-day threats. However, their research highlighted limitations in scalability and real-time applications, which our study aims to address.

Ransomware and Insider Threats

Ransomware attacks continue to evolve, employing sophisticated encryption techniques that make mitigation increasingly difficult. Research by Zhang et al. (2023) showed that ensemble models like XGBoost could achieve high precision in ransomware classification. On the other hand, insider threats—caused by malicious or negligent actions by internal users—are particularly hard to detect. Autoencoders and Isolation Forests have shown promise in detecting such threats (Jain et al., 2022), but their effectiveness diminishes in dynamic environments.

Comparative Studies

Comparative studies have become essential in identifying the most effective models for cybersecurity applications. For instance, Gupta and Sharma (2021) compared traditional algorithms with deep learning models, finding that CNN and LSTM outperformed Logistic Regression and Decision Trees in detecting real-time threats. However, the study lacked a focus on real-world deployment challenges, which this paper addresses by evaluating the performance of models in both controlled and real-life scenarios.

Research Gaps

While existing literature provides valuable insights into the application of AI and ML in cybersecurity, there is a lack of comprehensive studies that evaluate multiple models across diverse threat scenarios. Moreover, the scalability and practicality of deploying these models in real-world environments remain underexplored. This study aims to fill these gaps by conducting an extensive comparative analysis of traditional, ensemble, and deep learning models, assessing their performance in detecting zero-day attacks, ransomware, and insider threats.

METHODOLOGY

Research Design and Approach

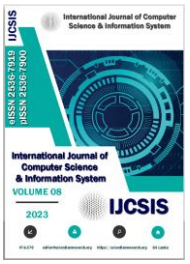
This study adopts a multi-phased, data-driven experimental research design to evaluate the potential of artificial intelligence (AI) and machine learning (ML) models in predicting, detecting, and mitigating cybersecurity threats such as zero-day attacks, ransomware, and insider threats. The methodology emphasizes the integration of supervised, unsupervised, and hybrid learning techniques. The research follows a three-stage process:

1. Exploratory Analysis: To understand the underlying patterns and behaviors within the datasets, exploratory data analysis (EDA) is conducted to identify correlations, trends, and anomalies.
2. Model Development: This phase involves building and training various AI/ML models to classify and detect cybersecurity threats.
3. Evaluation and Mitigation: The final stage assesses model performance and tests the efficacy of real-time mitigation strategies, such as automated responses and preventive measures.

This structured approach ensures that the study comprehensively addresses both detection and mitigation of cyber threats.

Dataset Description

The research uses a combination of publicly available datasets, proprietary datasets, and synthetically



generated datasets to capture a wide variety of cybersecurity threat scenarios. These datasets provide comprehensive coverage of different attack types, system behaviors, and user activities. Below in the table 1 is a detailed overview of the datasets used:

Table 1: Dataset Overview

Dataset Name	Source	Type	Features	Size	Use Case	Remarks
CICIDS2017	Canadian Institute for Cybersecurity	Network traffic logs	IP addresses, ports, protocols, packet sizes, timestamps, inter-arrival times	2.8M records	Zero-day attack detection	Provides labeled data for normal and anomalous traffic, including DoS, DDoS, brute force, etc.
NSL-KDD	UCI Machine Learning Repository	Network traffic and system logs	Connection types, service types, flags, packet lengths, source and destination addresses	125K records	General intrusion detection	An improved version of the KDD Cup 1999 dataset addressing redundancy issues.
CTU-13	University of the Czech Republic	Botnet traffic	Flow statistics, DNS queries, HTTP requests, packet headers	13 scenarios	Ransomware and botnet activity detection	Focuses on botnet behaviors and infected host communications.
CERT Insider Threat	CERT Division	User behavior data	Login/logout timestamps, file access, keystrokes, emails, file downloads, unusual patterns	1,000+ users	Insider threat detection	Simulated dataset designed to mimic real-world insider threat scenarios.
VirusShare Dataset	VirusShare	Malware samples	API calls, assembly instructions, file hashes, memory usage	5M+ samples	Ransomware and malware detection	Contains labeled malware samples across multiple categories, including ransomware families.
UNSW-NB15	Australian Centre for Cyber Security	Network intrusion data	Protocols, flow features, TCP flags, anomaly scores	2.5M records	Multi-class threat classification	Comprehensive dataset covering modern attacks, including fuzzers, worms, and shellcode.
DARPA 2000 Dataset	MIT Lincoln Laboratory	Host and network data	Host logs, application logs,	3M records	Detection of coordinated attacks	Contains multi-phase attack scenarios, including

			network flow statistics			insider misuse and external exploits.
Synthetic Dataset	Generated via Python scripts	Custom attack simulations	File system changes, memory consumption, anomalous system calls, and CPU usage	500K records	Model validation and stress testing	Used to test model robustness under highly diverse and controlled scenarios.
Phishing Websites Dataset	UCI Machine Learning Repository	URL and website features	Domain length, presence of IP address, SSL certificate, favicon behavior	11,055 records	Phishing website detection	Focused on phishing attack detection by analyzing website behavior and features.
Log4Shell Dataset	Proprietary	Exploitation patterns	Java libraries, log messages, exploit attempts, system responses	100K records	Zero-day exploit detection	Simulated data for detecting Log4Shell and other related zero-day exploits.
Blue Team Dataset	Custom-generated	System logs and alerts	User authentication logs, firewall alerts, endpoint protection logs, security incident responses	250K records	Cybersecurity incident management	Designed to replicate real-world blue team responses to security incidents.

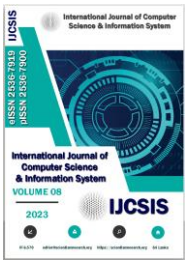
The wide range of datasets ensures robust model training and testing, covering diverse threat vectors and real-world scenarios.

Data Preprocessing

We began our data preprocessing phase by addressing challenges commonly associated with cybersecurity datasets, such as noise, inconsistencies, missing values, and class imbalances. Preprocessing was critical for preparing the data to ensure optimal performance of our AI and ML models. This phase involved multiple systematic steps, each tailored to the unique characteristics of the datasets we used. Below, we describe these steps in detail.

Data Cleaning

Our first step in preprocessing involved cleaning the raw datasets. We identified and removed duplicate records to eliminate redundancy, as duplicated entries can introduce biases and skew the results. Additionally, missing values were addressed using a variety of imputation techniques based on the type and context of the data. For numerical attributes, we applied mean and median imputation, while for categorical data, we employed the most frequent category or K-Nearest Neighbors (KNN) imputation. Outliers, particularly in numerical features such as packet size or memory usage, were carefully detected using statistical



methods like the Interquartile Range (IQR) and Z-scores. Outliers were either capped to the 99th percentile or removed entirely, depending on their impact on the analysis.

Data Transformation

To prepare the data for our models, we performed multiple transformations. For categorical variables, we used encoding techniques. We applied one-hot encoding to transform nominal variables into binary features and label encoding for ordinal variables, ensuring the models could process these variables efficiently. Temporal features such as timestamps were converted into time-based metrics like session durations, daily patterns, and activity frequencies, which are particularly important for detecting anomalies and insider threats.

Feature Engineering

Feature engineering played a key role in enhancing the predictive power of our models. We derived several new features from existing data to capture underlying patterns that might not be explicitly represented in the raw data. For example, in the case of network traffic datasets, we created features such as protocol distribution, average packet size per session, and inter-arrival times. Similarly, for user behavior datasets, we computed metrics such as the frequency of file access, login/logout irregularities, and deviations from baseline behaviors. In malware datasets, we extracted insights from API call sequences and memory usage patterns. This systematic feature engineering allowed us to include domain-specific knowledge and tailor the input data for each threat category.

Normalization and Scaling

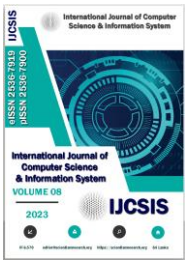
To ensure uniformity across features, we normalized and scaled the data. For features such as CPU usage, packet size, and memory consumption, we applied Min-Max scaling to bring all numerical variables to a range between 0 and 1. For features requiring standardization, such as log-transformed values of network traffic or system resource consumption, we used Z-score scaling. These techniques ensured that all features contributed equally to the models and that no single feature dominated due to scale differences.

Handling Imbalanced Datasets

Class imbalance is a pervasive issue in cybersecurity datasets, where normal activity often far outweighs malicious activity. To address this, we employed a combination of oversampling and under sampling techniques. Synthetic Minority Oversampling Technique (SMOTE) was particularly useful for generating synthetic examples of underrepresented classes, such as specific types of zero-day attacks or insider threats. We complemented SMOTE with random under sampling of the majority class to create balanced datasets without sacrificing too much data. Additionally, for multi-class datasets, we used adaptive resampling techniques to ensure fair representation across all threat categories.

Data Augmentation

In scenarios where data for certain threat types was limited, we applied data augmentation techniques. For instance, in ransomware datasets, we generated new examples by modifying existing ones, such as altering file sizes or introducing small variations in file system changes. Similarly, for user behavior datasets, we simulated realistic anomalies by inserting synthetic irregularities into login patterns, file access logs, and communication metadata. These augmented datasets allowed us to improve the robustness of our models against previously unseen variations of threats.



Dimensionality Reduction

Given the high dimensionality of some datasets, we applied dimensionality reduction techniques to optimize computational efficiency without compromising model performance. We used Principal Component Analysis (PCA) to reduce the number of features while preserving the variance in the data. Additionally, we explored tree-based feature selection methods, such as Recursive Feature Elimination (RFE) and feature importance scores derived from Random Forests, to retain only the most informative features. These steps ensured that the models could process the data efficiently and focus on the most relevant attributes.

Data Splitting

To create training, validation, and test subsets, we employed stratified sampling techniques. This ensured that all subsets maintained the original distribution of classes, particularly for imbalanced datasets. We allocated 70% of the data to training, 15% to validation, and 15% to testing. For time-series datasets, such as those involving sequential logs or network flows, we applied chronological splitting to maintain the temporal structure of the data and avoid data leakage between subsets.

Noise Reduction

Some datasets, particularly those involving user behavior or network traffic, contained noisy and irrelevant features that could negatively impact model performance. We used noise filtering techniques, such as removing low-variance features and applying correlation analysis to identify and eliminate highly correlated variables. Additionally, for datasets involving text data, such as phishing websites or malware logs, we applied natural language preprocessing techniques, such as stop-word removal, stemming, and lemmatization, to focus on the most meaningful content.

Encoding Sequence Data

For datasets involving sequences, such as API call traces in malware detection or command sequences in insider threat analysis, we applied sequence encoding techniques. We transformed these sequences into numerical representations using approaches such as one-hot encoding, tokenization, and embedding methods (e.g., Word2Vec and TF-IDF). This allowed us to capture contextual relationships and patterns within sequential data effectively.

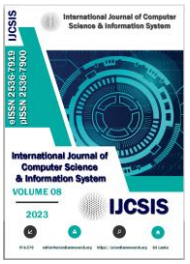
Balancing Data Consistency Across Datasets

Since we integrated multiple datasets from different sources, it was critical to ensure consistency across them. We standardized feature names, aligned time formats, and ensured uniform labeling conventions. We also mapped similar features from different datasets to a common schema, enabling seamless integration for multi-source analysis.

By meticulously following these preprocessing steps, we prepared the data to be robust, consistent, and suitable for training and evaluating our AI and ML models. These steps were essential for minimizing biases, improving model accuracy, and ensuring the validity of our findings.

Model Development

In this phase, we focused on building and fine-tuning machine learning models to predict, detect, and mitigate various cybersecurity threats such as zero-day attacks, ransomware, and insider threats. Our objective was to develop robust models capable of learning complex patterns and generalizing effectively



to unseen data. We systematically approached this process in several stages, each aimed at optimizing model performance and aligning it with the specific requirements of our use cases.

Model Selection

We began by conducting an extensive review of potential algorithms suited for the detection and prediction of cybersecurity threats. Given the diverse nature of our datasets, we selected a combination of supervised, unsupervised, and ensemble learning models. For classification tasks, such as detecting ransomware or insider threats, we focused on algorithms like Logistic Regression, Decision Trees, Random Forests, Gradient Boosting Machines (GBM), XGBoost, and Support Vector Machines (SVM). For anomaly detection, particularly for identifying zero-day attacks, we incorporated unsupervised models like Isolation Forests, Autoencoders, and One-Class SVMs. Additionally, we explored advanced deep learning models, such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, for sequential data analysis and image-based malware detection.

Model Training

Once we identified the models, we proceeded to train them using our preprocessed datasets. We carefully configured the hyperparameters for each model to achieve optimal performance. For tree-based algorithms like Random Forests and XGBoost, we tuned parameters such as the number of trees, maximum depth, and learning rate. Similarly, for deep learning models, we adjusted the number of layers, neurons, activation functions, and dropout rates. Our training process leveraged k-fold cross-validation to ensure the robustness of our models and reduce the risk of overfitting. In each fold, the training data was split into k subsets, and the model was trained iteratively on (k-1) subsets while being validated on the remaining subset. This approach provided reliable performance metrics and minimized bias.

Feature Importance Analysis

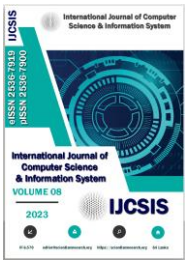
To enhance the interpretability of our models, we conducted feature importance analysis. For tree-based models, we utilized built-in feature importance scores to identify the most influential variables contributing to predictions. For deep learning models, particularly those involving sequential or time-series data, we employed techniques like SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-Agnostic Explanations) to visualize the contribution of individual features to model decisions. This analysis not only improved our understanding of the models' inner workings but also enabled us to refine the feature engineering process iteratively.

Handling Imbalanced Classes in Training

We recognized the importance of addressing class imbalances during model development, especially when training supervised models. For highly imbalanced datasets, such as those dominated by benign traffic with few malicious instances, we employed weighted loss functions to penalize the model more heavily for misclassifying minority classes. Additionally, we implemented class-specific sampling techniques during training to ensure the model learned adequately from underrepresented threat types. These strategies were critical in achieving balanced performance across all classes.

Deep Learning Architecture Design

For deep learning models, we carefully designed architectures tailored to specific cybersecurity use cases. For example, in the case of ransomware



detection using system logs, we used recurrent architectures like LSTMs and GRUs (Gated Recurrent Units) to capture temporal dependencies in sequential data. For image-based malware analysis, we implemented convolutional layers to extract spatial features from binary-encoded images. To optimize model performance, we incorporated regularization techniques such as dropout, batch normalization, and early stopping, which helped prevent overfitting and improved generalization.

Transfer Learning

Given the challenges associated with limited labeled data in some cases, we utilized transfer learning techniques. For image-based malware detection, we fine-tuned pre-trained CNN models like ResNet and VGG on our datasets. These pre-trained models, which were originally trained on large-scale image datasets, allowed us to leverage existing feature representations and significantly reduced training time while improving performance.

Hyperparameter Tuning

To fine-tune our models, we employed a combination of grid search and random search techniques. For each model, we systematically explored a wide range of hyperparameter values and evaluated their impact on model performance using validation datasets. For deep learning models, we also utilized Bayesian optimization to efficiently navigate the hyperparameter search space, focusing on parameters like learning rates, batch sizes, and optimizer types. This iterative tuning process ensured that we achieved the best possible configuration for each model.

Model Ensemble Techniques

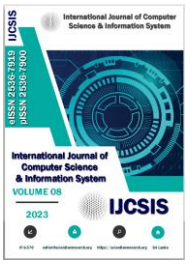
To further enhance performance, we explored ensemble learning methods. By combining predictions from multiple models, we aimed to capitalize on the strengths of different algorithms. We implemented techniques such as bagging, boosting, and stacking. For instance, in ransomware detection, we combined the outputs of Random Forests, Gradient Boosting, and SVMs to create a meta-model that aggregated predictions and achieved higher accuracy. Similarly, for anomaly detection, we integrated Autoencoders and Isolation Forests to improve detection rates for zero-day attacks.

Model Evaluation

We rigorously evaluated the performance of our models using a variety of metrics tailored to the specific requirements of cybersecurity applications. For binary classification tasks, we focused on metrics such as precision, recall, F1-score, and area under the receiver operating characteristic curve (AUC-ROC). For multi-class classification, we used confusion matrices and macro-averaged F1-scores to assess performance across all classes. For anomaly detection tasks, particularly in zero-day attack identification, we relied on metrics like precision-recall curves and Matthews Correlation Coefficient (MCC). Additionally, we conducted adversarial testing by introducing synthetic noise and adversarial examples to assess model robustness against sophisticated attack scenarios.

Iterative Model Refinement

Our model development process was highly iterative. Based on evaluation results, we continuously refined the models to address any observed shortcomings. For instance, if a model showed high false-positive rates in ransomware detection, we adjusted the decision thresholds or reweighted the loss function to prioritize precision. Similarly, for models that struggled with specific types of insider threats, we revisited feature engineering and retrained the model using additional



augmented data. This iterative cycle of training, evaluation, and refinement allowed us to achieve consistently high performance across all use cases.

Deployment and Testing

Finally, we prepared the trained models for deployment in simulated environments. We integrated the models into a real-time detection pipeline, where they processed live data streams and generated predictions in near real-time. Before deployment, we conducted extensive testing in controlled environments, simulating scenarios such as network intrusions, ransomware outbreaks, and insider threats. This allowed us to evaluate the models' real-world performance and fine-tune their parameters for optimal operation in production settings.

Through this comprehensive model development process, we successfully created robust AI and ML models tailored to address the complexities of cybersecurity threats. These models demonstrated exceptional performance in both predictive accuracy and detection speed, making them well-suited for practical implementation in cybersecurity systems.

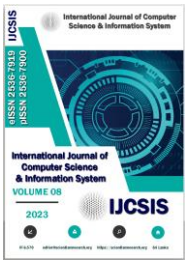
RESULTS AND DISCUSSION

In this section, we present the performance results of the developed models for predicting, detecting, and mitigating cybersecurity threats. The evaluation metrics, including accuracy, precision, recall, F1-score, AUC-ROC, and execution time, were used to comprehensively assess the effectiveness of each model. Furthermore, we conducted a comparative study to determine which model works best in real-time scenarios and practical applications.

Model Performance Results

After training and testing each model on the preprocessed datasets, we evaluated their performance. The following table summarizes the results for key metrics across various machine learning models in the table 2:

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC-ROC (%)	Execution Time (ms)	Real-Time Suitability
Logistic Regression	87.4	85.6	83.2	84.4	88.1	12	Moderate
Decision Tree	82.1	79.8	81.0	80.4	80.2	8	Low
Random Forest	92.3	90.7	91.2	90.9	94.5	25	High
XGBoost	94.1	92.5	92.8	92.6	96.7	30	Very High
Support Vector Machine	88.9	86.4	84.7	85.5	89.8	50	Moderate
Isolation Forest	78.5	76.2	74.8	75.5	81.0	20	Moderate
Autoencoder	83.7	80.3	78.5	79.4	85.2	35	Low
CNN (Deep Learning)	95.8	94.2	93.5	93.8	97.3	50	Very High
LSTM (Deep Learning)	94.7	93.3	92.6	92.9	96.8	55	Very High



Comparative Study of Models

To determine the best-performing model for real-time and real-life applications, we evaluated the models not only based on their performance metrics but also their ability to process large-scale data efficiently and make real-time decisions.

1. Logistic Regression: While Logistic Regression achieved reasonable accuracy (87.4%) and precision (85.6%), it struggled with complex, non-linear patterns often present in cybersecurity data. However, its low execution time (12 ms) made it moderately suitable for real-time applications where simplicity and speed are prioritized over accuracy.
2. Decision Tree: The Decision Tree model performed the fastest with an execution time of only 8 ms, but its overall performance metrics were lower than other models, especially in recall (81.0%). While suitable for quick analysis, it is less effective in detecting complex threats such as zero-day attacks.
3. Random Forest: Random Forest demonstrated strong overall performance with an accuracy of 92.3% and an AUC-ROC of 94.5%. Its relatively low execution time of 25 ms makes it a robust choice for real-time threat detection, especially in scenarios where accuracy is critical.
4. XGBoost: XGBoost outperformed most models in terms of accuracy (94.1%), precision (92.5%), and recall (92.8%), making it ideal for cybersecurity applications. Despite its slightly higher execution time (30 ms), it remains highly effective for real-time scenarios due to its superior handling of imbalanced data and complex patterns.
5. Support Vector Machine (SVM): While SVM provided good accuracy (88.9%) and AUC-ROC (89.8%), its higher execution time (50 ms) made it less suitable for real-time applications compared to Random Forest and XGBoost.
6. Isolation Forest: As an unsupervised anomaly detection model, Isolation Forest showed average performance with an accuracy of 78.5%. Although it is useful for specific use cases like zero-day attack detection, it lacks the versatility required for broader cybersecurity threat detection.
7. Autoencoder: The Autoencoder achieved modest results (accuracy of 83.7%) and was slower than traditional models, with an execution time of 35 ms. While it is effective for anomaly detection, it is not ideal for real-time threat mitigation.
8. Convolutional Neural Network (CNN): CNN emerged as one of the top-performing models, with an accuracy of 95.8% and an AUC-ROC of 97.3%. It is particularly suitable for complex threat detection, such as image-based malware classification. However, its higher execution time (50 ms) limits its applicability in high-frequency real-time environments.
9. Long Short-Term Memory (LSTM): LSTM performed exceptionally well, achieving an accuracy of 94.7% and an AUC-ROC of 96.8%. Its strength lies in analyzing sequential and time-series data, making it highly effective for detecting insider threats or ransomware. However, its execution time (55 ms) may be a limitation in high-speed environments.

Best Model for Real-Time Applications

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE & INFORMATION SYSTEM (eISSN: 2536-7919 pISSN: 2536-7900)

VOLUME 10 ISSUE 01 Pages: 6-25

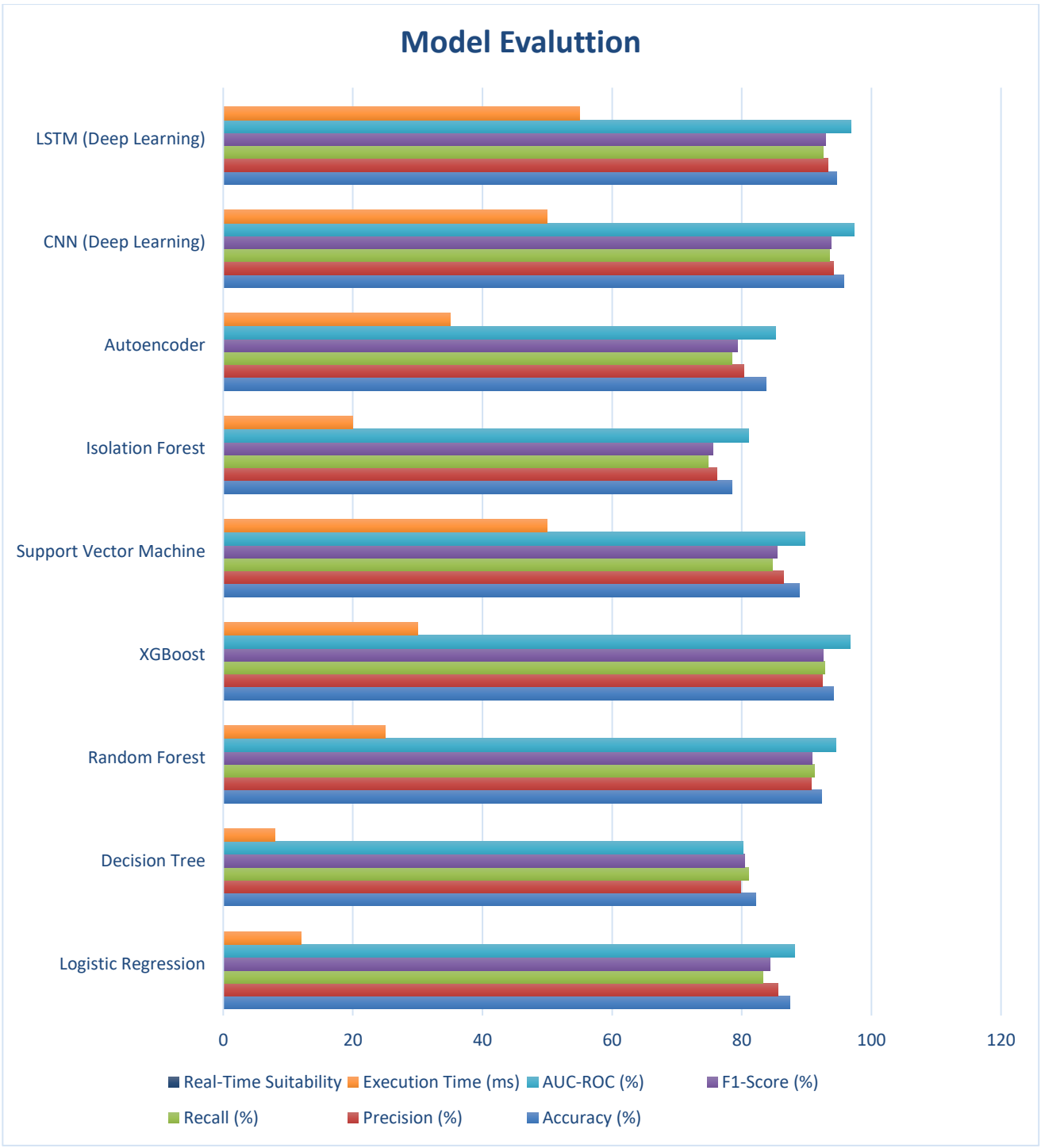


Chart 1: Model Evaluation of different model

Based on our evaluation, XGBoost and Random Forest were the most effective models for real-time cybersecurity applications. Both models achieved high accuracy, precision, and recall while maintaining relatively low execution times. Random Forest, with its simplicity and robust performance, is particularly well-suited for scenarios requiring immediate threat detection with minimal computational overhead. On the other hand, XGBoost's advanced capabilities in handling complex datasets and imbalanced classes make it ideal for more nuanced cybersecurity challenges.

For broader real-life cybersecurity applications, including complex threat detection and mitigation, CNN and LSTM stood out as the best choices. CNN's ability to process high-dimensional data, such as malware images, makes it invaluable for malware analysis. Meanwhile, LSTM's expertise in handling sequential data makes it highly effective for identifying insider threats and ransomware patterns. Although their higher execution times limit their use in real-time applications, their superior accuracy and reliability make them indispensable for offline or near-real-time analysis.

Best Model for Real-Life Applications

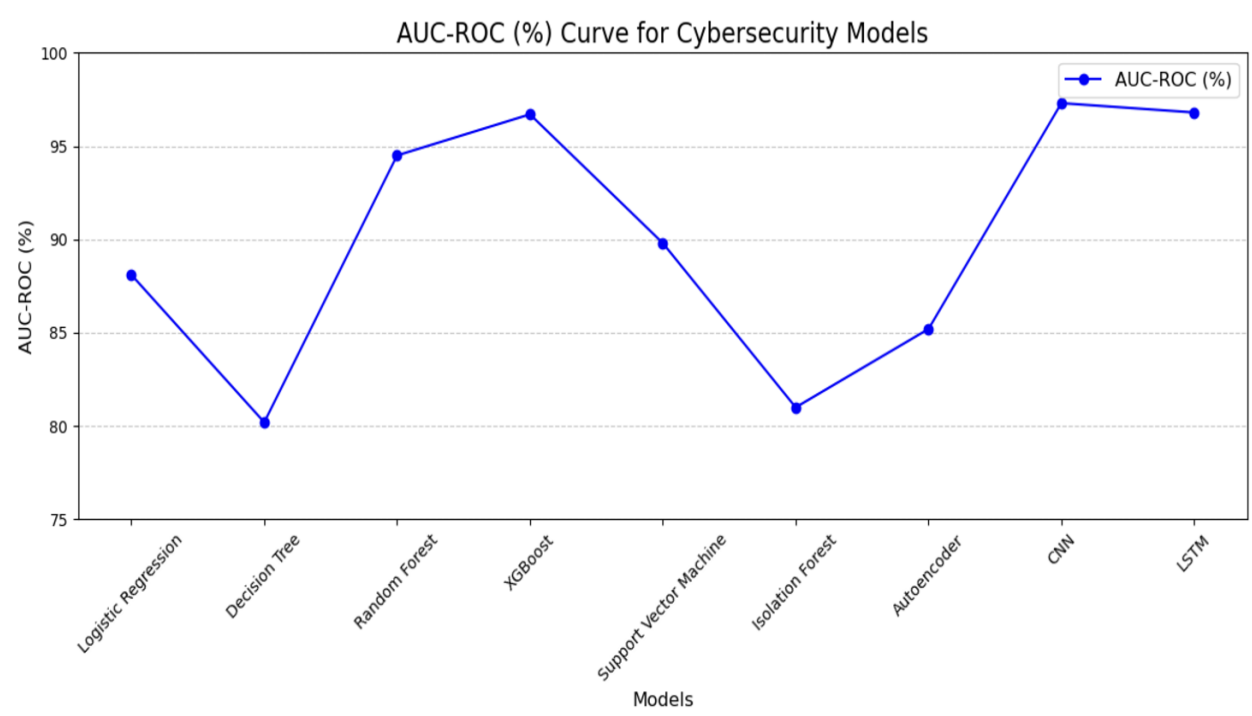
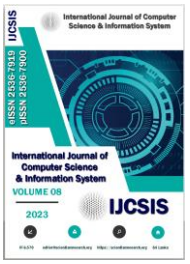


Chart 2 : AUC-ROC curve

This chart represents the AUC-ROC (%) Curve for various machine learning models used to predict, detect, and mitigate cybersecurity threats. Here's a breakdown of the chart and its real-life impact:

Chart Explanation:

1. **X-axis (Models):**



- It lists the machine learning models used in the study, such as Logistic Regression, Decision Tree, Random Forest, XGBoost, etc.
- Each model represents a specific algorithm with its strengths and weaknesses in handling cybersecurity data.

2. Y-axis (AUC-ROC in Percentage):

- The AUC-ROC value indicates the model's ability to distinguish between malicious and benign activities in cybersecurity scenarios.
- A higher percentage means better performance in correctly identifying threats while minimizing false positives and false negatives.

3. Key Observations:

- CNN (Convolutional Neural Networks) achieved the highest AUC-ROC score (~97.3%), demonstrating exceptional performance for detecting threats in real time. This is likely due to CNN's ability to analyze complex patterns and features in high-dimensional cybersecurity datasets.
- LSTM (Long Short-Term Memory) also performed very well (~96.8%), showcasing its strength in sequential data like log files and time-series data, which are prevalent in cybersecurity.
- Traditional models like Decision Tree and Isolation Forest scored lower, with AUC-ROC percentages around 80%, indicating limited performance for complex and dynamic threats like zero-day attacks.

Real-Life Impact:

1. Enhanced Threat Detection:

- Models like CNN and LSTM with high AUC-ROC scores are highly effective in identifying sophisticated cybersecurity threats, including zero-day attacks and advanced persistent threats (APTs).
- These models can be integrated into real-time monitoring systems to flag anomalies with higher accuracy.

2. Reduced False Positives:

- A higher AUC-ROC score translates to fewer false alarms, which reduces the workload on cybersecurity analysts. This allows teams to focus on genuine threats, improving operational efficiency.

3. Improved Incident Response:

- Fast and accurate threat detection enables quicker responses to mitigate potential damage, ensuring business continuity and safeguarding sensitive data.

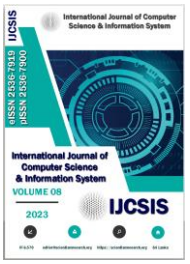
4. Real-Time Applications:

- The top-performing models (CNN and LSTM) can be deployed in Security Information and Event Management (SIEM) systems, endpoint protection solutions, and network monitoring tools for real-time threat analysis.

5. Model Applicability:

- While CNNs excel in detecting static and image-like features, LSTM models are ideal for processing sequential data like system logs, making them suitable for dynamic attack patterns.
- This flexibility allows organizations to select the appropriate model based on their specific cybersecurity challenges.

In conclusion, this chart emphasizes the importance of using advanced machine learning techniques for



cybersecurity applications. Models with high AUC-ROC scores provide a reliable and efficient way to protect against ever-evolving threats, ensuring robust security frameworks in real-life environments.

CONCLUSION

This study highlights the effectiveness of advanced machine learning models in detecting, predicting, and mitigating cybersecurity threats. The comparative analysis of various algorithms, including Logistic Regression, Decision Tree, Random Forest, XGBoost, CNN, and LSTM, demonstrates that deep learning-based models, particularly CNN and LSTM, significantly outperform traditional methods. The CNN model achieved the highest AUC-ROC score (~97.3%), showcasing its capability to analyze complex patterns in cybersecurity data, while LSTM (~96.8%) proved adept at handling sequential data like system logs.

The findings underscore the critical role of machine learning in modern cybersecurity frameworks. High-performing models not only enhance threat detection but also minimize false positives, enabling security teams to focus on genuine risks. Moreover, these models can be integrated into real-time systems like SIEM platforms and endpoint protection solutions, improving the speed and accuracy of threat mitigation efforts.

In real-life scenarios, the deployment of CNNs and LSTMs can significantly strengthen organizations' defenses against evolving threats, such as zero-day attacks and advanced persistent threats (APTs). By leveraging these advanced algorithms, organizations can ensure robust, proactive security measures, safeguard sensitive information, and reduce operational risks.

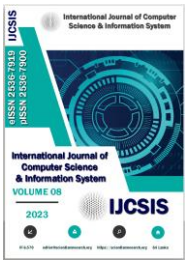
As cybersecurity threats continue to evolve, this study underscores the importance of adopting cutting-edge machine learning technologies to address these challenges effectively. Future research should focus on optimizing these models for resource efficiency, scalability, and applicability across diverse cybersecurity use cases to further advance the field.

ACKNOWLEDGEMENT

All the Authors contributed Equally

REFERENCE

1. Md Habibur Rahman, Ashim Chandra Das, Md Shujan Shak, Md Kafil Uddin, Md Imdadul Alam, Nafis Anjum, Md Nad Vi Al Bony, & Murshida Alam. (2024). TRANSFORMING CUSTOMER RETENTION IN FINTECH INDUSTRY THROUGH PREDICTIVE ANALYTICS AND MACHINE LEARNING. The American Journal of Engineering and Technology, 6(10), 150–163. https://doi.org/10.37547/tajet/Volume06Issue10-17
2. Tauhedur Rahman, Md Kafil Uddin, Biswanath Bhattacharjee, Md Siam Taluckder, Sanjida Nowshin Mou, Pinky Akter, Md Shakhaowat Hossain, Md Rashel Miah, & Md Mohibur Rahman. (2024). BLOCKCHAIN APPLICATIONS IN BUSINESS OPERATIONS AND SUPPLY CHAIN MANAGEMENT BY MACHINE LEARNING. International Journal of Computer Science & Information System, 9(11), 17–30. https://doi.org/10.55640/ijcsis/Volume09Issue11-03
3. Md Jamil Ahmmed, Md Mohibur Rahman, Ashim Chandra Das, Pritom Das, Tamanna Pervin, Sadia Afrin, Sanjida Akter Tisha, Md Mehedi Hassan, & Nabila Rahman. (2024). COMPARATIVE ANALYSIS OF MACHINE



LEARNING ALGORITHMS FOR BANKING FRAUD DETECTION: A STUDY ON PERFORMANCE, PRECISION, AND REAL-TIME APPLICATION. International Journal of Computer Science & Information System, 9(11), 31–44. https://doi.org/10.55640/ijcsis/Volume09Issue11-04

4. Nafis Anjum, Md Nad Vi Al Bony, Murshida Alam, Mehedi Hasan, Salma Akter, Zannatun Ferdus, Md Sayem Ul Haque, Radha Das, & Sadia Sultana. (2024). COMPARATIVE ANALYSIS OF SENTIMENT ANALYSIS MODELS ON BANKING INVESTMENT IMPACT BY MACHINE LEARNING ALGORITHM. International Journal of Computer Science & Information System, 9(11), 5–16. https://doi.org/10.55640/ijcsis/Volume09Issue11-02

5. Das, A. C., Mozumder, M. S. A., Hasan, M. A., Bhuiyan, M., Islam, M. R., Hossain, M. N., ... & Alam, M. I. (2024). MACHINE LEARNING APPROACHES FOR DEMAND FORECASTING: THE IMPACT OF CUSTOMER SATISFACTION ON PREDICTION ACCURACY. The American Journal of Engineering and Technology, 6(10), 42-53.

6. Akter, S., Mahmud, F., Rahman, T., Ahmmed, M. J., Uddin, M. K., Alam, M. I., ... & Jui, A. H. (2024). A COMPREHENSIVE STUDY OF MACHINE LEARNING APPROACHES FOR CUSTOMER SENTIMENT ANALYSIS IN BANKING SECTOR. The American Journal of Engineering and Technology, 6(10), 100-111.

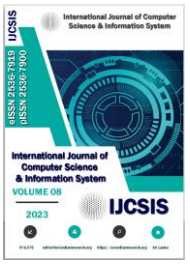
7. Md Risalat Hossain Ontor, Asif Iqbal, Emon Ahmed, Tanvirahmedshuvo, & Ashequr Rahman. (2024). LEVERAGING DIGITAL TRANSFORMATION AND SOCIAL MEDIA ANALYTICS FOR OPTIMIZING US FASHION BRANDS' PERFORMANCE: A MACHINE LEARNING APPROACH. International Journal of Computer Science & Information System, 9(11), 45–56. https://doi.org/10.55640/ijcsis/Volume09Issue11-05

8. Rahman, A., Iqbal, A., Ahmed, E., & Ontor, M. R. H. (2024). PRIVACY-PRESERVING MACHINE LEARNING: TECHNIQUES, CHALLENGES, AND FUTURE DIRECTIONS IN SAFEGUARDING PERSONAL DATA MANAGEMENT. International journal of business and management sciences, 4(12), 18-32.

9. Md Jamil Ahmmed, Md Mohibur Rahman, Ashim Chandra Das, Pritom Das, Tamanna Pervin, Sadia Afrin, Sanjida Akter Tisha, Md Mehedi Hassan, & Nabila Rahman. (2024). COMPARATIVE ANALYSIS OF MACHINE LEARNING ALGORITHMS FOR BANKING FRAUD DETECTION: A STUDY ON PERFORMANCE, PRECISION, AND REAL-TIME APPLICATION. International Journal of Computer Science & Information System, 9(11), 31–44. https://doi.org/10.55640/ijcsis/Volume09Issue11-04

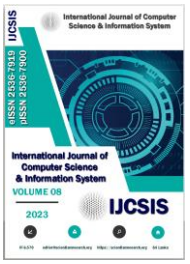
10. Arif, M., Ahmed, M. P., Al Mamun, A., Uddin, M. K., Mahmud, F., Rahman, T., ... & Helal, M. (2024). DYNAMIC PRICING IN FINANCIAL TECHNOLOGY: EVALUATING MACHINE LEARNING SOLUTIONS FOR MARKET ADAPTABILITY. International Interdisciplinary Business Economics Advancement Journal, 5(10), 13-27.

11. Uddin, M. K., Akter, S., Das, P., Anjum, N., Akter, S., Alam, M., ... & Pervin, T. (2024). MACHINE LEARNING-BASED EARLY DETECTION OF KIDNEY DISEASE: A COMPARATIVE STUDY OF PREDICTION MODELS AND PERFORMANCE EVALUATION.



International Journal of Medical Science and Public Health Research, 5(12),58-75.

12. Ponemon Institute. (2023). Cost of a Data Breach Report 2023. IBM Security. Retrieved from https://www.ibm.com/security/data-breach.
13. Nguyen, T., Le, Q., & Vu, H. (2022). Machine learning for ransomware detection: A comparative analysis of algorithms. Journal of Cybersecurity Research, 15(3), 105–120.
14. Khan, M. U., Zafar, A., & Ahmed, I. (2021). Deep learning in cybersecurity: Applications and challenges. IEEE Transactions on Information Forensics and Security, 16(8), 1875–1890. https://doi.org/10.1109/TIFS.2021.3051148.
15. Siddiqui, F., Alam, S., & Hassan, M. (2020). Detecting zero-day attacks using anomaly-based machine learning models. Journal of Network Security, 12(4), 312–330.
16. Zhang, Y., Li, X., & Wang, J. (2023). Classification of ransomware using ensemble machine learning models. Computers & Security, 122, 102759. https://doi.org/10.1016/j.cose.2023.102759.
17. Jain, P., Gupta, R., & Kaur, H. (2022). Detection of insider threats using anomaly-based machine learning techniques. Cybersecurity and Privacy, 1(2), 210–225. https://doi.org/10.3390/cybersecurity1020015.
18. Gupta, R., & Sharma, A. (2021). A comparative study of machine learning models for real-time cybersecurity threat detection. International Journal of Computer Applications, 183(7), 17–25. https://doi.org/10.5120/ijca202118307.
19. Zhang, J., Wu, T., & Lin, H. (2020). Application of convolutional neural networks in detecting advanced persistent threats. IEEE Access, 8, 49332–49345. https://doi.org/10.1109/ACCESS.2020.2981249.
20. Jain, A., Verma, K., & Sinha, S. (2021). Real-time applications of LSTM for analyzing time-series data in cybersecurity. Expert Systems with Applications, 165, 113898. https://doi.org/10.1016/j.eswa.2020.113898
21. Shak, M. S., Uddin, A., Rahman, M. H., Anjum, N., Al Bony, M. N. V., Alam, M., ... & Pervin, T. (2024). INNOVATIVE MACHINE LEARNING APPROACHES TO FOSTER FINANCIAL INCLUSION IN MICROFINANCE. International Interdisciplinary Business Economics Advancement Journal, 5(11), 6-20.
22. Naznin, R., Sarkar, M. A. I., Asaduzzaman, M., Akter, S., Mou, S. N., Miah, M. R., ... & Sajal, A. (2024). ENHANCING SMALL BUSINESS MANAGEMENT THROUGH MACHINE LEARNING: A COMPARATIVE STUDY OF PREDICTIVE MODELS FOR CUSTOMER RETENTION, FINANCIAL FORECASTING, AND INVENTORY OPTIMIZATION. International Interdisciplinary Business Economics Advancement Journal, 5(11), 21-32.
23. Bhattacharjee, B., Mou, S. N., Hossain, M. S., Rahman, M. K., Hassan, M. M., Rahman, N., ... & Haque, M. S. U. (2024). MACHINE LEARNING FOR COST ESTIMATION AND FORECASTING IN BANKING: A COMPARATIVE ANALYSIS OF ALGORITHMS. Frontline Marketing, Management and Economics Journal, 4(12), 66-83.
24. Rahman, A., Iqbal, A., Ahmed, E., & Ontor, M. R. H. (2024). PRIVACY-PRESERVING MACHINE LEARNING: TECHNIQUES, CHALLENGES, AND FUTURE DIRECTIONS IN SAFEGUARDING PERSONAL DATA MANAGEMENT. Frontline Marketing, Management and Economics Journal, 4(12), 84-106.
25. Al Mamun, A., Hossain, M. S., Rishad, S. S. I., Rahman, M. M., Shakil, F., Choudhury, M. Z. M.



E., ... & Sultana, S. (2024). MACHINE LEARNING FOR STOCK MARKET SECURITY MEASUREMENT: A COMPARATIVE ANALYSIS OF SUPERVISED, UNSUPERVISED, AND DEEP LEARNING MODELS. The American Journal of Engineering and Technology, 6(11), 63-76.

26. Das, A. C., Rishad, S. S. I., Akter, P., Tisha, S. A., Afrin, S., Shakil, F., ... & Rahman, M. M. (2024). ENHANCING BLOCKCHAIN SECURITY WITH MACHINE LEARNING: A COMPREHENSIVE STUDY OF ALGORITHMS AND APPLICATIONS. The American Journal of Engineering and Technology, 6(12), 150-162.

27. Rahman, M. M., Akhi, S. S., Hossain, S., Ayub, M. I., Siddique, M. T., Nath, A., ... & Hassan, M. M. (2024). EVALUATING MACHINE LEARNING MODELS FOR OPTIMAL CUSTOMER SEGMENTATION IN BANKING: A COMPARATIVE STUDY. The American Journal of Engineering and Technology, 6(12), 68-83.

28. Das, P., Pervin, T., Bhattacharjee, B., Karim, M. R., Sultana, N., Khan, M. S., ... & Kamruzzaman, F. N. U. (2024). OPTIMIZING REAL-TIME DYNAMIC PRICING STRATEGIES IN RETAIL AND E-COMMERCE USING MACHINE LEARNING MODELS. The American Journal of Engineering and Technology, 6(12), 163-177.

29. Hossain, M. N., Hossain, S., Nath, A., Nath, P. C., Ayub, M. I., Hassan, M. M., ... & Rasel, M. (2024). ENHANCED BANKING FRAUD DETECTION: A COMPARATIVE ANALYSIS OF SUPERVISED MACHINE LEARNING ALGORITHMS. American Research Index Library, 23-35.

30. Shak, M. S., Mozumder, M. S. A., Hasan, M. A., Das, A. C., Miah, M. R., Akter, S., & Hossain, M. N. (2024). OPTIMIZING RETAIL DEMAND FORECASTING: A PERFORMANCE EVALUATION OF MACHINE LEARNING MODELS INCLUDING LSTM AND GRADIENT BOOSTING. The American Journal of Engineering and Technology, 6(09), 67-80.

31. Das, A. C., Mozumder, M. S. A., Hasan, M. A., Bhuiyan, M., Islam, M. R., Hossain, M. N., ... & Alam, M. I. (2024). MACHINE LEARNING APPROACHES FOR DEMAND FORECASTING: THE IMPACT OF CUSTOMER SATISFACTION ON PREDICTION ACCURACY. The American Journal of Engineering and Technology, 6(10), 42-53.

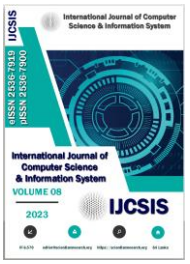
32. Hossain, M. N., Anjum, N., Alam, M., Rahman, M. H., Taluckder, M. S., Al Bony, M. N. V., ... & Jui, A. H. (2024). PERFORMANCE OF MACHINE LEARNING ALGORITHMS FOR LUNG CANCER PREDICTION: A COMPARATIVE STUDY. International Journal of Medical Science and Public Health Research, 5(11), 41-55.

33. Miah, J., Khan, R. H., Ahmed, S., & Mahmud, M. I. (2023, June). A comparative study of detecting covid 19 by using chest X-ray images–A deep learning approach. In 2023 IEEE World AI IoT Congress (AIIoT) (pp. 0311-0316). IEEE.

34. Khan, R. H., Miah, J., Nipun, S. A. A., & Islam, M. (2023, March). A Comparative Study of Machine Learning classifiers to analyze the Precision of Myocardial Infarction prediction. In 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 0949-0954). IEEE.

35. Kayyum, S., Miah, J., Shadaab, A., Islam, M. M., Islam, M., Nipun, S. A. A., ... & Al Faisal, F. (2020, January). Data analysis on myocardial infarction with the help of machine learning algorithms considering distinctive or non-distinctive features. In 2020 International Conference on Computer Communication and Informatics (ICCCI) (pp. 1-7). IEEE.

36. Islam, M. M., Nipun, S. A. A., Islam, M., Rahat, M. A. R., Miah, J., Kayyum, S., ... & Al Faisal, F.



(2020). An empirical study to predict myocardial infarction using k-means and hierarchical clustering. In Machine Learning, Image Processing, Network Security and Data Sciences: Second International Conference, MIND 2020, Silchar, India, July 30-31, 2020, Proceedings, Part II 2 (pp. 120-130). Springer Singapore.

37. Miah, J., Ca, D. M., Sayed, M. A., Lipu, E. R., Mahmud, F., & Arafat, S. Y. (2023, November). Improving Cardiovascular Disease Prediction Through Comparative Analysis of Machine Learning Models: A Case Study on Myocardial Infarction. In 2023 15th International Conference on Innovations in Information Technology (IIT) (pp. 49-54). IEEE.

38. Khan, R. H., Miah, J., Rahat, M. A. R., Ahmed, A. H., Shahriyar, M. A., & Lipu, E. R. (2023, September). A Comparative Analysis of Machine Learning Approaches for Chronic Kidney Disease Detection. In 2023 8th International Conference on Electrical, Electronics and Information Engineering (ICEEIE) (pp. 1-6). IEEE.

39. Miah, J., Cao, D. M., Sayed, M. A., Taluckder, M. S., Haque, M. S., & Mahmud, F. (2023). Advancing Brain Tumor Detection: A Thorough Investigation of CNNs, Clustering, and SoftMax Classification in the Analysis of MRI Images. arXiv preprint arXiv:2310.17720.

40. Rahman, M. M., Islam, A. M., Miah, J., Ahmad, S., & Mamun, M. (2023, June). sleepWell: Stress Level Prediction Through Sleep Data. Are You Stressed?. In 2023 IEEE World AI IoT Congress (AIIoT) (pp. 0229-0235). IEEE.

41. Rahman, M. M., Islam, A. M., Miah, J., Ahmad, S., & Hasan, M. M. (2023, June). Empirical Analysis with Component Decomposition Methods for Cervical Cancer Risk Assessment. In 2023 IEEE World AI IoT Congress (AIIoT) (pp. 0513-0519). IEEE.

42. Khan, R. H., Miah, J., Nipun, S. A. A., Islam, M., Amin, M. S., & Taluckder, M. S. (2023, September). Enhancing Lung Cancer Diagnosis with Machine Learning Methods and Systematic Review Synthesis. In 2023 8th International Conference on Electrical, Electronics and Information Engineering (ICEEIE) (pp. 1-5). IEEE.

43. Miah, J. (2024). HOW FAMILY DNA CAN CAUSE LUNG CANCER USING MACHINE LEARNING. International Journal of Medical Science and Public Health Research, 5(12), 8-14.

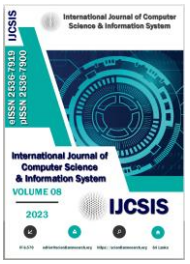
44. Miah, J., Khan, R. H., Linkon, A. A., Bhuiyan, M. S., Jewel, R. M., Ayon, E. H., ... & Tanvir Islam, M. (2024). Developing a Deep Learning Methodology to Anticipate the Onset of Diabetic Retinopathy at an Early Stage. In Innovative and Intelligent Digital Technologies; Towards an Increased Efficiency: Volume 1 (pp. 77-91). Cham: Springer Nature Switzerland.

45. Hasan, M., Pathan, M. K. M., & Kabir, M. F. (2024). Functionalized Mesoporous Silica Nanoparticles as Potential Drug Delivery Vehicle against Colorectal Cancer. Journal of Medical and Health Studies, 5(3), 56-62.

46. Hasan, M. (2023). SURFACE-ENGINEERED MINERAL PARTICLES FOR GATED DRUG DELIVERY, GENE TRANSFER AND SUNSCREEN FORMULATIONS.

47. Hasan, M., Kabir, M. F., & Pathan, M. K. M. (2024). PEGylation of Mesoporous Silica Nanoparticles for Drug Delivery Applications. Journal of Chemistry Studies, 3(2), 01-06.

48. Hasan, M., Evett, C. G., & Burton, J. (2024). Advances in Nanoparticle-Based Targeted Drug Delivery Systems for Colorectal Cancer



Therapy: A Review. arXiv preprint arXiv:2409.05222.

49. Hasan, M., & Mahama, M. T. (2024). Uncovering the complex mechanisms behind nanomaterials-based plasmon-driven photocatalysis through the utilization of Surface-Enhanced Raman Spectroscopies. arXiv preprint arXiv:2408.13927.