ISSN–2278-5655

AMIERJ Aarhat Multidisciplinary International Education Research Journal

Volume–XIV, Issues– II (B) March – April, 2025

OPEN ACCESS Original Research Article

STUDY AND COMPARISON OF SUPPORT VECTOR MACHINE AND RANDOM FOREST ALGORITHM FOR PAYMENT FRAUD DETECTION IN GIG ECONOMY FREELANCE PLATFORMS

* Gyaneshwari K. Pawar & ** Sumit S. Samanta

Department of Computer Science, Annasaheb Vartak College (Vasai West).

Abstract

In this new era of digitization, fraud detection in online payment (Gig Economy) has become very crucial. The study focuses on the significant selection of appropriate Machine Learning algorithms to stabilize performance factors and detect fraud transactions in the real world. Hence, this paper constitutes the comparison between two algorithms that are Max-Margin Model (Support Vector Machine) & Bagging of Decision Tress (Random Forest) to locate fraudulent transactions. We estimate the effectiveness of both the algorithms on the basis of performance metrics (precision, accuracy, F1-score, recall, ROC AUC), interpretability, robustness to imbalanced data, and scalability. The analysis conveys that the Random Forest triumphs over SVM in most measures, which may offer stability in practical applications for high performance. Hence, conversing the suggestions to apply an algorithm that is suitable for the real world, considering operational impediments for transparency in payment fraud detection systems in freelance platforms.

Keywords: *Online Payment Fraud, Gig Economy, Freelance Platforms, Comparison, Support Vector, Machine (SVM), Random Forest (RF).*

Introduction:

Gig economy means constantly being subjected to last-minute scheduling, specified by the freelancing platforms like Upwork, Fiverr, Freelancer. In this platform you can get work for a short period and also at a shorter notice, but this platform is usually decentralized which increases the risk of fraudulent transactions which can create huge hazardous issues. These fraud transactions can be in any form such as unauthorized transactions, identity theft, and chargeback fraud (Bhattacharyya et al., 2011). We use machine learning algorithms because the traditional rule based approach remained ineffective adapting on complex data which was facilitated using machine learning algorithms showing far better results.

Support Vector works by finding the perfect hyperplane to separate fraudulent and non-fraudulent transactions, whereas RF exploits ensemble learning to increase the efficiency, performance and robustness. Both SVM and RF are unique and effective in their own way when finding payment fraud of complex and non-linearly separable data. This paper thus compares these two algorithms by using some example dataset to focus on computational efficiency and practical applicability.

Literature Review:

Lessmann, S., Baesens, B., Seow, H.-V., & Thomas, L. C. (2015) studied credit scoring so as to identify and predict the creditworthiness. Lessmann and colleagues studied that advanced classification algorithms can be used to judge the authenticity and quality of credit cards so as to improve the risks of transaction information based using credit cards.

Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2003)

They aimed to explain the Support Vector algorithm key steps, techniques such as processing the data previously, effective use of kernel and train models to use SVM so that the model will be able to classify, tune and prepare effective data to uplift the performance. It was a technical report intended to connect and harmonize the theory and practical approach towards adopting SVM techniques.

"Data Mining for Credit Card Fraud: A Comparative Study" by Siddhartha Bhattacharyya, Sanjeev Jha, Kurian Tharakunnel, and J. Christopher Westland,2011

Differentiates the effectiveness between various fact scraping techniques such as SVM which uses vector values for support, RF algorithm, and LR based on regression of logistics—for detecting payment fraud using actual transaction data. The observation mainly highlights that random forest (RF) performs better then both LR and SVM. They reached to this conclusion by comparing their accuracy on decision making and how well the algorithm adapts to complex dataset.

Dal Pozzolo, A., Caelen, O., Le Borgne, Y.-A., Waterschoot, S., & Bontempi, G. (2014).

It emphasizes practical challenges in fraud detection, advocating for adaptive models like random forests, XGBoost, and irregularity finding methods which handle class imbalance and concept drift. It stresses balancing accuracy with operational constraints, continuous model updates, and collaboration between data scientists and domain experts for robust, effective systems. No model is universally better, performance varies based on various use case and estimation metrics (e.g., precision, recall, false positive rate).

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32.

He introduced random feature selection at each node of the decision trees. This unsteadiness reduces correlation between trees, improving overall accuracy and stability compared to single decision trees or standard bagging. This paper emphasizes on the out of the bag error estimator are used to improve the accuracy of the model. He highlights applications beyond prediction, such as clustering and outlier detection

Methodology:

1. Data Collection:

We used a hypothetical dataset of transactions from kaggle platform, comprising 10,00,000 records. Features included Step, Type, Amt, Name Original, Oldbal Original, Newbal Original, Name Destination, Oldbal Destination, Newbal Destination, Is Fraud, is Flagged Fraud. The result column was binary, which indicates 1 as fraudulent transaction class and 0 as non- fraudulent transaction class. The set of data was very uneven, with only 0.1291% of transactions labeled as fraud without pre-processing, reflecting actual transactions.

2. Data Preprocessing:

- **EDA (Exploratory Data Analysis):** Dataset was studied and imbalancing of fraud and legitimate data was observed. Quantile- based outliers were detected.
- **Handling Missing Data:** Non- existing data was found and managed.
- **Normalization:** Features were normalized using StandardScaler for SVM, as it is sensitive to feature scales. Random Forest, being tree-based, did not require normalization.
- **Feature Engineering:** Features were engineered to capture fraud patterns; using variance_inflation_factor multicollinearity variables were reduced.
- **Handling imbalance data:** Data was highly imbalance with class 1 (Not Fraud): 98.9113% and class 2 (Fraud): 1.0887%

3. Model Implementation

1. Max- Margin Model (SVM)

SVM was implemented using the sklearn library in Python. This algorithm seeks to search a separating hyperplane that expand the edge between classes (Cortes & Vapnik, 1995). For non-linearly separable data, we used the exact interpolation techniques (RBF) kernel to convert the data into a multiple dimensional space. Key parameters included:

- **C:** It commutes the values by margin maximization and minimizing mislabeled data points.
- **γ (gamma):** Explains the ascendancy of a single training example, which affects the incurvation of the decision edges.

Parameters were tuned using Bayesian Optimization with a parameter of C = [0.1, 1, 10, 100] and γ = [1, 0.1, 0.01, 0.001] values to be changed

2. Bagging of Decision Tree (RF)

It is an aggregation of decision trees, which was implemented using Scikit-learn's RandomForestClassifier. Individual tree is adapted on a random fragment of the data and its feature, forecasts are combined through majority voting (Breiman, 2001). Key parameters included:

- number of estimators: Specifies the number of trees present in the forest.
- Maximum depth: It describes the highest depth of individual tree.
- Minimum samples split: Specifies minimal trial required to split a node.
- Minimum leaf samples: Specifies minimum samples essential at a terminal node.
- Maximum features: The number of features to examine when looking for the optimal bifurcation.

Parameters were tuned using Bayesian Optimization (with Optuna library) with a parameter grid of n_estimators = [10, 20, 50], max_depth = [None,10,20], min_samples_split = [10,2,5], and min_samples_leaf = [1, 5,10], max_features = ['sqrt', 'log2'].

4. Evaluation Metrics

Evaluation of models were given using the following metrics:

- Accuracy: It is calculated by enclosing both true positive and true negative divided by total no. of predictions.
- Precision: The quality of the model which generates optimistic values.
- Recall (Sensitivity): The positive instances identified by the machine correctly.
- F1-Score: Combines and gives results based on the mean of precision and recall value.
- ROC AUC: It specifies the ability of model to classify binary classes.

5. Experimental Setup

The referred dataset was splitted as training data=80% and testing data=20%. Bayesian Optimization (using Optuna library) was used where no. of trials (n_trials) =100 for both the models.

Results:

1. Performance Metrics: Table presents the performance metrics for fraud class SVM and RF on the testing data.

Model	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**ROC AUC**
SVM	0.98	0.28	0.93	0.43	0.99
Random Forest	0.99	0.78	1.00	0.88	1.00

The correctness of both the algorithm is calculated for the individual model but the remaining parameters report is generated based on the class 1 i.e. fraud based on the test dataset generated after removing outliers**.** The problem of overfitting was generated so to tune it Bayesian Optimization (with Optuna library) was used for both the models and as the accuracy was same for both the model ton resure we used out-of-bag **(OOB)** score which was **0.998.**

2. Interpretability

SVM offered limited interpretability, with decision boundaries being complex, especially with the RBF kernel. Random Forest provided feature importance scores, highlighting transaction frequency and user rating as the powerful features in identifying fraud.

Discussion:

1. **Performance Analysis**

   Random Forest outperformed SVM across all aspects, with higher performance metrics mentioned above in the table. It is likely because of RF's ensemble nature, which reduces overfitting and improves generalization, especially in imbalanced datasets (Breiman, 2001). SVM, while effective, was more sensitive to parameter tuning and struggled with recall, potentially missing some fraud cases.

2. **Handling Imbalanced Data**

   We have applied SMOTE on both the algorithms considering the need of highly imbalanced data to achieve comparable performance, highlighting RF's inherent robustness (Dal Pozzolo et al., 2014).

3. Computational Efficiency

Random Forest was significantly more efficient, with faster training and prediction times, getting it more suitable for actual live fraud detection in large-scale freelance platforms. SVM's computational complexity, particularly with the RBF kernel, limits its scalability (Hsu et al., 2003).

4. **Interpretability**

   Random Forest provided practical explicability through model- agnostic feature outcomes, which is precious in understanding fraud patterns and explaining decisions to stakeholders. SVM, due to its complex decision boundaries, offered little interpretability, which could be a disadvantage in regulatory or business contexts (Lessmann et al., 2015).

5. **Robustness to Noise and Outliers**

   Random Forest's ensemble approach is more powerful to outliers as compared to SVM, a common issue in transaction data. SVM was sensitive to outliers if they lay near the decision boundary, potentially affecting performance (Chen et al., 2004).

6. Practical Implications

For gig economy freelance platforms, Random Forest is recommended as the key model due to its balance of inflated performance, robustness, scalability, interpretability. However, SVM could be considered for smaller datasets where computational resources are not a constraint, and non-linear patterns are suspected to dominate.

Limitations:

This study used a hypothetical dataset, which generally may not be applicable on real-world problems. Further research should verify findings with actual transaction data from freelance platforms. Additionally, the comparison focused on SVM and RF; other algorithms, such as gradient boosting or neural networks, were not considered but may offer further improvements.

Conclusion: This paper compared SVM and Random Forest for payment fraud detection in gig economy freelance platforms. Random Forest emerged as the superior model, offering higher performance, better handling of imbalanced data and practical interpretability. These findings suggest that RF is well-suited for real-world deployment in fraud detection systems, balancing accuracy with operational feasibility. Future research could explore ensemble methods combining SVM and RF or incorporate real-time data to find out the changing patterns in fraud.

References:

1. *Bhattacharyya, S., Jha, S., Tharakunnel, K., & Westland, J. C. (2011). Data mining for credit card fraud: A comparative study. Decision Support Systems, 50(3), 602–613. https://doi.org/10.1016/j.dss.2010.08.008*
2. *Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324*
3. *Chen, Y., Lin, Z., & Xing, E. P. (2004). Pairwise ranking using support vector machines for fraud detection. In Proceedings of the 2004 SIAM International Conference on Data Mining (pp. 123–132). https://doi.org/10.1137/1.9781611972740.12*
4. *Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273–297. https://doi.org/10.1007/BF00994018*
5. *Dal Pozzolo, A., Caelen, O., Le Borgne, Y.-A., Waterschoot, S., & Bontempi, G. (2014). Learned lessons in credit card fraud detection from a practitioner perspective. Expert Systems with Applications, 41(10), 4915–4928. https://doi.org/10.1016/j.eswa.2014.02.026*
6. *Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2003). A practical guide to support vector classification (Technical Report). Department of Computer Science, National Taiwan University. https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf*
7. *Lessmann, S., Baesens, B., Seow, H.-V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. European Journal of Operational Research, 247(1), 124–136. https://doi.org/10.1016/j.ejor.2015.05.030*

Cite This Article:

Pawar G.K. & Samanta, S.S. **(2025).** *Study and Comparison of Support Vector Machine and Random Forest Algorithm for Payment Fraud Detection in Gig Economy Freelance Platforms.* **In Aarhat Multidisciplinary International Education Research Journal**: Vol. XIV (Number II, pp. 106–110).