Mining E-Commerce Transaction Data for Fraud Detection: An Analytical Approach

Shameka Williams

CUS 610: Data Mining I

**Abstract**

In the age of e-commerce platforms, consumers are making more purchases through online retailers. However, these platforms have their security limitations and, thus, are increasingly vulnerable to fraudulent transactions, posing significant challenges to consumers and businesses. This paper will comprehensively analyze many e-commerce transactions to uncover patterns and indications of dishonest behavior. Using a dataset of legitimate and fraudulent e-commerce transactions, we will apply data mining techniques such as association rule mining, decision trees, and random forest classification to identify key attributes and frequent patterns linked to fraudulent activity. These findings will reveal essential factors such as account age, time of the day, and transaction amount that contribute to the likelihood of fraud. Using these results, we propose a predictive model that improves fraud detection accuracy to prevent fraud early on. This study provides actionable insights for enhancing e-commerce fraud detection systems, with potential applications in real-time transaction monitoring and risk assessment.

**Introduction**

Fraudulent activity has an extensive history going back more than twenty centuries. Transact fraud has become more sophisticated in the digital age using newer advanced tools. To give context on the longevity of fraudulent practices, the first recorded fraud happened in 300 BC when two Greek sea merchants, Hegestratos and Zenosthemis, committed insurance fraud (Insert Citation Here from Fraud.com). There are many common fraud types; however, this paper will focus on fraudulent e-commerce transactions. A fraudulent transaction is best defined as “…an unauthorized or illegal activity involving the use of payment instruments or financial systems – typically to obtain money, goods, or services – without proper consent or authorization from the account holder. This type of transaction often involves identity theft, stolen payment information or deception, and is intended to deceive and cause financial harm to the account holder, business, or financial institution.” (Insert Citation Here from Stipe).

Although fraud has advanced, so have the techniques used to detect them. Institutions and e-commerce platforms have implemented data mining techniques such as decision trees, logistic regression, and neural networks to detect abnormal or fraudulent activities. These detection systems have mitigated financial loss and maintained the integrity of the economic system by analyzing vast amounts of financial data from their customers and other data sources to prevent suspicious conduct (Insert Citation Here from Nected). One of the case studies I found was a case study by SPD Technology on credit card fraud.

The case study was conducted by Olena Kovalenko, a project manager at SPD Technology, who explained SPD Technology’s implementation of a machine learning-based credit card fraud detection system for an e-commerce and financial services client. SPD Technology developed a solution using classification models like XGBoost, CatBoost, and LightGBM instead of classical anomaly detection. Their project generated 700 features and applied feature selection to improve the model’s performance. The system categorized transactions based on fraud probability into three levels: low (<10%) allowed, medium (10% - 80%) required extra authentication, and high (>80%) frozen for manual review. One of the challenges of this project was achieving good performance for users with limited transaction history. After 6 months in production, the platform saw reduced fraud-related losses and increased customer satisfaction. SPD Technology maintains and improves its fraud detection accuracy as new data becomes available (Insert Citation Here from SPD). This study shows how classification models can improve the detection accuracy of fraud detection systems and solve fraud issues. Using my research on case studies of fraudulent transactions, I will apply some techniques, such as decision trees, to my dataset.

**Dataset: Fraudulent E-Commerce Transactions**

The dataset used to analyze fraudulent e-commerce transactions is called “Fraudulent E-Commerce Transactions.” It is a synthetic dataset designed to simulate transaction data from an e-commerce platform, similar to Etsy or Shopify, focusing on fraud detection. It contains two files in a CSV format, one with around one million transaction records and the other with twenty-three thousand. It includes 16 features, which are as follows: Transaction ID, Customer ID, Transaction Amount, Transaction Date, Payment Method, Product Category, Quantity, Customer Age, Customer Location, Device Used, IP Address, Shipping Address, Billing Address, Is Fraudulent, Account Age Days, and Transaction Hour. The class feature/attribute is “Is Fraudulent.”

Before applying any data mining techniques, I had to set up my training and testing data. As mentioned before, there are two CSV files. The larger file will represent the training dataset, while the latter will represent the testing dataset. This will help when we make predictive models using decision trees. First, I created a new column named “Address Match” to see if the Shipping and Billing addresses are the same. The data type of this column will be binary but is changed later to a category. Then, I removed unique and unnecessary attributes from the datasets to reduce noise and complexity for preprocessing. These included Transaction ID, Customer ID, IP Address, Transaction Date, Shipping Address, Billing Address, and Customer Location. Then, I checked for missing values in the dataset. I found no missing values in the training or the test set, so I moved on to the next step. Afterward, I changed the “Is Fraudulent” attribute from Boolean to a category and the value to “fraudulent” and “not fraudulent,” respectively, to their Boolean values (0 and 1). Lastly, I discretized my attributes into bins to make them easier to analyze. For example, I cut the “Transaction Amount” data into bins labeled with low, medium, or high.

**Methodology**

I used the Apriori algorithm to discover the standard features of fraudulent transactions. This algorithm helps to determine frequent item sets and association rules that might indicate fraudulent transactions. I had to filter the fraudulent transactions from my training dataset to apply this algorithm to my dataset. Then, I picked out relevant attributes from the dataset. Then, I will use TransactionEncoder to transform the transactions into a one-hot encoded format suitable for Apriori. Then, I called from the Apriori function to find the frequent item sets. The min\_support parameter controls the minimum frequency of an item set to be considered frequency. I ended up setting the value of the support to 0.1. Lastly, I generated the rules from the frequent item sets. I used the metric lift to filter the rules based on my desired criteria.

Using features like transaction amount and account age, I developed a decision tree model that classifies transactions as fraudulent or non-fraudulent. Before implementing the classifier, the data sets need to be encoded. The decision tree model will use entropy for its calculations. In addition, due to the class imbalance of more non-fraudulent transactions than fraudulent ones, I will add the parameter “class\_weight = balanced.” The comparison of this class imbalance is shown below for both the test and training datasets.

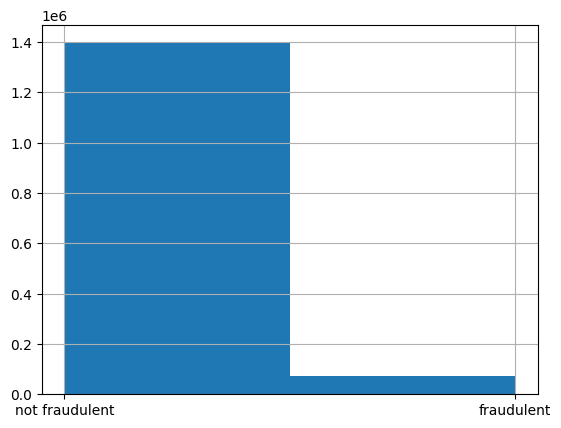


Figure Fraudulent Data Training Dataset

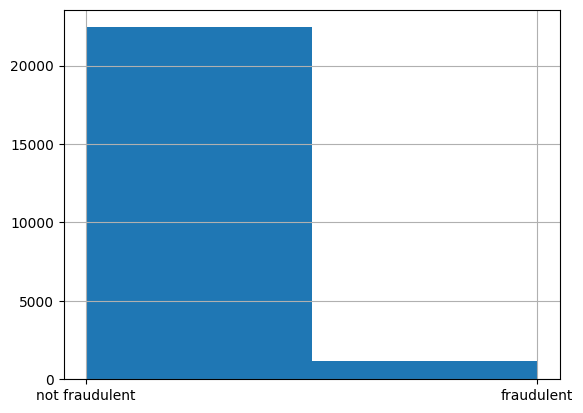


Figure Fraudulent Data Testing Dataset

Then, using the decision tree classifier results, I will create a confusion matrix to display the performance of the decision tree. Along with that, the accuracy and the cross-validation scores will be calculated.

To improve the performance of the classification models, I will be using the ensemble Random Forest. Like the decision tree, the “class\_weight” parameter will be used and set to balanced. Then, the accuracy of the test data will be calculated to see if the performance has improved, and the confusion matrix will be compared to the confusion matrix of the decision tree. Now, let’s examine the results of the application of these techniques.

**Results**

From my Apriori algorithm, the following are a few significant association rules uncovered:

A screenshot of a computer screen

Description automatically generated

Figure Apriori Association Rules

These association rules show that certain transaction features, like using a debit card, high quantities, or veteran accounts, are associated with fraudulent behavior. Some of these rules align with previous research on fraudulent transactions. For example, large quantities and transaction categories like “health & beauty” (Insert Citation Here). Confidence is also high, around 90%, suggesting that these are present and the transactions are likely to be fraudulent. The lift and other metrics suggest that relationships are positive and strongly associated with fraudulent transactions, such as debit card usage or high transaction amounts. In addition, the support values indicate that these conditions are relatively common in the dataset, and the “match” outcome (Address Match) is also persistent. These patterns can detect fraudulent transactions based on these features. However, the overall strength of the rules is moderate, suggesting that further refinement and additional features might improve the model’s predictive performance.

For my decision tree classification, below is the generated confusion matrix.

A yellow and purple squares with black text

Description automatically generated

Figure Entropy Decision Trees Confusion Matrix

The accuracy of the classification technique was 0.88 (88%). The goal of implementing the decision tree was to build a model that classifies transactions as fraudulent or non-fraudulent based on features. The decision tree classified many transactions as non-fraudulent. However, it needs to improve on classifying transactions as fraudulent. According to the cross-validation scores, the average was 0.89 (89%).

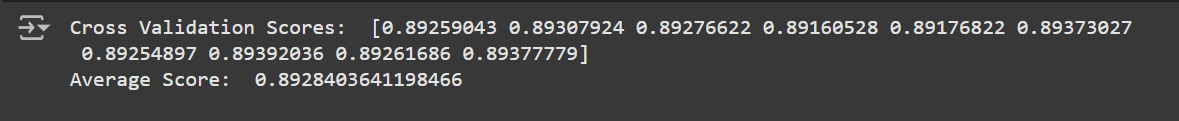


Figure Cross Validation Scores for Decision Trees

To improve the performance of my training model, I applied the ensemble classification method, Random Forest. The confusion matrix for the Random Forest ensemble method is shown below.

A yellow and purple squares with black text

Description automatically generated

Figure Random Forest Confusion Matrix

The accuracy of the test data has improved compared to the decision tree classification. The accuracy for this model was 0.91 (91%). The Random Forest classification did a better job of classifying fraudulent transactions. However, the model still struggles with the classification of fraudulent transactions. Compared to the previous model, it has more misclassified fraudulent than non-fraudulent transactions. Random Forest has proven that it is the best model for this task; however, there is still room for improvement.

**Conclusion**

In conclusion, while a persistent challenge, fraud detection continues to evolve alongside the increasingly sophisticated methods of fraudsters in the digital era. This research applied multiple machine learning techniques, such as the Apriori algorithm, Decision Tree, and Random Forests, to classify and predict fraudulent e-commerce transactions. The Apriori algorithm identified patterns linking features like high transaction amounts and debit card usage to fraud. The Decision Tree model achieved 88% accuracy but struggled with identifying fraudulent transactions due to class imbalance. The Random Forest model performed better, with 91% accuracy, but it still faced fraud classification challenges. While the Random Forest showed promise, improvements through additional features and alternative models such as Naïve Bayes could enhance fraud detection accuracy.

Works Cited

*22 signs of credit card fraud for Merchants (W/ examples)*. 22 Signs of Credit Card Fraud for Merchants (Examples). (n.d.). https://www.tidalcommerce.com/learn/signs-of-credit-card-fraud

Al-Hashedi, K. G., & Magalingam, P. (2021, April 23). *Financial fraud detection applying data mining techniques: A comprehensive review from 2009 to 2019*. Computer Science Review. https://www.sciencedirect.com/science/article/abs/pii/S1574013721000423

DataVisor. (2024, November 25). *Transaction fraud explained: Types, impact, and detection*. DataVisor. https://www.datavisor.com/wiki/transaction-fraud/

Gupta, P. (2024, June 19). *Fraud Detection Data Mining: Techniques & Data Insights: Nected Blogs*. Fraud Detection Data Mining: Techniques & Data Insights | Nected Blogs. https://www.nected.ai/blog/fraud-detection-data#:~:text=Use%20Cases%20of%20Data%20Mining%20in%20Fraud%20Detection&text=Complex%20algorithms%20such%20as%20decision,spot%20and%20stop%20fraudulent%20transactions.

International, F. (2022, July 30). *The history and evolution of fraud*. Fraud.com. https://www.fraud.com/post/the-history-and-evolution-of-fraud#The\_first-ever\_fraud\_case

Jagtap, S. (2024, April 7). *🚨 fraudulent e-commerce transactions 💳*. Kaggle. https://www.kaggle.com/datasets/shriyashjagtap/fraudulent-e-commerce-transactions/data

Kovalenko, O. (2019, July 23). *Credit Card Fraud Detection Case Study*. SPD Technology. https://spd.tech/machine-learning/credit-card-fraud-detection-case-study/

Stripe. (2023, June 26). *Fraudulent transactions 101*. Stripe. https://stripe.com/in/resources/more/fraudulent-transactions-101#what-is-a-fraudulent-transaction

Ughade, N. (2024, September 3). *An overview of data mining for fraud detection*. hyperverge.co. https://hyperverge.co/blog/data-mining-for-fraud-detection/

/