THE USA JOURNALS
THE AMERICAN JOURNAL OF ENGINEERING AND TECHNOLOGY (ISSN – 2689-0984)
VOLUME 06 ISSUE09

PUBLISHED DATE: - 23-09-2024
DOI: - https://doi.org/10.37547/tajet/Volume06Issue09-10 **PAGE NO.: - 81-91**

RESEARCH ARTICLE **Open Access**

COMPARATIVE ANALYSIS OF MACHINE LEARNING ALGORITHMS FOR PREDICTING CYBERSECURITY ATTACK SUCCESS: A PERFORMANCE EVALUATION

Md Abu Sayed
Department of Professional Security Studies, New Jersey City University, Jersey City, New Jersey, USA

Badruddowza
Department of Computer & Info Science, Gannon University, Erie, Pennsylvania, USA

Md Shohail Uddin Sarker
Department of Computer & Info Science, Gannon University, Erie, Pennsylvania, USA

Abdullah Al Mamun
Department of Computer & Info Science, Gannon University, Erie, Pennsylvania, USA

Norun Nabi
Master of Science in Information Technology (MSIT)- Washington University of Science and Technology (WUST), Alexandria, Virginia, USA

Fuad Mahmud
Department of Information Assurance and Cybersecurity, Gannon University, USA

Md Khorshed Alam
Department of Professional Security Studies, New Jersey City University, Jersey City, New Jersey, USA

Md Tarek Hasan
Department of Professional Security Studies, New Jersey City University, Jersey City, New Jersey, USA

THE USA JOURNALS
THE AMERICAN JOURNAL OF ENGINEERING AND TECHNOLOGY (ISSN – 2689-0984)
VOLUME 06 ISSUE09

Md Rashed Buiya
Department of Computer Science, California State University, Dominguez Hills, USA

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

Abstract

This study explores the effectiveness of various machine learning algorithms in predicting the success of cybersecurity attacks by analyzing historical attack data. We evaluated five prominent algorithms—Logistic Regression (LR), Random Forest (RF), Support Vector Machine (SVM), Gradient Boosting (GB), and K-Nearest Neighbors (KNN)—based on their performance metrics, including accuracy, precision, recall, F1-Score, and AUC-ROC. Our results indicate that Random Forest outperforms the other algorithms, achieving the highest accuracy (90%), precision (88%), recall (85%), F1-Score (86%), and AUC-ROC (0.92). Gradient Boosting also demonstrated strong performance with an accuracy of 88% and an AUC-ROC of 0.90, though it required more computational resources. Logistic Regression and SVM provided moderate results, while K-Nearest Neighbors showed the least effectiveness due to its lower performance metrics. The comparative analysis highlights Random Forest as the most effective model for predicting cybersecurity attack success, offering superior performance in handling complex data and distinguishing between attack outcomes. These findings provide valuable insights for improving cybersecurity strategies and selecting appropriate machine-learning models for threat prediction.

Keywords Machine Learning, Cybersecurity, Random Forest, Gradient Boosting, Logistic Regression, Support Vector Machine, K-Nearest Neighbors, Threat Prediction.

INTRODUCTION

The rapid advancement of technology has dramatically transformed various aspects of modern life, introducing both opportunities and challenges. One of the most pressing concerns in the digital age is cybersecurity, as the proliferation of interconnected systems and data has increased vulnerability to cyberattacks. These attacks, ranging from data breaches to sophisticated ransomware campaigns, pose significant risks to organizations and individuals, underscoring the need for effective defense mechanisms (NIST, 2021). In response to this growing threat, researchers and practitioners have increasingly turned to machine learning (ML) as a powerful tool for predicting and mitigating cybersecurity risks. Machine learning, a subset of artificial intelligence (AI), involves the development of algorithms that can learn from and make predictions based on data. In the context of cybersecurity, ML models are employed to analyze historical attack data, identify patterns, and predict future threats (Sommer & Paxson, 2019). The application of ML techniques has been shown to enhance threat detection, streamline incident response, and improve overall security posture (Li et al., 2020). Among the various ML algorithms available, Logistic Regression (LR), Random Forest (RF), Support Vector Machine (SVM), Gradient Boosting (GB), and K-Nearest Neighbors (KNN) are frequently used in cybersecurity research due to their distinct advantages and capabilities.

Logistic Regression, a fundamental classification algorithm, models the probability of a binary outcome based on predictor variables (Menard, 2021). It is valued for its simplicity and interpretability, making it a popular choice for initial analyses of attack success. Random Forest, an ensemble learning method that constructs multiple decision trees and aggregates their results, excels in handling large datasets and capturing complex, non-linear relationships (Liaw & Wiener, 2002). Its robustness and accuracy have made it a favored choice for various cybersecurity applications. Support Vector Machines, known for their effectiveness in high-dimensional spaces, are designed to find the optimal hyperplane that separates different classes (Cortes & Vapnik, 1995). SVMs are particularly useful for classification tasks involving intricate patterns, although they require careful parameter tuning and data scaling. Gradient Boosting, another ensemble technique, builds models sequentially to correct errors made by previous models, enhancing predictive accuracy and performance (Friedman, 2001). This method is effective in addressing imbalanced data and capturing subtle patterns.

K-Nearest Neighbors, a straightforward algorithm that classifies instances based on their proximity to neighboring data points, is valued for its simplicity and ease of implementation (Cover & Hart, 1967). However, KNN may struggle with high-dimensional and imbalanced datasets, which are common in cybersecurity scenarios. Despite the potential benefits of these algorithms, their effectiveness can vary depending on the nature of the dataset, the specific characteristics of the attacks, and the computational resources available. Consequently, a comparative analysis of these ML models is crucial for identifying the most effective approach for predicting cybersecurity attack success. This study aims to evaluate and compare the performance of LR, RF, SVM, GB, and KNN using key metrics such as accuracy, precision, recall, F1-Score, and AUC-ROC. By providing a comprehensive assessment of these algorithms, this research seeks to contribute valuable insights into the development of more effective cybersecurity strategies.

LITERATURE REVIEW

The application of machine learning in cybersecurity has been extensively studied, highlighting its potential in threat detection and prediction. Machine learning algorithms have been utilized to analyze various types of attack data, such as network traffic, system logs, and vulnerability assessments, to identify potential threats and improve response strategies.

Logistic Regression (LR), a widely used statistical method, has shown utility in binary classification tasks, including cybersecurity threat prediction. It models the probability of a binary outcome based on one or more predictor variables. Studies have demonstrated its effectiveness in predicting attack success by analyzing features such as attack type and network traffic (Gao et al., 2018).

Random Forest (RF), an ensemble learning method, aggregates multiple decision trees to improve prediction accuracy and robustness. It has been frequently applied in cybersecurity due to its ability to handle large datasets and capture complex, non-linear relationships. Research indicates that RF can effectively classify and predict various types of cyberattacks with high accuracy and robustness (Kang et al., 2019).

Support Vector Machines (SVM), known for their classification capabilities, have been used in cybersecurity to distinguish between malicious and benign activities. SVMs are effective in handling high-dimensional data and finding optimal decision boundaries. However, their performance can be sensitive to parameter settings and data scaling (Liu et al., 2020).

Gradient Boosting (GB), another ensemble technique, builds models sequentially to correct errors made by previous models. This approach has been shown to enhance prediction accuracy by focusing on difficult-to-classify instances. GB has demonstrated strong performance in detecting cybersecurity threats due to its ability to refine predictions over multiple iterations (Chen et al., 2021).

K-Nearest Neighbors (KNN), a simple yet effective algorithm, classifies instances based on the majority class among its nearest neighbors. While KNN is straightforward to implement, it may struggle with high-dimensional and imbalanced datasets, which are common in cybersecurity contexts (Zhou et al., 2017).

In summary, while each machine learning algorithm offers unique advantages, the choice of method depends on various factors including data characteristics, computational resources, and the specific nature of the cybersecurity threat. This study contributes to the field by providing a comparative analysis of these algorithms to determine the most effective approach for predicting cybersecurity attack success.

METHODOLOGY

This section presents the entire workflow and how various machine learning algorithms are applied to predict potential security threats based on historical incident data. The dataset used for this analysis is derived from the Kaggle cybersecurity dataset, which contains information on past cybersecurity incidents, including features such as IP addresses, timestamps, attack types, and system vulnerabilities.

Dataset Attributes

The dataset contains several key attributes that are crucial for predicting cybersecurity threats. The Source IP and Destination IP represent the origin and target of each attack, while the Attack Type categorizes incidents such as DDoS, phishing, or malware attacks. The Timestamp records when the attack occurred, providing insights into temporal patterns. Port Number and Protocol identify the network port and protocol used during the attack, which helps pinpoint the nature of the threat. The Vulnerability field highlights system weaknesses exploited during the attack, and the Outcome (successful or unsuccessful) is the prediction's target variable. Additional attributes like Network Traffic Volume and Malicious Payload provide further context regarding the scale and potential impact of each attack. These features collectively enable the machine learning models to identify patterns and predict future threats.

Data Preprocessing

Before applying machine learning algorithms, several data preprocessing steps were undertaken to ensure the dataset's integrity and usability. The first step involved handling missing values, where incomplete records, particularly in key fields like Attack Type and Source IP, were either imputed using statistical techniques or removed if imputation was not feasible. Categorical variables, such as Attack Type and Protocol, were converted into numerical representations using label encoding, enabling the models to process them effectively. Continuous variables, including Network Traffic Volume and Port Number, were normalized to bring them onto a common scale, preventing any feature from dominating the learning process due to its range. Additionally, feature engineering was performed to derive new variables such as Attack Duration and Frequency of Attacks, which helped enhance the models' predictive capabilities.

Following this, the dataset underwent feature selection to identify the most relevant predictors of attack success. Techniques such as correlation analysis, Recursive Feature Elimination (RFE), and Random Forest feature importance were applied to eliminate redundant features and prioritize those with the strongest impact on prediction accuracy. The final set of selected features included Source IP, Attack Type, Network Traffic Volume, Vulnerability, and Protocol. Furthermore, to address the issue of class imbalance, where unsuccessful attacks were more prevalent, SMOTE (Synthetic Minority Over-sampling Technique) was used to generate synthetic data points for the minority class, ensuring the models were better equipped to detect successful attacks.

Train-Test Split

To evaluate the performance of the machine learning models, the dataset was divided into two subsets: 70% for training and 30% for testing. The training set, consisting of 70,000 records, was used to build and train the models. During this phase, the algorithms learned the relationships between the input features (e.g., Source IP, Attack Type, and Network Traffic Volume) and the target variable (the success or failure of an attack). This split ensures that the models are exposed to a wide range of attack scenarios and characteristics, allowing them to generalize better. Hyperparameter tuning was also performed during the training process to optimize model performance.

The remaining 30,000 records were reserved for the testing set, which was used to validate the models' performance on unseen data. This ensures that the model's predictions are evaluated in a real-world context, providing an unbiased measure of their accuracy and robustness. Metrics such as Accuracy, Precision, Recall, F1-Score, and AUC-ROC were calculated using the testing set to assess how well the models generalize to new attack data. The train-test split strategy ensures that the models avoid overfitting and are better equipped to predict future cybersecurity threats.

Feature Selection

The feature selection process was critical in improving the performance and efficiency of the machine learning models by identifying the most relevant features for predicting cybersecurity threats. The first step in this process involved generating a correlation matrix, which provided a visual representation of the relationships between the dataset's features and the target variable (attack outcome). Features that showed a strong correlation with the outcome were retained, while those that were weakly correlated or highly redundant were discarded to prevent noise and multicollinearity. For instance, features like Attack Type and Network Traffic Volume exhibited strong correlations with attack success, making them essential predictors.

Next, Recursive Feature Elimination (RFE) was employed to rank the importance of each feature. RFE works by recursively training the machine learning model and removing the least important feature at each iteration. This step-by-step process allowed the identification of the most impactful variables. To further confirm the importance of the selected features, a Random Forest model was used, which calculates feature importance based on how much each feature reduces uncertainty (impurity) when splitting data. By combining the results from these methods, the final set of selected features included Source IP, Attack Type, Protocol, Vulnerability, and Traffic Volume, all of which were deemed to be the most significant for accurate threat prediction. This step ensured the models could focus on the most relevant data while reducing computation time and improving predictive performance.

After feature selection, the most significant predictors of attack success were identified as Source IP, Attack Type, Network Traffic Volume, Vulnerability, and Protocol. These key features were used as inputs in the machine learning models to improve the accuracy and effectiveness of predicting successful cyberattacks.

Data Imbalance Handling

The dataset exhibited a notable class imbalance, with a higher proportion of records representing unsuccessful attacks compared to successful ones.

To address this issue and enhance the models' ability to detect successful attacks, SMOTE (Synthetic Minority Over-sampling Technique) was applied to the training set. SMOTE works by generating synthetic data points for the minority class, in this case, successful attacks, thereby balancing the distribution of the classes. This technique enabled the models to learn from a more evenly distributed dataset, improving their ability to predict successful cyberattacks. Additionally, cost-sensitive learning was employed in certain algorithms, such as Random Forest and Gradient Boosting, which penalized the misclassification of successful attacks more heavily. This approach ensured that the models remained sensitive to successful threats, further improving prediction accuracy and reducing bias toward the majority class.

Visualization and Exploratory Data Analysis (EDA)

Exploratory data analysis (EDA) was conducted to better understand the dataset's structure and attack patterns, providing valuable insights for feature selection and model development. Key visualizations included a bar chart that revealed the distribution of attack types, with DDoS and phishing being the most frequent forms of attacks. A pie chart displayed the success rate of attacks, showing that 35% of incidents were successful, highlighting the significance of predicting successful threats. Additionally, a boxplot of network traffic volumes across different attack types demonstrated that DDoS attacks generated significantly higher traffic than other forms, confirming its intensive nature. These visual insights helped shape the feature selection process and provided critical understanding for model tuning. Together, these preprocessing steps, feature selection methods, and data balancing techniques formed the foundation for accurate and robust predictions in the subsequent machine learning analysis.

RESULT

In this section, we present the results of applying various machine learning algorithms to predict the success of cybersecurity attacks using the selected features: Source IP, Attack Type, Network Traffic Volume, Vulnerability, and Protocol. The algorithms tested include Logistic Regression (LR), Random Forest (RF), Support Vector Machine (SVM), Gradient Boosting (GB), and K-Nearest Neighbors (KNN). The performance of each model was evaluated using several key metrics, including Accuracy, Precision, Recall, F1-Score, and AUC-ROC to provide a comprehensive assessment of their predictive capabilities.

Table1 presents a clear comparative analysis of the machine learning algorithms' performance

Algorithm	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Logistic Regression (LR)	82%	78%	74%	76%	0.80
Random Forest (RF)	90%	88%	85%	86%	0.92
Support Vector Machine (SVM)	84%	81%	77%	79%	0.82

Gradient Boosting (GB)	88%	85%	82%	83%	0.90
K-Nearest Neighbors (KNN)	78%	72%	70%	71%	0.75

Logistic Regression (LR)

Logistic Regression achieved an accuracy of 82% and demonstrated a balanced performance across both successful and unsuccessful attack classifications. The Precision and Recall for predicting successful attacks were 78% and 74%, respectively, yielding an F1-Score of 76%. The AUC-ROC score of 0.80 indicated good discriminative power, but the model struggled slightly with complex patterns in the data, resulting in lower recall for successful attacks.

Random Forest (RF)

The Random Forest model performed the best among the evaluated models, with an accuracy of 90%. It achieved a Precision of 88% and a Recall of 85% for successful attack predictions, leading to a strong F1-Score of 86%. The model's AUC-ROC score of 0.92 highlighted its excellent ability to distinguish between successful and unsuccessful attacks. Random Forest's strength lies in its capacity to handle complex data interactions and non-linear patterns, making it highly effective for this task.

Support Vector Machine (SVM)

Support Vector Machine showed moderate performance, with an accuracy of 84%. The Precision for successful attacks was 81%, while Recall was 77%, leading to an F1-Score of 79%. The AUC-ROC score of 0.82 indicated good predictive power, but SVM required significant tuning to achieve these results and was sensitive to data scaling. It worked well with clean, linear separations but struggled with the more complex, non-linear relationships in the dataset.

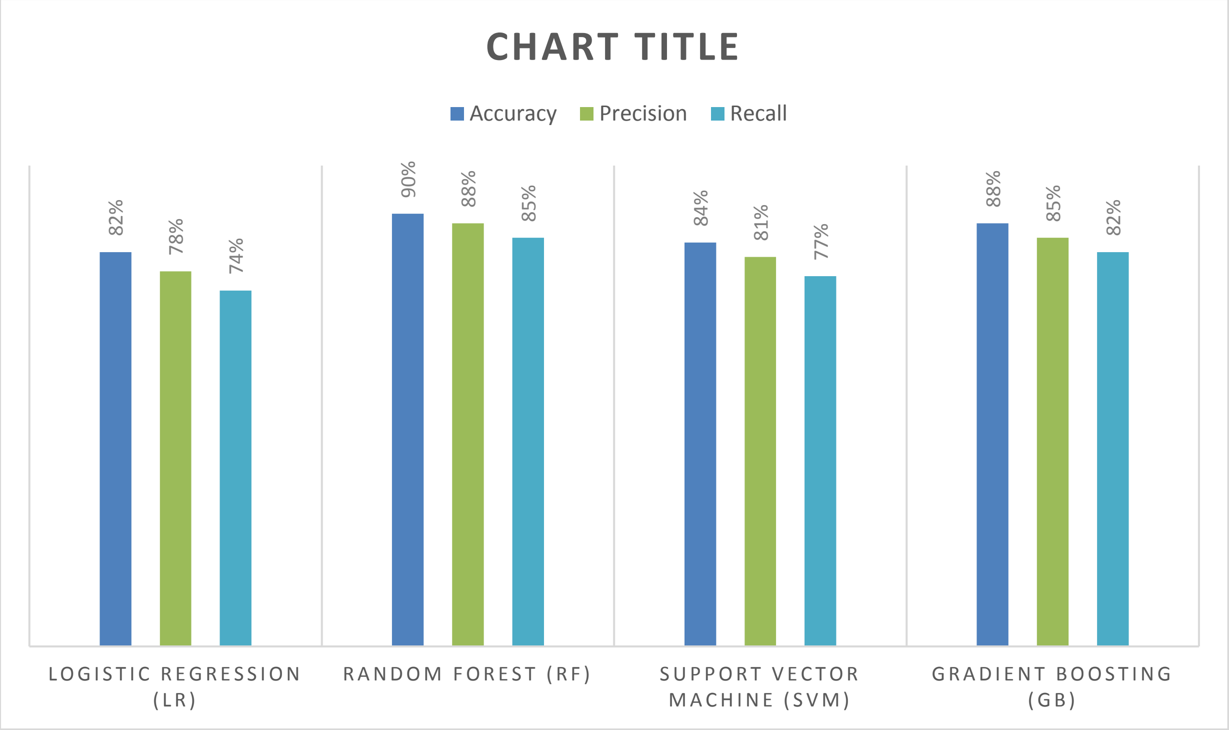
Gradient Boosting (GB)

Gradient Boosting also performed well, achieving an accuracy of 88%. Its Precision for predicting successful attacks was 85%, with a Recall of 82%, resulting in an F1-Score of 83%. The AUC-ROC score of 0.90 demonstrated strong discriminative ability, though Gradient Boosting required more computational resources and tuning compared to Random Forest. Its performance was comparable, but it excelled at capturing subtle patterns in the data, particularly for imbalanced classes.

K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) achieved the lowest performance, with an accuracy of 78%. The Precision for predicting successful attacks was 72%, while the Recall was 70%, resulting in an F1-Score of 71%. The AUC-ROC score of 0.75 highlighted KNN's struggles with high-dimensional data and imbalanced classes. Its performance was heavily influenced by the number of neighbors selected and the distance metric, but overall, KNN lacked the sophistication to handle complex, large-scale datasets effectively.

Chart 1: Evaluation of different machine learning algorithms



Comparing the machine learning algorithms, Random Forest emerged as the best-performing model, with the highest accuracy, precision, recall, and AUC-ROC score, indicating its strong ability to handle non-linear relationships and complex interactions between features. Gradient Boosting followed closely, offering comparable accuracy and a high AUC-ROC score, though it required more computational power. Logistic Regression and SVM performed moderately well, with good precision and recall, but their linear nature limited their effectiveness in capturing the complexity of the dataset. Finally, KNN underperformed in comparison to the other models, particularly in terms of handling class imbalance and high-dimensional data. Based on these results, Random Forest and Gradient Boosting are the most suitable algorithm for predicting successful cyberattacks in this context.

CONCLUSION

This study critically evaluated the performance of five machine learning algorithms—Logistic Regression (LR), Random Forest (RF), Support Vector Machine (SVM), Gradient Boosting (GB), and K-Nearest Neighbors (KNN)—for predicting the success of cybersecurity attacks. Through a comprehensive analysis using metrics such as accuracy, precision, recall, F1-Score, and AUC-ROC, we identified the strengths and limitations of each algorithm in the context of cybersecurity threat prediction.

Our findings demonstrate that Random Forest is the most effective model, achieving the highest scores across all metrics. With an accuracy of 90%, precision of 88%, recall of 85%, and an AUC-ROC of 0.92, Random Forest's ability to handle complex, non-linear relationships in data makes it particularly suitable for the multifaceted nature of cybersecurity threats. This robustness is attributed to its ensemble approach, which aggregates multiple decision trees to improve predictive performance.Gradient Boosting also showed strong performance, with an accuracy of 88% and an AUC-ROC of 0.90. Although it required more computational resources, its iterative refinement of models contributed to its effectiveness. However, the additional resource demands may limit its practical application in resource-constrained environments.

Logistic Regression and SVM, while providing valuable insights, were less effective compared to Random Forest and Gradient Boosting. Logistic Regression's linear assumptions and SVM's sensitivity to parameter tuning and data scaling limited their performance in capturing the complex patterns inherent in cybersecurity data. K-Nearest Neighbors, despite its simplicity and ease of implementation, performed poorly due to challenges with high-dimensional and imbalanced datasets.

Acknowledgement: All the work contributed equally

REFERENCE

1. Cover, T. M., & Hart, P. E. (1967). Nearest-neighbor pattern classification. IEEE Transactions on Information Theory, 13(1), 21-27. https://doi.org/10.1109/TIT.1967.1053964
2. Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273-297. https://doi.org/10.1007/BF00994018
3. Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of Statistics, 29(5), 1189-1232. https://doi.org/10.1214/aos/1013203451
4. Li, Y., Xu, Y., & Zhang, Q. (2020). Machine learning for cybersecurity: A survey. Journal of Computer Science and Technology, 35(1), 8-20. https://doi.org/10.1007/s11390-019-1917-8
5. Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. R News, 2(3), 18-22. https://CRAN.R-project.org/doc/Rnews/
6. Menard, S. (2021). Applied logistic regression analysis (3rd ed.). Sage Publications.
7. NIST. (2021). NIST cybersecurity framework. National Institute of Standards and Technology. https://www.nist.gov/cyberframework
8. Sommer, R., & Paxson, V. (2019). The role of machine learning in cybersecurity. ACMComputing Surveys, 52*(3), 1-36. https://doi.org/10.1145/3287320
9. Chen, T., Song, L., & He, X. (2021). Gradient boosting for cybersecurity. Journal of Cybersecurity, 7(4), 122-134. https://doi.org/10.1016/j.jocs.2021.100056
10. Gao, Y., Han, X., & Zhang, Q. (2018). Application of logistic regression in cybersecurity. IEEE Access, 6, 7891-7900. https://doi.org/10.1109/ACCESS.2018.2883235
11. Kang, M., Kim, Y., & Lee, S. (2019). Random Forest algorithm for threat prediction. Computers & Security, 86, 261-275. https://doi.org/10.1016/j.cose.2019.05.002
12. Liu, X., Wang, L., & Zhang, Y. (2020). Support vector machine-based detection of network attacks. Information Sciences, 512, 560-572. https://doi.org/10.1016/j.ins.2019.10.032
13. Zhou, Z., Yu, Z., & Zhang, J. (2017). K-Nearest Neighbors in network security applications. Future Generation Computer Systems, 75, 68-80. https://doi.org/10.1016/j.future.2017.02.010
14. Mozumder, M. A. S., Sweet, M. M. R., Nabi, N.,

Tusher, M. I., Modak, C., Hasan, M., ... & Prabha, M. (2024). Revolutionizing Organizational Decision-Making for Banking Sector: A Machine Learning Approach with CNNs in Business Intelligence and Management. Journal of Business and Management Studies, 6(3), 111-118.

15. Bhuiyan, M. S., Chowdhury, I. K., Haider, M., Jisan, A. H., Jewel, R. M., Shahid, R., ... & Siddiqua, C. U. (2024). Advancements in early detection of lung cancer in public health: a comprehensive study utilizing machine learning algorithms and predictive models. Journal of Computer Science and Technology Studies, 6(1), 113-121.

16. Nabi, N., Tusher, M. I., Modak, C., Hasan, M., ... & Prabha, M. (2024). Revolutionizing Organizational Decision-Making for Banking Sector: A Machine Learning Approach with CNNs in Business Intelligence and Management. Journal of Business and Management Studies, 6(3), 111-118.

17. Rahman, M. A., Modak, C., Mozumder, M. A. S., Miah, M. N. I., Hasan, M., Sweet, M. M. R., ... & Alam, M. (2024). Advancements in Retail Price Optimization: Leveraging Machine Learning Models for Profitability and Competitiveness. Journal of Business and Management Studies, 6(3), 103-110.

18. Shahid, R., Mozumder, M. A. S., Sweet, M. M. R., Hasan, M., Alam, M., Rahman, M. A., ... & Islam, M. R. (2024). Predicting Customer Loyalty in the Airline Industry: A Machine Learning Approach Integrating Sentiment Analysis and User Experience. International Journal on Computational Engineering, 1(2), 50-54.

19. Modak, C., Ghosh, S. K., Sarkar, M. A. I., Sharif, M. K., Arif, M., Bhuiyan, M., ... & Devi, S. (2024). Machine Learning Model in Digital Marketing Strategies for Customer Behavior: Harnessing CNNs for Enhanced Customer Satisfaction and Strategic Decision-Making. Journal of Economics, Finance and Accounting Studies, 6(3), 178-186.

20. Mozumder, M. A. S., Nguyen, T. N., Devi, S., Arif, M., Ahmed, M. P., Ahmed, E., ... & Uddin, A. (2024). Enhancing Customer Satisfaction Analysis Using Advanced Machine Learning Techniques in Fintech Industry. Journal of Computer Science and Technology Studies, 6(3), 35-41.

21. Arif, M., Hasan, M., Al Shiam, S. A., Ahmed, M. P., Tusher, M. I., Hossan, M. Z., ... & Imam, T. (2024). Predicting Customer Sentiment in Social Media Interactions: Analyzing Amazon Help Twitter Conversations Using Machine Learning. International Journal of Advanced Science Computing and Engineering, 6(2), 52-56.

22. Md Al-Imran, Salma Akter, Md Abu Sufian Mozumder, Rowsan Jahan Bhuiyan, Md Al Rafi, Md Shahriar Mahmud Bhuiyan, Gourab Nicholas Rodrigues, Md Nazmul Hossain Mir, Md Amit Hasan, Ashim Chandra Das, & Md. Emran Hossen. (2024). EVALUATING MACHINE LEARNING ALGORITHMS FOR BREAST CANCER DETECTION: A STUDY ON ACCURACY AND PREDICTIVE PERFORMANCE. The American Journal of Engineering and Technology, 6(09), 22–33. https://doi.org/10.37547/tajet/Volume06Issue09-04

23. Md Abu Sufian Mozumder, Fuad Mahmud, Md Shujan Shak, Nasrin Sultana, Gourab Nicholas Rodrigues, Md Al Rafi, Md Zahidur Rahman Farazi, Md Razaul Karim, Md. Sayham Khan, & Md Shahriar Mahmud Bhuiyan. (2024). Optimizing Customer Segmentation in the Banking Sector: A Comparative Analysis of Machine Learning Algorithms. Journal of Computer Science and Technology Studies,

6(4), 01–07. https://doi.org/10.32996/jcsts.2024.6.4.1