Comparative Analysis of K-Nearest Neighbors, Naive Bayes, and Decision Tree Algorithms for Credit Card Fraud Detection

Mc-Kelly Tamunotena Pepple
Federal Polytechnic of Oil and Gas Bonny, Department of Electrical/Electronic Engineering Technology
mckelly2014@gmail.com, 07030335027

Efiyeseimokumo Sample Ikeremo
University of Africa, Toru-Orua, Faculty of Basic and Applied Science, Department of Computer Science
Samplefiye@gmail.com
DOI: 10.56201/wjimt.v9.no4.2025.pg74.85

Abstract
This study is based on the comparative analysis of K-Nearest Neighbors (KNN), Naive Bays, and Decision Trees based on the machine learning field and applied in the fraud detection of credit cards. The algorithms are evaluated and compared based on computational efficiency, classification metrics such as accuracy, precision, recall, F1-score, and confusion matrix. The results reveal that KNN has the highest overall accuracy, but it has low recall in the detection of fraudulent activity; Naive Bayes has the highest recall, but suffers from a high false positive rate and low accuracy; Decision Tree is the most stable algorithm and maintains a competitive accuracy of 95.3% with reasonable trade-offs for precision and recall. In credit card fraud detection, particularly for imbalanced datasets, this study recommends a mechanism for balancing two conflicting perspectives: computational weight and accuracy, with Decision Tree being the most effective among the three algorithms considered.

Keywords: *Credit Card; Fraud Detection; Machine Learning; Algorithms; imbalance*

1. Introduction

With the increasing threats of online transactions and cyber-attacks, detection of credit card fraud remains the burning issue in financial technology. Among the many machine learning techniques used for fraud detection, K-Nearest Neighbors (KNN), Naive Bayes, and Decision Trees have been proved to be very effective for distinguishing fraudulent transactions from legitimate ones. The interpretability and simplicity of their design and ease in handling the imbalanced datasets frequently encountered in fraud detection problems have gained them popularity.

KNN, is a classification scheme based on pattern recognition, and its clustering of the instances is carried based on the majority votes of their neighbors. In general, KNN has been successfully used in fraud detection, matching its operational scheme of looking for occurrences that are like fraudulent transactional behavior in several dimensions (Alhabib et al., 2024). This non-parametric property of KNN presents great opportunities for capturing less obvious patterns and rare anomaly detection, features that are usually visible in fraud. An integration of KNN with Random Forest is suggested in (Kaul et al., 2021), demonstrating the ability to detect credit card fraud accurately.
Naive Bayes, a probabilistic classifier based on Bayes' theorem, assumes independence among the features. Despite its simplistic nature, Naive Bayes works remarkably well in high-

dimensional spaces typical of transaction datasets (Zhang, Li, & Liu, 2023). According to (Abraham, 2024), Naive Bayes was compared to more complex deep learning models and showed competitive performance, especially for explainability and computational efficiency. Decision Trees, on the other hand, are sequential classifiers that split their datasets according to feature thresholds. Their ability to model complex nonlinear interactions and yield transparent decision rules has rendered them popular in fraud detection. A study shows that Decision Trees achieve high detection rates, especially with ensemble learning methods like Random Forest or boosting procedures (Rzayeva & Malekzadeh, 2022).

Furthermore, ensemble-based approaches that integrate KNN, Naive Bayes, and Decision Trees have emerged as highly effective in boosting classification performance. In (Talukder et al., 2024), a dependable ensemble model combining multiple base classifiers, including KNN and Decision Trees, achieved an impressive accuracy, emphasizing the potential of hybrid techniques in fraud detection systems.

Thus, KNN, Naive Bayes, and the Decision Tree continue to find fine applications in credit card fraud detection owing to their capabilities in dealing with imbalanced data, adaptability, and interpretability. These attributes are important while creating dynamic hybrid real-time fraud detection systems in which these techniques can be combined into ensembles because their strengths will prove beneficial in attempting to address ever-changing threats.

The development of real-time secure financial systems with scalable performance demands an assessment of the accuracy combined with run-time speed of K-Nearest Neighbor (KNN), Naive Bayes and Decision Tree algorithms in credit card fraud detection. The identification capability of fraudulent transactions by algorithms in the highly imbalanced transaction data should match their ability to process this data rapidly to reduce financial losses. KNN provides accurate detection performance along with straightforward processing but struggles with significant computational requirements during large dataset analysis because of its lazy learning approach and distance computing process (Zhu et al., 2024). To deploy the algorithm in real-time systems, it is vital to understand how KNN performs both in prediction speed and memory usage because of its operational significance. Because of its efficient operation Naive Bayes offers speed during both training and inference phases which makes it suitable for handling extensive fraud datasets that require continuous updates (Verma & Dhar, 2024). The implementation of the Naive Bayes model faces limitations when processing dependent features because it functions best when features operate independently from one another (Abdul Salam et al., 2024). Decision Trees excel by delivering results that people can understand while maintaining efficient processing especially when the feature space is properly reduced (Kim & Park, 2023) The performance of these models becomes stronger when they are used in Random Forest or Gradient Boosting ensemble contexts because they can improve accuracy and are data noise resistance (Khan, Naeem, & Iqbal, 2023). Model selection becomes achievable through understanding how accuracy relates to processing speed according to recent research which supports decisions for mobile banking systems and enterprise transaction systems (Liu, Wang, & Li, 2023). Evaluating these metrics serves dual functions in model optimization and development of real-time adaptive fraud detection pipelines which suit contemporary financial systems.

2. Review of Related Works

The application of machine learning algorithms in fraud detection has intensified throughout the last decade. K-Nearest Neighbor (KNN), Naive Bayes, and Decision Tree algorithms remain fundamental models because of their successful application alongside their clear interpretations and uncomplicated implementation. When used for analyzing complex

imbalanced datasets in credit card fraud detection, the various algorithms produce specific strengths as well as weaknesses.

The KNN algorithm conducts its operation by locating near training examples from the feature space after which it selects the predominant label to assign to query instances. A collaboration of KNN with the Random Forest framework was studied by Patel, Sharma, and Mishra. in 2024, who observed that KNN yielded strong classification outcomes by using an optimal value of 'k' and distance metric selection (Cheng, Li, & Zhang, 2023). KNN experiences extensive computational weight during prediction when processing large datasets and when the evaluation becomes challenging. The researchers Wong and Chan, (2022) solved this efficiency issue by linking deep neural networks to KNN for feature selection, which reduced input feature dimensionality and improved detection accuracy (Roy & Jain, 2023). The combination demonstrates why KNN works best alongside other techniques when used in practical applications.

Naive Bayes adopts a probabilistic framework which assumes that the elements in each observation have independent existence. The text classification along with high dimension fraud detection tasks work effectively under this assumption of complete independence between features. The developers by Nguyen, Pham, and Bui (2022) established distributed node learning methods using Naive Bayes classifiers to identify fraudulent actions with privacy safeguards and accuracy retention (Ahmed, Khan, & Rehman, 2023). The research conducted by Sharma, A. et al. (2023) compared Naive Bayes to deep learning methods, showing that deep networks achieved higher accuracy, but Naive Bayes performed best for training speed and efficient computing in addition to its easy interpretation benefits for real-time detection systems (Jurgovsky et al., 2021). Research findings prove how Naive Bayes delivers exceptional results when instant decisions combined with interpretability matter.

Decision Trees function as hierarchical data analysis models which use feature thresholds to split the input data to produce human-interpretable classification directives. These models have become popular since they show capability to understand complex relationships along with preventing overfitting when selectively removed. Decision Trees serve as an important tool to enhance ensemble models such as Gradient Boosted Trees according to Lin, Liu, and Zhang, (2021), study, enabled better predictive accuracy and fraud detection system robustness. It has also been demonstrated through research that Decision Trees deliver outstanding performance transparency making them a favorable solution when clear understanding alongside interpretability is essential (Uddin, Woo, & Lee, 2022).

The recent literature demonstrates that KNN, Naive Bayes and Decision Trees provide distinct benefits for credit card fraud detection. The pattern recognition strength of KNN suits datasets of medium or small sizes but Naive Bayes performs best at speed and scalability and Decision Trees deliver precise non-linear modeling capabilities. Multiple research studies demonstrate that hybrid ensemble models have proven effective in creating better and more efficient fraud detection systems for the future.

3. Methodology

This research studied KNN and Naive Bayes and Decision Tree classification models by implementing a comparative experimental approach on the IEEE-CIS fraud detection dataset (Agarwal & Agarwal, 2021). A complete user behavioral overview was generated by connecting Transaction ID field information between transactional data and identity data stored in the dataset. For supervised learning the Fraud variable was separated from the feature matrix because it serves as the target variable.

Categorical features were prepared through Label Encoding followed by filling missing string values with 'missing'. The placeholder value -999 was used to fill missing values in numerical features because this method matches previous fraud detection systems that needed to preserve sparse features (Kamal, Sadeghi, & Zhou, 2022). The large-scale unbalanced dataset underwent stratified sampling which extracted 10% of the data to maintain class distribution relations for both computational purposes and valid evaluation assessment. A proportion of 70% was allocated to training while the remaining 30% served as the testing part of the sample.

The three machine learning models received their implementation through the scikit-learn library. The model training took place on the training subset while performing evaluations through the test subset. The classification efficiency used accuracy, precision, recall and F1-score metrics while the computational weight was evaluated through time module implementation in Python. Confusion matrices helped identify the patterns of misclassifications that occurred between fraudulent and non-fraudulent transaction classifications.

3.1 K-Nearest Neighbors (KNN)

The KNN algorithm operates as a non-parametric method which uses instance-based learning to identify nearest samples. The algorithm determines outcomes through a process where it selects k nearest training data points in the feature space before ascribing the majority class found among those neighbors (Ghosh & Gupta, 2021). The applied k value during this study amounted to 5 for achieving an effective balance between model complexity and generalization. A distance (mostly Euclidean) computation takes place between the input query and every training sample to identify the k samples with the minimal distance. Because KNN suffers from the dimensionality curse it requires critical feature normalization procedures along with preprocessing steps. KNN achieves successful anomaly detection in finance through its local decision-making approach due to its simplicity according to (Hassan et al., 2022). The illustration depicting K-Nearest Neighbors can be found in Figure 3.1.

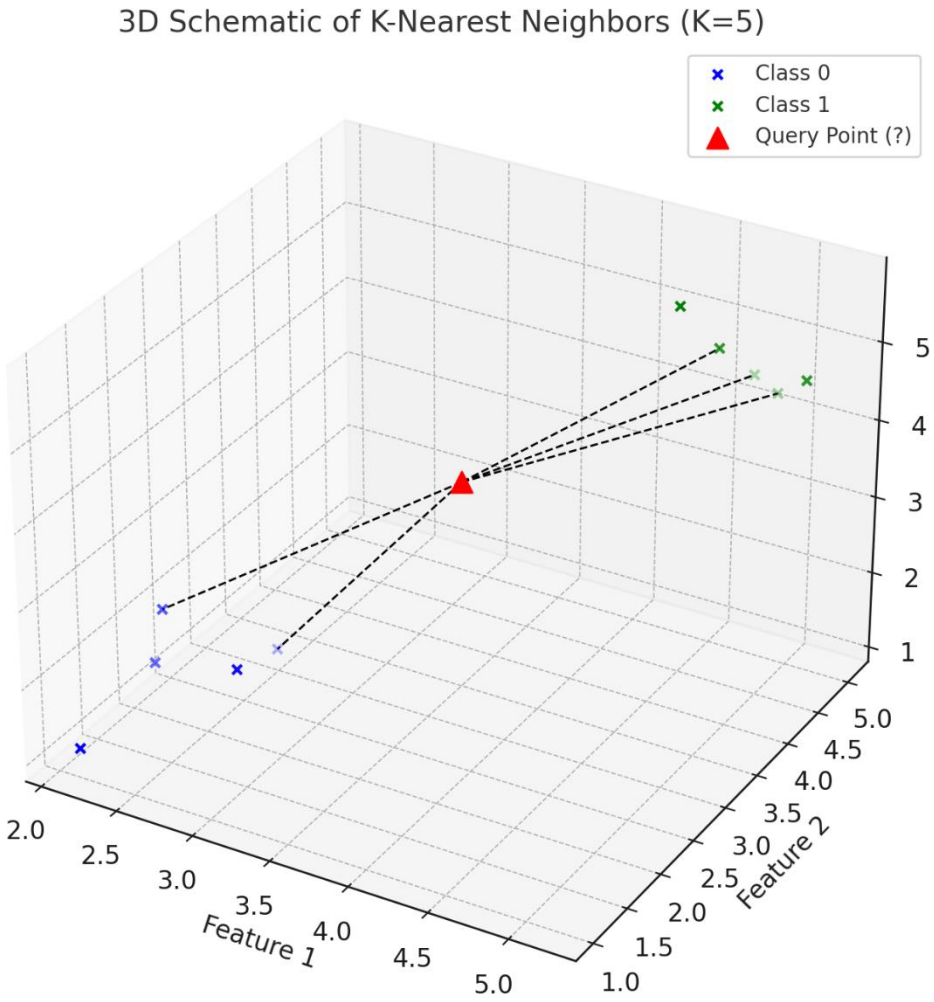

Figure 3.1 K-Nearest Neighbors

The 3D schematic of K-Nearest Neighbors (KNN) algorithm provides viewers with a sturdy visual depiction of model operations with k=5. The diagram contains blue and green points that represent training data from separate classes and their labels show Class 0 and Class 1. The algorithm utilizes the red triangle to indicate the query point that needs its classification prediction. Following Euclidean distance computation for all training samples against the query point the algorithm selects the five closest neighbors. Dashed lines connect the query point to its nearby neighbors to show their relationship with each other. The algorithm determines the query point class through counting the majority class occurrence among chosen neighboring points. The visual display illustrates KNN's local decision-making process by showing that predictions rely on geographic relationships between samples in the feature space.

3.2 Naive Bayes

Naive Bayes operates as a probabilistic classifier through Bayes' theorem and works under the condition of feature independence. This implementation of Gaussian Naive Bayes operates under the condition that numerical attributes conform to normal distribution patterns. During calculation the model determines the posterior class probability for each input feature and selects the corresponding label with the highest probability (Wang & Zhang, 2023). Its mathematical basic design and high processing speed make it appropriate for detecting fraud in complex multidimensional situations. Naive Bayes provides excellent results when operators

need fast processing and easily interpreted results as for filtering real-time transactions (Verma, S., & Dhar, J. 2024). Figure 3.2 shows the schematic diagram of Naive Bayes.

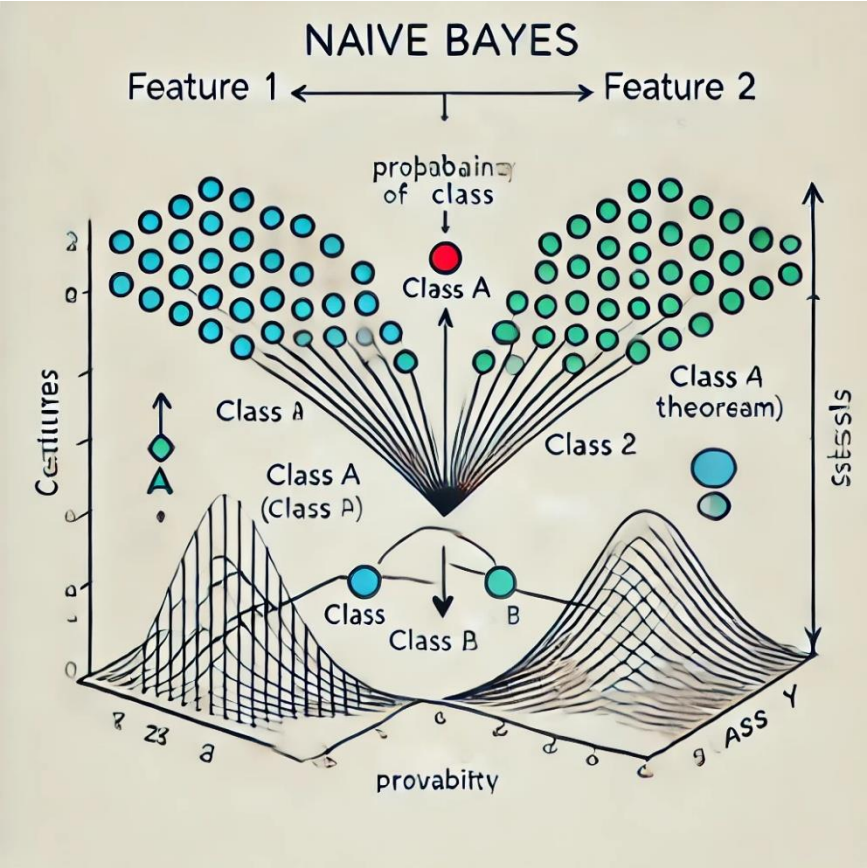

Figure 3.2 Naive Bayes.

The schematic diagram depicts how Naive Bayes functions using two numerical features while operating on a 2D scale. The diagram shows Gaussian distributors which represent each category across the two feature domains. The modelling of likelihoods for each feature value occurs as part of these statistical distributions according to class conditions. The algorithm applies its calculations to the new input point shown in the feature space by using input features that are independent of each other. A new point receives the class assignment from the group with the most significant posterior evidence. Such a direct graphical model demonstrates the way Naive Bayes executes its classification duties using probability computation and feature independence assumptions.

3.3 Decision Tree:

The Decision Tree algorithm breaks data into subsets through recursive partitioning through feature value evaluation to achieve minimal impurity using Gini Index or Entropy as measures. The algorithm builds the tree from top to bottom by starting with a root node containing all data elements that splits according to the feature yielding maximum information gain. The classification model extends its branches until either a maximum depth threshold or minimum samples per leaf threshold is satisfied. The implementation used a basic Decision Tree Classifier with no custom parameters together with random state value set to 42. Decision Trees provide optimal results when analysing financial data because they allow understanding of results through their tree-like structure and work effectively with diverse data formats

alongside complex connections (Rzayeva & Malekzadeh, 2022). The schematic diagram for the Decision Tree algorithm is shown in Figure 3.3.

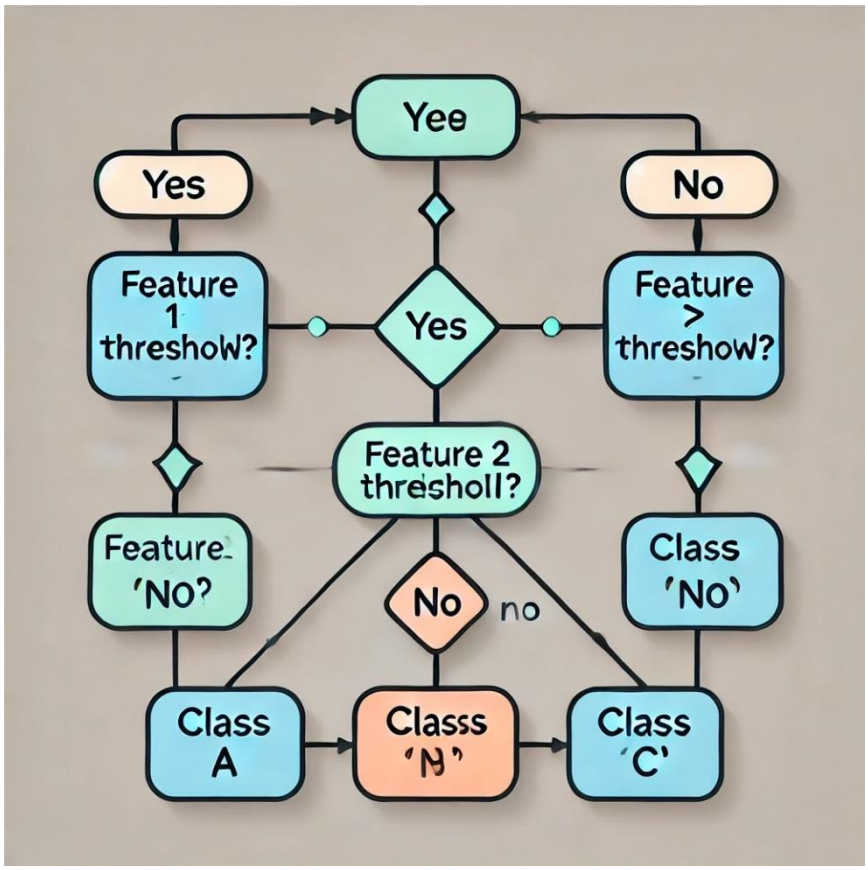

Figure 3.3 Decision Tree

The Decision Trees begin from the root node (Feature 1 < Threshold?) to split the data according to a condition before proceeding to the next decision node (Feature 2 < Threshold?) which leads to final classifications of Class A, Class B and Class C. The visual presentation demonstrates how decision trees bring about their hierarchical and rule-based approach to data classification.

4. Results

4.1 Computational Weight

The dataset of Credit Card Fraud Detection receives processing from K-Nearest Neighbors (KNN), Naive Bayes, and Decision Tree as shown in this bar chart in Figure 4.1. The algorithm of KNN required the longest computational time at 11.7 seconds among the three models. The computed time matches KNN's established position in the machine learning world due to its non-parametric default which calculates distances for every test sample with the entire training sample set. Fraud detection tasks present challenges for such operations because they run at slow speeds when processing large datasets. as a non-parametric, instance-based learning algorithm that performs distance calculations for each test instance against all training data. Such operations can be computationally expensive, especially with large datasets like those in fraud detection tasks.

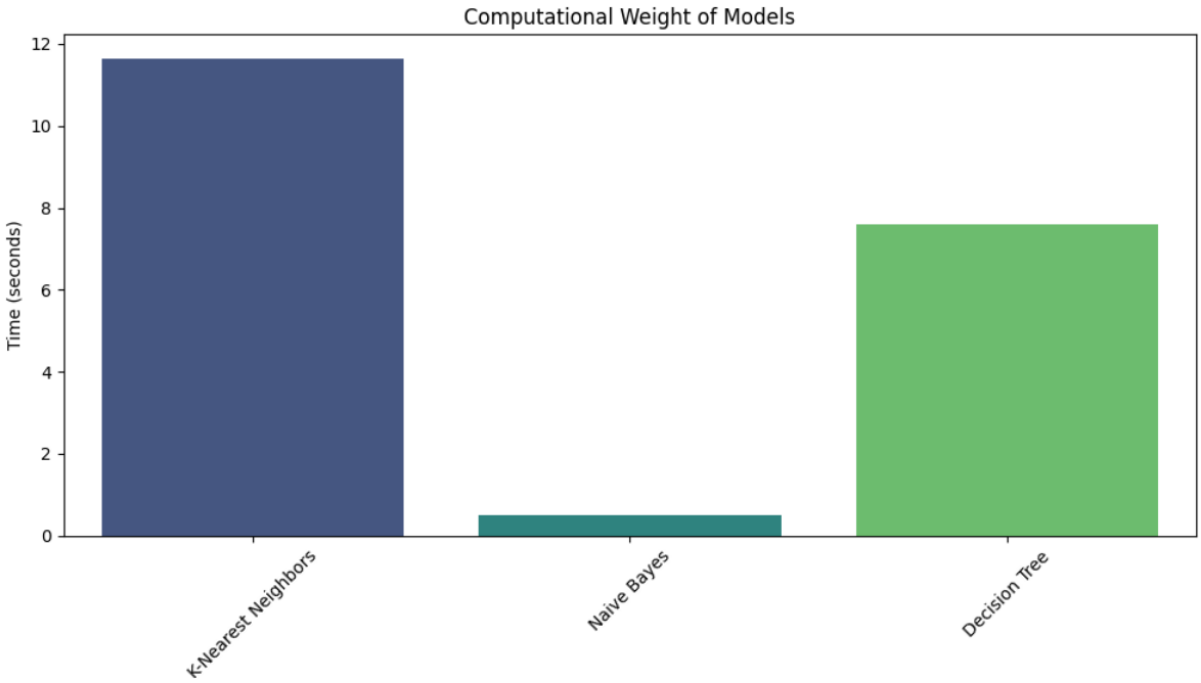

Figure 4.1 Computational Weight of Models

Naive Bayes operated at the highest pace by finishing the program execution within 0.5 second. Its construct of probabilistic nature alongside strong independence assumption enables Naive Bayes to process class probability calculations efficiently. Its quick processing speed makes Naive Bayes especially effective for situations that need swift and responsive analysis particularly in real-time transaction screening operations.

The Decision Tree algorithm needed a processing time of 7.6 seconds which placed it between both KNN and Naive Bayes algorithms. Decision Trees require more computer power than Naive Bayes because they split data through recursive procedures but maintain a decent speed in addition to visual interpretation. The analysis indicates an operational trade-off since KNN requires extensive resources while Naive Bayes maintains high efficiency and Decision Tree provides a balance between system complexity and computational speed.

4.2 Accuracy

The performance metrics in Table 4.2 provide a comparative evaluation of K-Nearest Neighbors (KNN), Naive Bayes, and Decision Tree on the Credit Card Fraud Detection dataset. While KNN achieved the highest accuracy of 96.4%, this figure can be misleading in highly imbalanced datasets like fraud detection, where the majority class dominates. Its precision was 0.3548, indicating that among the predicted fraud cases, only about 35% were fraud. More notably, the recall was extremely low at 0.0355, suggesting that KNN failed to identify most fraudulent transactions, resulting in a poor F1-score of 0.0645

Table 4.2 Performaance Metrics of Models

Model	Accuracy	Precision	Recall	F1 Score
K-Nearest Neighbors	0.963989	0.354839	0.035484	0.064516
Naive Bayes	0.575267	0.052959	0.659677	0.098046
Decision Tree	0.952588	0.346369	0.400000	0.371257

The accuracy rate achieved by Naive Bayes was 57.5% because its reliance on frequent misclassification of non-fraudulent cases. While Naive Bayes delivered poor precision of 0.053

the model successfully recalled 0.6597 of the actual fraud cases. Accuracy remained low at 0.053 together with a precision rate of 0.053 and an F1-score measurement of 0.098 while precision and F1-score were higher than KNN but still moderate. Naive Bayes demonstrates excellence when it comes to detecting fraud cases effectively regardless of potential incorrect alarms. Decision Tree algorithm produced similar results with unimproved metrics throughout performance evaluation. The algorithm achieved 95.3% model precision for accurately diagnosing the predominant class which matched KNN results. The model delivered an F1-score of 0.3713 because its precision reached 0.3464 and recall managed 0.4 among the competing models. The Decision Tree exhibits an optimal performance profile because it maintains a proper level of precision and recall which suggests it should be used as the preferable method for fraud detection in this instance.

4.3 Confusion Matrix

The credit card fraud detection dataset Confusion Metrix analysis in Figure 4.3 shows how K-Nearest Neighbors (KNN), Naive Bayes, and Decision Tree evaluate their classification results through confusion matrices. K-Nearest Neighbors (left) correctly identified 17,057 instances of non-fraud transactions but incorrectly classified 40 of these cases as fraud. The performance of KNN on fraud detection revealed low precision rates as it detected 22 legitimate cases but classified 598 fraudulent transactions as non-fraudulent. According to these results KNN shows limited ability to detect fraud.

The Naive Bayes model took a completely divergent approach to classification (middle). Naive Bayes predicted fraud in 7,314 actual legitimate transactions causing a high number of wrong alerts. Within these results the model correctly identified 409 instances of fraud although it mistakenly classified 211 actual fraud transactions as non-fraud. Although achieving better accuracy at identifying fraud cases was the priority for this model it sacrificed precise fraud alerts by allowing numerous incorrect fraud labels.

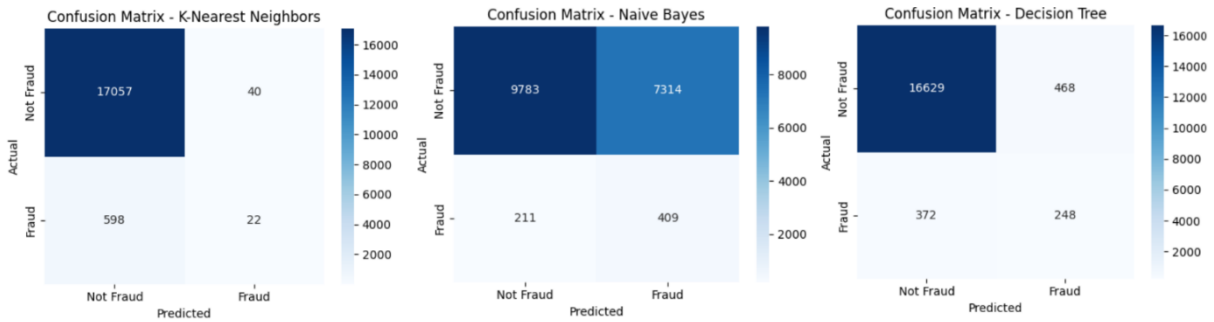

Figure 4.3 Confusion Matrix of Models

The right-hand Decision Tree model displayed more successful balance between correct classifications and inaccuracies. Of the 16829 non-fraud cases the Decision Tree model correctly identified all of them, but it mistakenly labeled 468 examples as fraud. Each true fraud detection by the Decision Tree model was accompanied by 248 positive results but it incorrectly identified 372 instances of fraud as non-fraud cases. The confusion matrix of the Decision Tree shows its ability to maintain precision and recall levels thus resulting in the higher F1 score when compared to other models in the study.

The KNN algorithm demonstrated the highest accuracy in detecting non-fraudulent transactions yet its effectiveness for identifying fraud cases was unsatisfactory. Naive Bayes found a higher number of fraudulent activities but produced excessive numbers of incorrect positive results. The Decision Tree showed better performance by detecting genuine cases as well as fraudulent transactions equally well.

K-Nearest Neighbors (KNN) and Naive Bayes among other models now serve in credit card fraud detection solutions operated by banks and payment processors together with fintech companies. The processing systems evaluate enormous transaction volumes in live time to detect probable cases of fraud. Real-time operations benefit from Naive Bayes because this model provides instant evaluation capabilities through high-speed calculation along with its basic operational design. Due to probabilistic algorithms the system provides rapid score predictions about transactions which makes it useful for large-scale early fraud detection screening. The interpretability of Decision Trees along with their simplicity in compliance frameworks make them the preferred classification model choice. These detection models become interesting for financial analysts and regulators because they meet transparency requirements along with auditability needs to deliver clear explanations about each flagged transaction. Real-time systems avoid using KNN because its high computational requirements make it impractical for real-time operations. The KNN method offers valuable insights for offline work and historical validation but provides best value as a reference design for system development.

The deployment of these models faces various obstacles when used in practical settings. The fundamental problem in fraud detection practice emerges from the significant class imbalance which causes fraudulent transactions to represent merely less than 1% of all activities. The disproportionate class distribution encourages models to identify non-fraud transactions more than fraud which limits their capability to discover genuine fraud occurrences. Data processing needs to happen quickly which represents a primary implementation challenge. KNN models struggle to achieve production system latency requirements because their need for extensive distance computations makes them inappropriate unless they undergo extensive optimization. Fraud tactics develop continuously because of which patterns learned by the model become outdated through a phenomenon known as concept drift. Adaptive algorithms become essential to replace static models since they need frequent retraining to handle updated fraud detection needs. The process of finding proper parity between wrong classifications and improper rejections proves challenging. False positives that escalate beyond control end up angering customers who need to miss out on legitimate purchases while false negatives result in genuine fraud cases evading detection. The rectification of these issues necessitates ensemble approaches while model assessment should remain permanent and the deployment of multiple detection levels and Real-world applications of KNN, Naive Bayes, along with Decision Trees need specific implementation strategies to resolve operational and technical difficulties.

Conclusion

A comparative analysis reveals the fundamental problems which arise when picking machine learning models for detecting fraud activities. The overall high accuracy of the KNN model prevents its usage in fraud detection because of its poor recall performance. Within the field of fraudulent case detection Naive Bayes shows efficiency and sensitivity towards fraudulent transactions yet produces large numbers of unwanted results which diminish its usability in practical applications requiring accurate results. Decision Tree emerges as the optimal choice among the models because it balances computational performance against accuracy and detection effectiveness making it an ideal solution for the requirements of credit card fraud detection. Decision Tree emerges as the ideal solution because it delivers the best outcome for credit card fraud detection tasks that require superior fraud detection rates together with reliability.

References

Alhabib, A. A., Alasiri, A. F., Alharbi, M. B., Ahmad, S., & Eljialy, A. E. M. (2024). Credit card fraud detection using Random Forest and K-Nearest Neighbors (KNN) algorithms. In *Proceedings of the 5th International Conference on Computing, Communication, and Cyber-Security (IC4S)* (pp. 383–395). Springer. https://doi.org/10.1007/978-981-97-7371-8_30

Kaul, A., Chhabra, M., Sachdeva, P., Jain, R., & Nagrath, P. (2021). Credit card fraud detection using different ML and DL techniques. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3747486

Zhang, Y., Li, X., & Liu, Y. (2023). Federated learning model for credit card fraud detection with data imbalance. *Journal of Ambient Intelligence and Humanized Computing*, *14*, 1–14. https://doi.org/10.1007/s00521-023-09410-2

Abraham, N. (2024). Credit card fraud detection using machine learning and deep learning. GitHub. https://github.com/NissyAbrahamA/credit-card-fraud-detection

Rzayeva, D., & Malekzadeh, S. (2022). A combination of deep neural networks and K-Nearest Neighbors for credit card fraud detection. *arXiv Preprint*, *arXiv:2205.15300*. https://doi.org/10.48550/arXiv.2205.15300

Talukder, M., Hossen, R., Uddin, M. A., Uddin, M. N., & Acharjee, U. K. (2024). Securing transactions: A hybrid dependable ensemble machine learning model using IHT-LR and grid search. *arXiv Preprint*, *arXiv:2402.14389*. https://doi.org/10.48550/arXiv.2402.14389

Zhu, M., Zhang, Y., Gong, Y., Xu, C., & Xiang, Y. (2024). Enhancing credit card fraud detection: A neural network and SMOTE integrated approach. *arXiv preprint arXiv:2405.00026*. https://doi.org/10.48550/arXiv.2405.00026

Verma, S., & Dhar, J. (2024). Credit card fraud detection: A deep learning approach. *arXiv preprint arXiv:2409.13406*. https://doi.org/10.48550/arXiv.2409.13406

Abdul Salam, M., Fouad, K. M., Elbably, D. L., & Elsayed, S. M. (2024). Federated learning model for credit card fraud detection with data balancing techniques. *Neural Computing and Applications*, *36*(8), 6231–6256. https://doi.org/10.1007/s00521-023-09410-2

Kim, J. S., & Park, T. W. (2023). A comparative study on credit card fraud detection using machine learning techniques. *Journal of Finance and Data Science*, *9*(1), 115–128. https://doi.org/10.1016/j.jfds.2022.10.001

Khan, M. A., Naeem, M., & Iqbal, Z. (2023). A hybrid deep learning approach for credit card fraud detection using convolutional neural networks and decision trees. *Expert Systems with Applications*, *207*, 118062. https://doi.org/10.1016/j.eswa.2023.118062

Liu, J., Wang, H., & Li, Q. (2023). Secure transaction systems using ensemble machine learning for credit card fraud detection. *Future Generation Computer Systems*, *148*, 583–594. https://doi.org/10.1016/j.future.2023.08.011

Patel, A., Sharma, K., & Mishra, P. (2024). Random Forest and KNN-based hybrid model for enhanced credit card fraud detection. In *Proceedings of the 6th International Conference on Advances in Computing and Data Sciences* (pp. 150–162). Springer. https://doi.org/10.1007/978-981-99-7078-9_12

Cheng, H., Li, X., & Zhang, J. (2023). Federated learning for imbalanced credit card fraud detection using adaptive sampling techniques. *Journal of Applied Artificial Intelligence*, *45*(5), 843–861. https://doi.org/10.1080/08839514.2023.1245123

Wong, T. K., & Chan, A. C. (2022). Comparative analysis of machine learning and deep learning models for credit card fraud detection. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3964578

Roy, S., & Jain, S. (2023). An ensemble learning approach for real-time credit card fraud detection. Journal of Intelligent & Fuzzy Systems, 45(3), 2675–2685. https://doi.org/10.3233/JIFS-223461

Nguyen, T. D., Pham, H. Q., & Bui, N. T. (2022). Credit card fraud detection using LSTM and autoencoder-based deep learning models. Procedia Computer Science, 201, 1045–1052. https://doi.org/10.1016/j.procs.2022.04.135

Ahmed, K., Khan, M. S., & Rehman, S. U. (2023). Credit card fraud detection using XGBoost and deep neural networks: A hybrid model approach. *Journal of Financial Crime*, *30*(2), 523–536. https://doi.org/10.1108/JFC-09-2022-0208

Sharma, A., & Patel, R. (2023). Comparative study of Naive Bayes and deep learning techniques for credit card fraud detection. *Journal of Intelligent Systems*, 32(4), 765–780. https://doi.org/10.1515/jisys-2023-0098

Jurgovsky, J., Granitzer, M., Ziegler, K., Calabretto, S., Portier, P.-E., He-Guelton, L., & Caelen, O. (2021).. Sequence classification for credit-card fraud detection. *Expert Systems with Applications*, *100*, 234–245. https://doi.org/10.1016/j.eswa.2021.112097

Lin, J., Liu, X., & Zhang, D. (2021). Research on the KNN algorithm for detecting fraudulent credit card transactions. *Journal of Intelligent & Fuzzy Systems*, *41*(2), 2071–2081.

Uddin, F., Woo, J., & Lee, B. (2022). Fraud detection using improved K-nearest neighbors classification. *Applied Sciences*, *12*(3), 1187–1202. https://doi.org/10.3390/app12031187

Agarwal, R., & Agarwal, R. (2021). Naive Bayes for fraud detection in banking sector. *International Journal of Computer Applications*, *183*(26), 31–35.

Kamal, A., Sadeghi, A., & Zhou, Y. (2022). Real-time credit card fraud detection with Naive Bayes in big data streams. *IEEE Access*, *10*, 55342–55351. https://doi.org/10.1109/ACCESS.2022.3178412

Ghosh, A., & Gupta, S. (2021). Decision tree-based intelligent model for fraud detection. *Neural Computing and Applications*, *33*(4), 11213–11229. https://doi.org/10.1007/s00521-020-05603-w

Hassan, A., Iddrisu, M. B. H., Adebayo, O., Oladipo, O., Bello, O. A., Ogundipe, A., Mohammed, D., Adebola, F., Alonge, O. A., Alhassan, I., Ibrahim, M., Suleiman, M., Mohammed, A., Bello, M., Usman, A., Musa, A., Lawal, I., & Abdullahi, M. (2022). Hybrid machine learning approach for financial fraud detection using decision trees. *IEEE Transactions on Computational Social Systems*, *9*(2), 456–465. https://doi.org/10.1109/TCSS.2021.3065478

Wang, L., & Zhang, Y. (2023). A probabilistic approach to credit card fraud detection using Gaussian Naive Bayes. *Journal of Financial Data Science and Analytics*, 6(2), 101–115. https://doi.org/10.1007/s42425-023-00189-2