Enhancing Credit Card Fraud Detection through Machine Learning and Deep Learning: A Comparative Analysis

Ya Wei Tsai

April 2024

Abstract

This proposal outlines the approach for enhancing credit card fraud detection using advanced machine learning (ML) and deep learning (DL) techniques to combat increasing fraud sophistication. Utilizing real-life credit card transaction data, the project will apply a variety of ML and DL models and will be supported by Python and its libraries for data manipulation and model implementation.

1 Introduction

In the digital era, combating credit card fraud remains a significant challenge. Our initiative leverages state-of-the-art machine learning (ML) and deep learning (DL) technologies to elevate the detection of fraudulent activities, thereby bolstering our defenses against fraudsters' ever-evolving tactics. Our strategy involves not just refining and tweaking existing models from the latest research but also synthesizing them through an ensemble method to forge a more potent detection system. Through meticulous data processing of existing datasets, we aim to cultivate a fraud detection model of greater precision.

The core of our approach lies in harnessing the distinct strengths of individual models via ensemble learning, enabling a synergistic enhancement of fraud detection capabilities. Tailoring these models to the unique aspects of the data improves efficacy, while rigorous data processing guarantees training on high-quality, pertinent data, augmenting the accuracy of fraud detection.

Furthermore, recognizing the limitations of strictly supervised learning frameworks, our study integrates Autoencoders to navigate the challenges presented by dynamic fraud patterns. Autoencoders, trained exclusively on normal transaction data, excel in identifying fraud as deviations from these patterns. Despite the high initial investment for training Autoencoders, their pivotal role in data labeling paves the way for refining supervised models, ensuring our system's adaptability and effectiveness in the face of the dynamic nature of credit card fraud. This comprehensive and adaptive strategy underscores our commitment

to maintaining a robust fraud detection system capable of confronting the nuanced threats of today's digital landscape.

2 Related Work

In the study titled "Credit Card Fraud Detection Using Machine Learning Techniques: A Comparative Analysis," researchers addressed the challenge of class imbalance inherent in fraud detection datasets. They preprocessed real-life credit card transaction data using a hybrid approach that included both under-sampling and over-sampling techniques. Their model development focused on several machine learning techniques, including Naïve Bayes, K-nearest Neighbor, and Logistic Regression. They chose performance metrics such as accuracy, sensitivity, specificity, precision, and the Matthews Correlation Coefficient to evaluate their models. Notably, the K-nearest Neighbor model demonstrated significant performance across these metrics, indicating the effectiveness of the hybrid sampling approach in classifier performance. The implications of this study for our project are substantial, highlighting the critical importance of appropriate data sampling methods in preprocessing, a key consideration for our fraud detection models.

Another relevant piece of research is "Fraud Detection using Machine Learning and Deep Learning," which explored the application of ensemble learning in fraud detection. The study used European credit card transaction data, processed with Principal Component Analysis to obtain 30 main input features for the models. The researchers experimented with a meta-classification strategy by integrating several base classifiers, including decision trees, Naïve Bayesian, and K-nearest neighbor algorithms. They assessed the performance of their ensemble approach by measuring the classification accuracy of both correct and incorrect instances of data. Their findings showed a 28% performance improvement when the Naïve Bayesian algorithm was applied at the meta-level. For our project, this strategy's success underscores the potential of ensemble models and meta-classification techniques, encouraging us to explore the integration of multiple base classifiers to potentially enhance the detection capabilities of our system.

In the paper by Raghavan and El Gayar from 2019, titled "Fraud Detection using Machine Learning and Deep Learning," the authors present a comprehensive comparison of machine learning and deep learning techniques to detect credit card fraud. They utilize datasets from the European, Australian, and German domains, understanding that fraud dynamics have no set patterns, making detection particularly challenging. Their methodology involves benchmarking machine learning methods such as k-nearest neighbor (KNN), random forest, and support vector machines (SVM), alongside deep learning methods like autoencoders, convolutional neural networks (CNN), restricted Boltzmann machines (RBM), and deep belief networks (DBN). They aim to assess the validity of these methods across datasets of different sizes and complexities.

For data preprocessing, they rely on principal component analysis (PCA) to

transform all fields except time and amount, addressing the issue of high dimensionality. Their experimental setup includes various libraries in Python, such as NumPy, Pandas, Keras, Scikit-Learn, and TensorFlow, occasionally using Rstudio for data cleaning. Hyperparameters and feature reduction techniques are finely tuned to improve classifier performance.

Their evaluation employs the Area Under the ROC Curve (AUC), Matthews Correlation Coefficient (MCC), and a cost of failure metric to provide a rounded measure of model performance. The study concludes that while models like SVM and CNN show promise with larger datasets, ensemble approaches that combine SVM, random forest, and KNN can enhance performance on smaller datasets. Autoencoders, trained only on normal transactions, show potential in identifying fraud as a deviation from normal patterns, especially in dynamic environments where fraud patterns evolve.

The results and approaches discussed by Raghavan and El Gayar have direct implications for our project. They suggest that an ensemble model that integrates various machine and deep learning methods might offer a more nuanced and adaptive approach to fraud detection, especially when dealing with datasets that vary in size and complexity. This information can help us tailor our methodology to create a fraud detection system that is dynamic and capable of learning from new data patterns as they emerge .

These studies provide us with valuable insights into the methodologies and challenges of credit card fraud detection. They not only benchmark the effectiveness of various machine learning and deep learning techniques but also shed light on the importance of data preprocessing and the selection of performance metrics. As we embark on our project, these research findings offer a foundation upon which we can build and refine our approach, giving us the necessary background to develop a sophisticated and adaptable fraud detection system.

3 Plan of Action

Current State of the Project:

Our project is centered on analyzing the *Fraudulent Transactions Data* from Kaggle (`https://www.kaggle.com/datasets/chitwanmanchanda/fraudulent transactions-data`). We are now using K-Nearest Neighbors (KNN), Convolutional Neural Networks (CNN), Logistic Regression, Linear and Lasso, to detect fraudulent transactions . One of the roadblocks I have met is the computational challenge previously encountered with KNN, I recognized that the `sklearn` package was constrained by its inability to leverage GPU acceleration for large datasets. To circumvent this limitation, I have transitioned to employing the `cuml` library, which is designed to interface seamlessly with GPU resources. This pivotal adjustment has significantly expedited the training process, enabling us to harness the full potential of our computational resources and rapidly iterate on our models. The `cuml` library's GPU-accelerated algorithms have thus been instrumental in refining our model's performance, ensuring that

our machine learning pipeline remains robust and highly efficient. In subsequent training phases, we may encounter scenarios where high-speed computation is necessary. To address this, it will be essential to judiciously adjust the model architecture and design to train our models more efficiently. Additionally, we face another challenge: the dataset is very imbalanced, with only 1% being fraudulent data. Therefore, we have to use the Matthews Correlation Coefficient (MCC) to evaluate our model's performance.

Remaining Steps to Complete the Project:

1. **PCA Dimension Reduction:** We plan to leverage the Explained Variance Ratio to determine the optimal number of components for Principal Component Analysis (PCA), aiming to efficiently reduce the dimensionality of our data.
2. **Ensemble Learning Implementation:** We plan to enhance our model's predictive capabilities by integrating ensemble learning techniques. This approach involves combining multiple models to leverage their collective strengths, aiming to achieve superior performance in fraud detection.
3. **Incorporating Naïve Bayesian Algorithm:** Our strategy includes exploring the Naïve Bayesian algorithm's potential benefits. Given its probabilistic foundation, we anticipate that it could significantly contribute to identifying fraudulent transactions within our dataset.
4. **Adjusting Deep Learning Architectures:** To further improve our models' accuracy, we intend to refine the architecture of our deep learning networks. This adjustment will be carefully tailored to address the unique characteristics of our dataset, optimizing the models for better performance.

Software and Tools

*

- NumPy, Pandas for data manipulation.
- Scikit-learn for machine learning algorithms.
- TensorFlow, Pytorch for deep learning model implementation.
- cuml, cupy for training our models on GPU
- matplotlib for visualization

Custom software modules may also be developed for specific tasks within the fraud detection process.

Anticipated Roadblocks:

Imbalanced Data: A significant challenge we face is the imbalanced nature of our dataset, which is common in scenarios involving fraudulent transaction detection. Due to the disproportion between the number of fraudulent and non-fraudulent transactions, there is a high risk of the model being biased towards the majority class. To address this issue, we are focusing on using the Matthews Correlation Coefficient (MCC) as our metric. MCC is particularly effective in such conditions as it provides a more reliable measure of model quality across unbalanced datasets. This metric helps ensure that our model accurately identifies fraudulent transactions without being overly biased towards the more frequent non-fraudulent transactions

Long Training Time: One significant challenge we foresee is the potential for excessively long computational times. To mitigate this, we will need to adjust our model architectures appropriately and utilize GPU resources strategically to ensure timely and efficient processing.

4 Evaluation of Work

To effectively evaluate the performance of our model in the context of fraud detection using machine learning and deep learning techniques, we will employ two critical metrics: the Area Under the Receiver Operating Characteristic Curve (AUC) and the Matthews Correlation Coefficient (MCC). These metrics have been chosen due to their relevance and effectiveness in assessing the quality of binary classification models, which is central to fraud detection tasks.

The Accuracy is defined as the ratio of correctly predicted observations to the total observations:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

where TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives.

However, in the case of imbalanced datasets, which is a common scenario in fraud detection where fraudulent transactions are much less frequent than legitimate ones, Accuracy alone can be misleading. Therefore, we also employ the Matthews Correlation Coefficient (MCC), which is a more reliable statistical rate which produces a high score only if the prediction obtained good results in all of the four confusion matrix categories (true positives, false negatives, true negatives, and false positives), proportionally both to the size of positive elements and the size of negative elements in the dataset.

The MCC is calculated using the formula:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}}$$

The use of MCC is particularly advantageous in our project as it provides a more nuanced view of the model's performance, especially important in our

imbalanced dataset where the prevalence of non-fraudulent transactions could otherwise overshadow the less frequent but more important fraudulent cases. By using MCC, we ensure that both classes are taken into account and the measure is not biased by the majority class, hence giving us a balanced measure of our model's effectiveness.

Additionally, we will use the CNN model from the study "Fraud Detection using Machine Learning and Deep Learning" as a benchmark to provide a comparative basis for our model's performance.

References

[1] Awoyemi, John O. and Adetunmbi, Adebayo O. and Oluwadare, Credit card fraud detection using machine learning techniques: A comparative analysis, International Conference on Computing Networking and Informatics, 2017.

[2] Johan Perols, Financial Statement Fraud Detection: An Analysis of Statistical and Machine Learning Algorithms, American Accounting Association, 2011.

[3] Pradheepan Raghavan and Neamat El Gayar, Fraud Detection using Machine Learning and Deep Learning, International Conference on Computational Intelligence and Knowledge Economy, 2019.