**Lincoln Brown**

**LSTM-Based Credit Card Fraud Detection on Synthetic Datasets**

**Milestone 2**

**DSC680-T301**

**Professor Iranitalab**

Business Problem

Credit card fraud is a significant and growing problem for financial institutions and businesses with global annual losses reaching $34.36 billion in 2022 (Kiernan, 2024). Credit card fraud affects more than just the bottom line of these businesses, eroding customer trust, brand reputation, and operational efficiency. Businesses face unique challenges when combating credit card fraud because they do not have direct influence over where purchases are made. This forces them into a reactive position where they must identify fraud quickly and accurately. Customers are offered protection through federal law and card network policies, often requiring financial institutions to reimburse fraudulent transactions (Kiernan, 2024). This is great for protecting customers from undue harm, but it leaves all the financial burden on the banks and merchants, further increasing the importance of reliable fraud detection systems.

Traditional fraud detection methods were often rule based, using criteria such as transaction amount or unusual locations to flag a transaction as fraudulent. These rule-based systems suffered from high false-positive rates and were unable to adapt to the evolving fraudulent techniques and landscape (Patel, 2023). As the world became more digital, the number of transactions rose, as well as the diversity of transaction types, making it even more difficult for traditional fraud detection systems to keep up.

The development of machine learning has enabled fraud detection systems to handle the increased number of credit card transactions, as well as improved the fraud detection capabilities of these systems. This project will investigate how advanced machine learning techniques, in particular Long Short-Term Memory (LSTM) and Random Forest models, can address the growing need for better fraud detection systems.

Background/History

Fraud detection systems started out as simple rule-based approaches that resulted in large amounts of false positives. As the number of transactions increases in these simple approaches, so does the fatigue of managing the large number false positives. Machine learning provides avenues that are more advanced than simple rules such as flag any transaction above a specific dollar amount or a transaction that is made from an unusual location. Machine learning models can be trained on large datasets and are capable of learning from past transactions to identify fraudulent transactions (Patel, 2023).

Further improvements to machine learning models have been made possible using real-time analytics, deep learning, and data mining. However, even with these improvements, there are still challenges in this domain. Notably, the imbalanced nature of fraudulent transactions. Since there are many more legitimate than fraudulent transactions, it can be challenging to produce a machine learning model that does not have a high false-positive or false-negative rate.

Data Explanation  
 The dataset used in this project was obtained from Kaggle. It is s a synthetic dataset, whose author is Erik Altman. The data was generated by IBM to simulate credit card purchases for over 2000 synthetic users with over 20 million total transactions. The dataset contains a transactions dataset and a user dataset.

The transactions dataset contains fifteen features including, User, Card, Year, Month, Day, Time, Amount, Use Chip, Merchant Name, Merchant City, Merchant State, Zip, MCC, ‘Errors?’, and ‘Is Fraud?’.

The user dataset, which contains additional demographic and user-specific information, will not be used in this project. This dataset was excluded in order to prevent adding unnecessary complexity and to allow for future model adaptation without including sensitive user data.

The exploratory data analysis revealed the following features that seemed promising for model input:

* User – This feature was kept because some users were more likely than others to have fraudulent transactions.
* Year – The number of transactions increased year over year. Naturally, the number of fraudulent transactions increased also.
* Day of Week – This feature revealed that most fraud occurs on Thursday, Friday, and Sunday.
* Hour – Most fraudulent transactions occurred between 10 a.m. – 12 p.m.
* Amount – The majority of fraudulent transactions were between $0 - $100.
* Use Chip – This feature describes the use of the credit card when a transaction is made. The three options were ‘Online Transaction’, ‘Swipe Transaction’, ‘Chip Transaction’.
* MCC – This feature is the Merchant Category Code. It shows the type of business were the transaction occurred.
* Is Fraud? – This is the target variable and reflects whether the transaction was legitimate or fraudulent.

Initially, both a Random Forest model and a LSTM model were created on all the records in the dataset. With limited computational resources, training the models took a long time and the results were less than desirable. To reduce the dataset size, the focus was shifted to transactions that were made online.

Although the majority of the fraudulent transactions were made online, the overall online dataset was still highly imbalanced:  
A graph of a graph

Description automatically generated with medium confidence

Methods **Random Forest (RF) Models**

Random Forests are an ensemble learning algorithm that use a tree machine learning algorithm that builds multiple decision trees during the training phase. Each decision tree is composed of a random subset of the data set to measure a random subset of features in each partition (GeeksforGeeks, 2024). Utilizing random subsets of features introduces variability among individual trees, which can help reduce the chances of overfitting and improve overall prediction performance. The RF model aggregates the predictions of many decision trees, which provides more reliable and accurate results compared to a single decision tree.

**LSTM (Long Short-Term Memory) Models**

LSTM models are an improved version of a recurrent neural network (RNN) designed by Hochreiter and Schmidhuber (GeeksforGeeks, 2024). Traditional RNNs have a single hidden state that is passed through time, which makes it difficult for the model to learn long-term relationships. LSTMs improve on this model using a memory cell that allows the model to hold information for an extended period of time. This capability to retain information makes LSTM models effective for sequential data such as time-series or transactional data in credit card fraud detection.

Analysis   
 The goal of the analysis is to evaluate and compare the performance and applicability of the Random Forest (RF) and Short-Term Memory (LSTM) machine learning models. Both models were trained and tested on a dataset comprised of the synthetic online transaction data. The objective is to determine which model offers a better fraud detection system.

The performance metrics used to evaluate the models were:

* Accuracy – is the proportion of all classifications that were correct, whether positive or negative (T, 2024).
* Precision – the proportion of all the model’s positive classifications that are actually positive (T, 2024).
* Recall – the proportion of all actual positives that were correctly classified as positive (T, 2024).
* F1 Score – the harmonic mean of precision and recall (GeeksforGeeks, 2024).
* Area Under the Curve – Receiver Operating Characteristic (AUC-ROC) – The area under the ROC curve, which measures the model’s ability to distinguish between classes (Narkhede, 2022). Higher values are better.

The Random Forest model achieved these results:

|  |  |  |  |
| --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1-Score** |
| 0 | 1.00 | 1.00 | 1.00 |
| 1 | 0.97 | 0.63 | 0.76 |
| **Accuracy** |  |  | **1.00** |
| **Macro Avg** | 0.99 | 0.81 | 0.88 |
| **Weighted Avg** | 1.00 | 1.00 | 1.00 |

A graph with a red square and a red square with white text

Description automatically generated A graph of a function

Description automatically generated

From the above results, we can see that the Random Forest model performed reasonably well at identifying fraud, with an F1 score of 0.76 for fraudulent transactions. The model maintains a recall of 0.63 for fraudulent transactions, but this metric could be improved to correctly identify more false negatives (fraudulent transactions marked as non-fraud). The ROC-AUC of 0.99 indicates that it is capable of distinguishing between the classes well.

The LSTM model achieved these results:

|  |  |  |  |
| --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1-Score** |
| 0 | 1.00 | 0.90 | 0.95 |
| 1 | 0.06 | 0.97 | 0.11 |
| **Accuracy** |  |  | **0.90** |
| **Macro Avg** | 0.53 | 0.93 | 0.53 |
| **Weighted Avg** | 0.99 | 0.90 | 0.94 |

A graph with red squares and black text

Description automatically generated with medium confidence A graph of a function

Description automatically generated

The LSTM model shows a significantly higher recall of 0.97 for detecting fraudulent transactions compared to the RF’s 0.68, indicating that it is better at minimizing false negatives (missed fraudulent transactions). However, it achieves this at the cost of a much lower precision score, which results in a significantly lower F1 score of 0.11 for fraudulent transactions. Therefore, this model is better suited for instances where identifying fraud outweighs the potential for many legitimate transactions being flagged as fraudulent.

Assumptions

Several assumptions were made in this project including that the synthetic dataset is an accurate representation of fraudulent credit card transactions and adequately mimics the purchases of an average American. The features used in this model are adequate sources to actually capture the patterns of fraud. The capabilities of the models created in this project are only used for testing and will need to be retrained and possibly altered in various ways to be applicable to real-world data.

Limitations

The dataset was generated synthetically by IBM. While the breadth of the data is impressive, there is a possibility that the model is biased or will not capture edge cases effectively. Any use of the model in real-world scenarios would require retraining on real-world data.

Ethical Assessment

This study prioritized the ethical use of the data and strove to protect any possible sensitive information. The dataset used for this study was a synthetic dataset, which greatly helps because no actual customer information was used in the project. Additionally, the user dataset was not included in the models used in this project to prevent the necessity for including any potentially sensitive data in real-world deployments.

Challenges

One of the main challenges in this project was addressing the imbalanced dataset, since fraudulent transactions only made up a small percentage of the total. This made it difficult for the models to identify fraud. To mitigate this, class weight balancing was applied, which increased the penalties for misclassifying fraudulent transactions. Another significant challenge was the sheer size of the dataset, which contained over twenty million records. Initial model training sessions took over five hours to complete, which made it difficult to adjust the models under the time constraints. To improve model training time, the models were only trained on online transactions, resulting in a dataset that contained about two million records, but was still sufficient to identify fraud.

Recommendations

Future recommendations for this model would be to work further with the full dataset instead of only online transactions. This will improve the model’s capabilities of identifying fraud across all transaction types. Additional recommendations would include working with different models to see if there are better candidates for identifying fraud.

Implementation Plan

The implementation for this project would first need to begin with identifying the objective of the model. Does the model need to prioritize identifying all fraud with the cost of a large number of false positives? Or is it better for the model to prioritize minimizing the number of false positives, while still capturing as many fraudulent transactions as possible? In the former case, the LTSM model will perform better, if the latter, the Random Forest model is the correct choice. Both models will need to be retrained before implementing them on real-world data. Further steps would be to incorporate the model into the data pipeline, with real-time data being ingested and sent through preprocessing before being sent to the model. The model could then also be set up to continuously learn and improve using incremental training.

The model could be hosted on cloud services such as AWS or Google Cloud Platform. Hosting in the cloud will increase the model’s availability to different geographic areas and will aid in using real-time APIs to ingest data into the model.

Using APIs, the fraud analysis team could also be notified when fraudulent transactions occur, enabling them to further review or notify the customer of the fraudulent transaction.

Future Uses/Additional Applications

The models created in this project are specific in use-case and would not translate directly over to other areas without some reworking. Other potential areas for fraud detection include anti-money laundering, loan default prediction, insurance, and health care.

Conclusion

In conclusion, the two models that were trained and evaluated as part of this project successfully identified fraudulent credit card transactions. Using class balancing to address the imbalance between legitimate and fraudulent transactions proved to be critical to producing models that were capable of detecting fraud.

The Random Forest model was better at producing a balanced approach to identifying fraud, maintaining a high precision, but also demonstrating effective recall. This model will perform better in environments where minimizing false positives is necessary. Alternatively, the LSTM model had a significantly higher recall but suffered a low precision as a result. This model will perform better in environments where detecting fraudulent transactions are worth the cost of a high false positive rate.

Further refinement of the models could be achieved with more time and computational resources. Using the full dataset would help this model perform better for all transaction types instead of those only found in online transactions.

Ultimately, this project provided insights into the methodology behind developing robust fraud detection systems and illustrated how machine learning has made it possible for financial institutions and businesses to detect fraud using more than traditional rule-based systems. These models can be further refined for use with real-world datasets and with some enhancements, may find applicability outside of the target domain of detecting credit card fraud.

References:

GeeksforGeeks. (2024, December 11). *Random Forest algorithm in machine learning*. GeeksforGeeks. <https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/>

GeeksforGeeks. (2024, June 10). *What is LSTM Long Short Term Memory?* GeeksforGeeks. <https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/>

GeeksforGeeks. (2024, October 17). *F1 score in machine learning*. GeeksforGeeks. https://www.geeksforgeeks.org/f1-score-in-machine-learning/

Kiernan, J. (2024, March). *Credit card fraud statistics for 2024*. WalletHub. <https://wallethub.com/edu/cc/credit-card-fraud-statistics/25725>

Narkhede, S. (2022, March 5). Understanding AUC - ROC Curve - towards data science. *Medium*. https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

Patel, K. (2023). Credit Card Analytics: A review of fraud detection and risk assessment techniques. *International Journal of Computer Trends and Technology*, *71*(10), 69–79. <https://doi.org/10.14445/22312803/ijctt-v71i10p109>

T, J. (2024, December 12). *Understanding Machine learning accuracy: Metrics and methods to measure model performance*. Udacity. https://www.udacity.com/blog/2024/12/understanding-machine-learning-accuracy-metrics-and-methods-to-measure-model-performance.html

Appendix:

Below are some code snippets showing the project’s EDA and model creation.

**Missing Values**

**A screenshot of a computer

Description automatically generated**

**Extracting Days of Week and Time**

**A screenshot of a computer

Description automatically generated**

**Fraudulent Transactions by Amount**

A graph of a number of money

Description automatically generated with medium confidence

**Fraudulent Transactions by Day of Week**

A graph of blue rectangular bars

Description automatically generated with medium confidence

**Random Forest Model**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**LSTM Model**

**A screenshot of a computer program

Description automatically generated**

**A screenshot of a computer program

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer program

Description automatically generated**

**10 Questions an audience would ask:**

1. Why did you choose Random Forest and LSTM models?

I chose Random Forest as a baseline because it has strong performance and interpretability. I chose LSTM to investigate Deep Learning’s ability to capture temporal patterns in the data.

1. How did you handle the imbalance in the dataset?  
     
   I used class weight balancing to penalize misclassification of the minority class, which was fraudulent transactions, so that the models would focus on identifying fraud correctly.
2. What were the trade-offs in your models? How do they impact real-world detection?  
     
   The Random Forest model achieved a more balanced precision and recall, which minimized false positives. LSTM prioritized recall, which led to more false positives, but also better detection of fraud. The tradeoff is that false positives can become frustrating for customer experience. I would recommend deploying the RF model in most scenarios, but in highly critical environments where the cost of fraud is high, the LSTM is an appropriate choice.
3. What are the impacts of false positives on customers?  
     
   False positives can lead to legitimate transactions being identified as fraud, which can be frustrating for customers and potentially hurting the business’ reputation.
4. Can your models scale to larger datasets?  
     
   Random Forest models are not as adaptable to larger scales as LSTM models are when paired with the appropriate computational resources.
5. What were some challenges that you faced?  
     
   The original dataset with over 20 million records produced really long training times. I also faced some difficulty with the highly imbalanced dataset which led to misclassification bias until the class weight balancing was implemented.
6. What are the practical implementations of these models?  
     
   These models have applicability in real-time fraudulent transaction detection systems used to generate alerts or flag potentially fraudulent transactions.
7. Did you have any ethical concerns when working on these models?  
     
   I focused only on transactional data, leaving user data out of it, so that these models may be employed without requiring exposure to sensitive user data.
8. What could be done to improve the models in the future?  
     
   Further improvements would include using the full dataset for model training, hyperparameter tuning, and testing additional datasets.
9. Which model do you recommend deploying and why?  
     
   I would recommend the Random Forest model for environments where minimizing false positives is important (such as customer facing applications). The LSTM model is better suited for high-risk environments where identifying fraud is crucial and dealing with the false positives isn’t a strong deterrent.