Credit Card Fraud Detection Model

Louis Ciccone

ABSTRACT

The task of binary classification is to classify data into two different categories of one and zero. The one and zero could mean apple or banana true or false or in this case credit card fraud or not. This model focuses on using binary classification to direct if there is credit card fraud based on a dataset of credit card transactions. The project consists of different models trying different methods of deep and shallow neural networks and class weight adjustments and meticulous hyperparameter tuning to find the most optimal model to detect credit card fraud. The results showed that because of the 80/20 split in favor of non fraud there was a bias towards non fraud. The goal of the project was to find the most accurate way to detect credit card fraud which can be used to make life easier and faster for credit card companies.

INTRODUCTION

As everything has become more digitized over the years the increase of the use of credit card transactions have come with it. With such an increase in volume of transactions the ability to detect fraud has to come with it. According to the International Journal of Computer Applications, “With the great increase in credit card transactions, credit card fraud has increased excessively in recent years" (Chaudhary, 2012). In order to keep up with such a high increase in fraud better ways have to be implemented to keep up. There have been many different ways as of recent companies have used in order to keep up. The International Journal of Computer Applications stated that “Modern techniques based on Data mining, Machine learning, Sequence Alignment, Fuzzy Logic, Genetic Programming, Artificial Intelligence etc., have been introduced for detecting credit card fraudulent transactions" (Chaudhary, 2012). The methods used in this project is binary classification of credit card fraud transactions into two categories one is a fraud transaction and the other is real transaction. The project created a Deep Neural Network using tensor flow and Keras. The model uses deep and shallow neural networks and was optimized through class weight and hyperparameter tuning. There was a lot of trial and error done with class weights due to the imbalanced data of 80 percent real transactions to 20 percent fraud transactions. This is because even though there was 99 percent accuracy in total without class weighing it was 70 percent accurate at detecting fraudulent transactions. But with class weighing the model was able to be over 90 percent accurate in detecting credit card fraud.

Dataset

The dataset used was found on Kaggle and it was collected and analyzed during a research collaboration of Worldline and the Machine Learning Group. The data was taken through principal component analysis to keep the anonymity of the transactions. Datasets like these are used to help test the accuracy of machine learning algorithms in order to find new ways to detect credit card fraud.

The dataset transactions made by European credit card holders in September of 2013. The transactions were made over the course of 2 days and have 492 labeled as fraud and 284,315 transactions labeled as genuine. It is very unbalanced and only 0.172 percent of the transactions are fraudulent.

Yeh and Lein made the dataset to act as if it were real credit card transactions. It was made to maintain privacy and only contains numerical inputs. The data was changed using a PCA transformation and only left time and amount. This dataset contains over 30,000 transactions labeled either genuine or fake. It tries to mimic real world transactions and shows real world patterns and techniques to help models be able to detect fraudulent transactions.

The aspect of its similarities to real world transactions is really important for this project. It will allow the model to be able to adapt to other real world datasets and identify fraudulent activities. It allows you to perform models on different time based techniques and machine learning approaches. The data distribution is really important too because it allows you to be able to train the model against a skewed dataset.

The dataset resemblance to real world transactions allows you to do a binary classification model. The model will classify between fraudulent and genuine transactions of the span of the dataset. This allows you to use evaluation formulas for imbalanced data like Precision-Recall AUC and F1-Score to find out how efficient the model is.

The dataset had to go through data cleansing in order to maintain accuracy. The model had to remove all null values and fields that were not necessary. This process allows the model to understand the dataset better and develop a more accurate result.

Deep Neural Network without SMOTE

My model Fraud Detection Model2 is a deep neural network that is simplistic and was made with 8 dense layers with 8 neurons in each dense layer using ReLU activation. This was made straightforward to show the effect of using SMOTE on accuracy. The model tries to find patterns in the credit card transactions and classify if they are fraudulent or not. The process I used was the Adam optimization, binary cross entropy and the loss function. The results from the dynamic neural network were very accurate but biased towards non fraudulent transactions because of the imbalance of data. The accuracy was very high and reached 0.9994 percent, but the f1 score was only 0.80. This was because it was able to detect all the non fraudulent data well but not detect the fraudulent data at a good percentage

Detection model with SMOTE and imblearn

The enhanced model of the project uses SMOTE and imblearn to address the issue of an imbalanced dataset. This was fixed with the use of oversampling the minority class to help the model not be biased towards non fraudulent transactions. With a new balanced training environment it was easy for the model to pick up more patterns instead of overfitting to the majority class. The enhanced model did not have a very good f1 score but its accuracy was at 0.9977 and detected non fraudulent transactions at a rate of 0.9897 and fraudulent transactions at a rate of 0.9082. So even with a poor f1 score it is still detecting both at a rate over 90 percent which is really high.

Training and Validation Loss Graphs

These graphs show the model's learning curve over each epoch. Both models have a declining trend in the training loss which indicates that the model is learning from the data. The validation loss shows how well the model works to data it has not seen before. A model that shows a parallel decrease in both shows it is not overfitting. The model without smote shows a steady decrease in loss which is good. The model using SMOTE, the validation loss decreases and aligns well with the training loss which shows the model generalizes well with oversampling the minority class.

Accuracy Graphs

Accuracy graphs mean the higher the accuracy the better the model's performance is. In the case of the project's graphs both models have high accuracy in their relative predictions. However, this is not an accurate representation to distinguish both models because it does not show the bias towards non fraudulent transactions.

Confusion Matrices

Confusion Matrices show a breakdown of true positives, true negatives, false positives, and false negatives. It helps you know if there is a bias to one side more than the other. This was very helpful in realizing how the model without SMOTE was overfitting towards the non fraudulent transactions. This was because the false negatives compared to the true negatives have a rate of 0.81 while the model using SMOTE had a rate of 0.9082.

Conclusion

This project has shown that there is a lot more to consider in a Machine learning model than just the accuracy. For example if the model without using SMOTE was implemented more than 20 percent of the time people would be accused of credit card fraud when they were innocent. By addressing the issue of an imbalanced dataset it allows you to get a more accurate model that can detect fraud at a higher rate. In addition, even though the f1 score was low the model was still a lot more accurate in both fields which show you can't just listen to one metric when evaluating a model.

References

Data Science Horizons Article:
Team DSH. (2023) 'Handling Imbalanced Datasets in scikit-learn: Techniques and Best Practices', Data Science Horizons, 8 months ago, 07 mins [Online]. Available at https://datasciencehorizons.com/handling-imbalanced-datasets-in-scikit-learn-techniques-and-best-practices/ (Accessed 23 March 2024).

Kaggle Dataset:

Machine Learning Group - ULB and 1 Collaborator. (2018) 'Credit Card Fraud Detection', Kaggle [Online]. Available at https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data (Accessed 15 March 2024).

International Journal of Computer Applications:

Chaudhary, K., Yadav, J., and Mallick, B. (2012) 'A review of Fraud Detection Techniques: Credit Card', International Journal of Computer Applications, vol. 45, no. 1, May 2012 [Online]. Available at http://www.ijcaonline.org/archives/volume45/number1/6676-8962 (Accessed 20 March 2024).