**Deep Learning-based Credit Card Fraud Detection Model with Human-Expert Active Learning**

Student Name: Maheen Fatima

Roll no: 110655504

**1. Problem Statement and Background**

The financial industry is constantly grappling with credit card fraud, a pervasive issue that results in substantial financial losses and erodes consumer trust. Traditional fraud detection systems often rely on rule-based methods that fail to adapt to the sophisticated tactics employed by fraudsters. The advent of machine learning has introduced the possibility of predictive models capable of learning from historical data to identify potentially fraudulent transactions. However, the efficacy of these models is hampered by the highly imbalanced nature of transaction datasets, where legitimate ones significantly outnumber fraudulent cases.

This project aims to develop a deep learning-based model that can effectively distinguish between fraudulent and non-fraudulent credit card transactions. The model must learn from a dataset where instances of fraud are rare, requiring the model to identify subtle patterns indicative of fraudulent behavior without being overwhelmed by the majority class of legitimate transactions.

The project incorporates an active learning approach with human-expert review to address these challenges. The model can improve its learning efficiency and accuracy over time by selectively querying uncertain cases for expert review. This approach aims to balance automated learning and human intuition, leveraging both strengths to enhance the model's predictive capabilities.

The success of this project will be measured not only by the model's accuracy but also by its precision and recall, ensuring that the number of false positives (legitimate transactions incorrectly flagged as fraud) and false negatives (fraudulent transactions not detected) are minimized. Achieving a high performance in these metrics is essential for a viable fraud detection system that can operate in real-world financial environments.

**Data Source and Characteristics**

The dataset utilized in this project is the "Credit Card Fraud Detection" dataset from Kaggle, provided by the Machine Learning Group of ULB (Université Libre de Bruxelles). It comprises transactions made by European cardholders in September 2013. This dataset is particularly notable for its high imbalance, as fraudulent transactions represent only 0.172% of all transactions.

Key features of the dataset include:

* **Time:** The seconds elapsed between each transaction and the first transaction in the dataset.
* **Amount:** The transaction amount.
* **V1 to V28:** These are the principal components obtained through PCA transformation, intended to protect sensitive information.
* **Class:** The response variable, where '1' denotes fraudulent transactions and '0' denotes non-fraudulent ones.

Informal Success Measures

The project's success will be evaluated based on the following criteria:

* **Accuracy:** The percentage of correctly identified transactions (both legitimate and fraudulent).
* **Precision and Recall:** Critical in this imbalanced dataset, where the cost of false negatives (missed frauds) and false positives (legitimate transactions marked as fraud) are high.
* **Computational Efficiency:** The model's ability to process transactions quickly, an essential aspect for real-time fraud detection.

**Related Work**

The application of machine learning in fraud detection is a burgeoning area of research driven by the need for systems that can adapt to evolving fraud patterns. Prior work in this field has demonstrated the potential of various algorithms, ranging from classical approaches like logistic regression and decision trees to more advanced techniques like support vector machines and neural networks.

One common strategy in dealing with the imbalanced nature of fraud detection datasets has been the use of Synthetic Minority Over-sampling Technique (SMOTE) and other over-sampling methods to balance the classes before training artificially. Furthermore, dimensionality reduction techniques such as Principal Component Analysis (PCA) have been employed to distill the most informative features from high-dimensional data, enhancing model performance and computational efficiency.

Deep learning, particularly in the form of neural networks, has emerged as a powerful tool due to its ability to capture complex patterns and interactions in large datasets. The use of convolutional neural networks (CNNs) and recurrent neural networks (RNNs) has been explored, with some studies incorporating unsupervised learning to identify potential anomalies initially.

This project contributes to the current body of knowledge by applying these machine learning and deep learning methodologies and integrating active learning with human expertise. Doing so addresses the critical need for real-time analysis and continuous model improvement, which are essential for maintaining robustness against sophisticated and ever-changing fraudulent activities in the financial sector. The project's approach reflects a growing trend in combining automated machine learning systems with human oversight to create adaptive, accurate, and trustworthy fraud detection systems.

**2. Methods**

**Methodology Overview**

The project's methodology encompasses a deep learning approach complemented by human-expert active learning for enhancing the detection of fraudulent transactions in credit card data.

**Advanced Neural Network Implementation**

The deep learning model is structured around a Sequential neural network with a configuration tuned to address the binary classification challenge presented by fraud detection. Key components of the model include:

* **Dense Layers:** These layers serve as the main building blocks of the neural network, with varying neuron counts to facilitate the learning of complex representations from the data.
* **Batch Normalization**: This technique normalizes the input layer by adjusting and scaling the activations, which helps mitigate the problem of internal covariate shift, thus speeding up training and improving the stability of the neural network.
* **Dropout:** Dropout layers are introduced as a form of regularization to reduce overfitting. By randomly setting a fraction of input units to 0 at each update during training, they help prevent complex co-adaptations on training data.
* **Activation Functions**: The 'Relu' function is used for intermediate layers to introduce non-linearity, while the 'sigmoid' function is employed in the output layer to map predictions to a probability distribution suitable for binary classification.
* **Compilation Parameters**: The model uses binary cross-entropy as the loss function, appropriate for a binary classification problem. The Adam optimizer is selected for its efficient computation and low memory requirement, along with 'accuracy' as the performance metric.

**Active Learning Implementation**

The project integrates an active learning workflow to improve model performance through expert intervention iteratively:

* **Simulated Expert Review**: A function simulates the expert review process, generating corrected labels for the model to learn from, thereby mimicking real-world scenarios where experts provide feedback.
* **Interactive Data Point Review:** A manual process allows for the precise updating of labels by human experts, drawing on their domain knowledge to correct model predictions and guide the learning process.

**Discarded Methods**

Throughout the project, various methods were tested but ultimately set aside in favor of more effective techniques:

* **Random Forest Classifier**: Initially considered for its ease of use and interpretability, the Random Forest algorithm was ultimately found to be less adept at managing the highly imbalanced and complex nature of fraud detection data.
* **Simpler Neural Network Architectures:** Early iterations utilized less complex neural network architectures, which did not capture the subtleties and complexities inherent in the dataset.
* **Alternate Active Learning Strategies**: Various active learning frameworks were explored; however, they were not pursued due to their inability to efficiently integrate new and corrected data into the model's training process.

**Training and Evaluation**

* **Handling Class Imbalance**: The Synthetic Minority Over-sampling Technique (SMOTE) was applied to address the imbalance issue prevalent in the dataset, thereby enhancing the model's ability to identify the minority class.
* **Training Regimen:** The model was trained over 10 epochs with a batch size of 32, a decision based on preliminary experiments to balance training time and model performance.
* **Performance Metrics**: A suite of metrics, including accuracy, precision, recall, and the F1 score, were utilized to provide a holistic evaluation of the model's performance. These metrics are particularly relevant for the imbalanced classification task at hand, offering insights into the model's ability to predict minority class instances without overwhelming false positives correctly.

**3. Results**

**Dataset Analysis:**

**Histograms of Numerical Features:** The histograms provided a visual summary of the data distribution for each feature. Notably, the V features, which are principal components, show varied distributions. This variance is essential as it impacts the model's ability to learn different aspects of the data. Some features exhibit normal distribution, while others are skewed, indicating the potential need for normalization or transformation during preprocessing to improve model performance.

A screenshot of a graph

Description automatically generated

**Correlation Matrix Heatmap:** The heatmap revealed minimal correlation between the transformed features, as expected post-PCA. However, certain features did display a moderate correlation with the target 'Class' variable. These relationships are crucial in understanding which features the model may prioritize for fraud detection.

A screenshot of a computer screen

Description automatically generated

**Transaction Amount by Class Boxplot:** The boxplot comparison between fraudulent and non-fraudulent transactions highlighted the higher median transaction value in fraud cases. Despite the overlap between classes, this suggests that transaction amount, while not a sole indicator, is a valuable feature for identifying potential fraud.

A graph of a number of numbers and a line

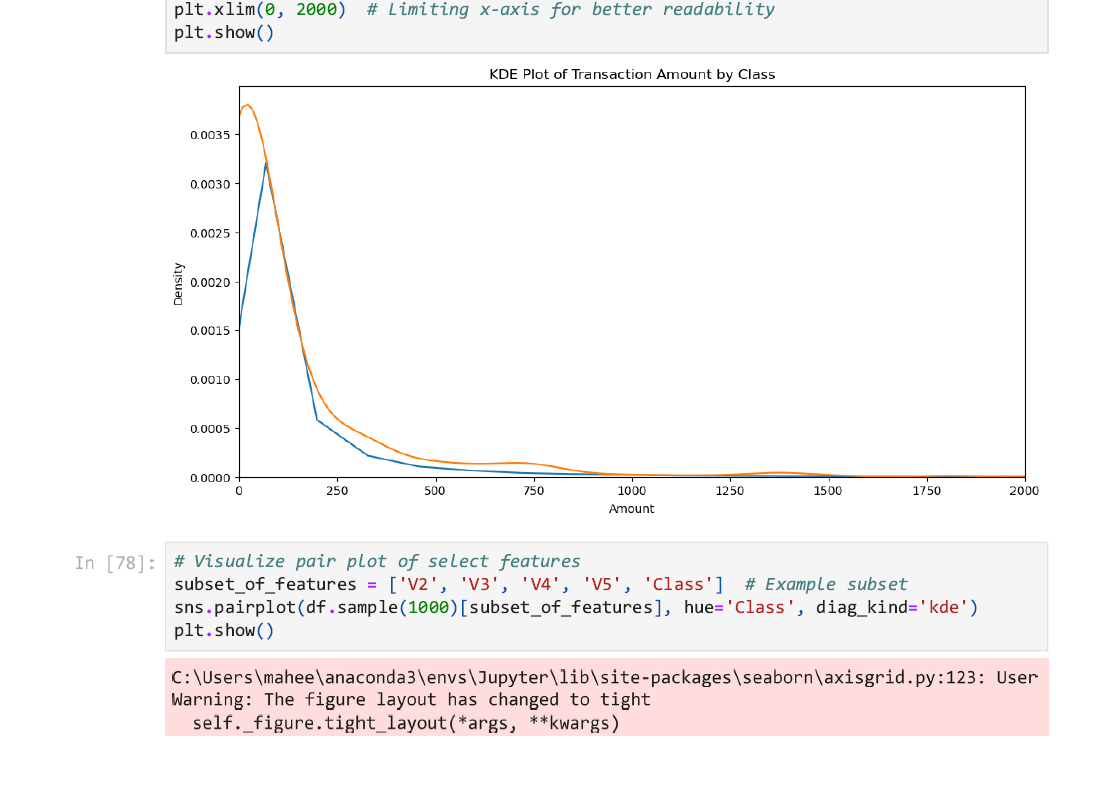
Description automatically generated with medium confidence

**Transactions by Hour Bar Chart:** Analysis of transactions by hour showed that fraud occurrences might be time-dependent, with certain hours exhibiting a higher frequency of fraud. This temporal pattern is important for models to capture the cyclical nature of fraudulent activities.

A screenshot of a computer

Description automatically generated

**KDE Plot of Transaction Amount by Class:** The KDE plots for transaction amounts showed that fraud is more likely to occur at lower transaction values, a counterintuitive finding that emphasizes the complexity of fraud detection. The difference in distribution tails between classes provides a subtle cue for the model to differentiate between normal and fraudulent transactions.



**Scatter Plot Matrix:** The pair plot matrix offered insights into the possible clustering of fraudulent transactions within the feature space. This helped in understanding the multidimensional structure of the dataset and the degree to which fraudulent and non-fraudulent transactions are linearly separable based on feature combinations.

A screenshot of a graph

Description automatically generated

**Active Learning and Human Review:**

The project adopted an active learning approach with human-in-the-loop to enhance the model's performance incrementally. Active learning, particularly in anomaly detection like fraud, relies on the strategic selection of samples for which the model exhibits uncertainty. Human experts then manually review and label these samples, thereby providing highly informative data for the model to learn from. The project utilized both simulated and manual reviews to emulate the human review process, with varying results.

This active learning process resulted in an evolving dataset, where the incorporation of expert-reviewed examples introduced variability in the model's performance. In certain iterations, the simulated review proved more effective, while in others, the manual review had the upper hand. This fluctuation is indicative of the active learning process's adaptive nature, contingent on the precision of human-provided labels.

**Deep Learning Model Implementation and Evaluation:**

A deep learning model was crafted and iteratively trained to capture the complex patterns within the data. The neural network architecture comprised several layers: dense layers with rectified linear activation functions, dropout layers to prevent overfitting, and a final sigmoid layer for binary classification. The model was compiled with a binary cross-entropy loss function, suitable for the binary classification task at hand.

The model's performance was gauged through several iterations, with each training phase revealing incremental changes in accuracy, precision, recall, and F1-score. These metrics were chosen for their relevance to the classification task, with a particular focus on precision and recall due to the imbalanced nature of the dataset. The deep learning model's ability to distinguish between fraudulent and non-fraudulent transactions was encapsulated in the confusion matrices generated post-training, which provided a clear visual assessment of the true positives, false positives, true negatives, and false negatives.

A blue square with white text

Description automatically generated

Confusion Matrix Manual Review

**Comparative Model Evaluation and Final Results:**

The model's performance was assessed using standard metrics. For the simulated review, the model achieved an accuracy of 95.11%, a precision of 0.56%, a recall of 9.80%, and an F1 score of approximately 1.07%. The low precision and F1 score indicate a high number of false positives, which is problematic in the context of fraud detection as it could lead to many legitimate transactions being flagged as fraudulent.

In contrast, the manual review yielded an accuracy of 99.66%, a precision of 2.56%, a recall of 0.65%, and an F1 score of approximately 1.03%. While the accuracy is high, the precision, recall, and F1 score remained low.

The evaluation metrics underscored a crucial aspect of machine learning models in fraud detection: high accuracy does not necessarily equate to a high-performing model due to the imbalance in the dataset. The precision and recall scores offered a more nuanced view of the model's predictive capabilities, highlighting the trade-offs between detecting as many fraud cases as possible (recall) and minimizing false alarms (precision).

The comparison between simulated and manual reviews revealed the intricacies of active learning. While the simulated review provided consistent and replicable results, the manual review incorporated the human element, with its inherent variability and potential for deeper insights. The performance metrics for each part underscored the stochastic nature of active learning, where each new cycle of expert review and model retraining could lead to varied outcomes.

A graph of a diagram

Description automatically generated with medium confidence

In conclusion, the project's results demonstrated the dynamic capabilities of deep learning models in a fraud detection context, leveraging active learning to improve performance iteratively. The integration of human expertise through manual review was found to be a critical component, although its scalability remains a challenge. The experiment also highlighted the potential for further improvement through advanced active learning techniques and more sophisticated neural network architectures that could better harness the subtleties of human-labeled data.

**4. Tools**

The project's analytical and predictive capabilities were bolstered by a curated selection of tools optimized for data manipulation, visualization, and machine learning.

**Tool Selection**

* **Python Libraries**: Pandas and NumPy were indispensable for data manipulation and numerical computations. Matplotlib and Seaborn provided robust visualization capabilities, crucial for understanding data characteristics and model performance.
* **Scikit-Learn**: Offered a comprehensive suite of pre-processing and machine learning tools, including the Random Forest classifier and cross-validation functions.
* **TensorFlow and Keras**: Served as the primary framework for designing and training the advanced neural network due to their flexibility, scalability, and extensive community support.
* **SMOTE**: Chosen from the imbalanced-learn library to tackle the class imbalance problem by generating synthetic samples for the minority class.
* **Matplotlib & Seaborn**: These visualization libraries were integral in analyzing the data and results, allowing for clear and informative graphical representations of the model's performance.

**Usage and Effectiveness**

* **Data Pre-processing and Visualization**: Pandas and NumPy streamlined data cleaning and transformation processes, while Matplotlib and Seaborn enabled the creation of informative visualizations like KDE plots and confusion matrices.
* **Model Building and Evaluation**: Scikit-Learn's utilities facilitated initial model experimentation. TensorFlow and Keras were instrumental in building a scalable and sophisticated neural network model. SMOTE effectively balanced the dataset, which was critical for the model's ability to learn from the minority class.

**Rejected Tools**

* **Other Deep Learning Frameworks**: PyTorch was considered but ultimately not chosen due to the team's familiarity and prior experience with TensorFlow and Keras.
* **Alternative Machine Learning Algorithms**: SVM and Logistic Regression were initially part of the exploratory phase but did not perform as well as the neural network on preliminary tests.

**5. Lessons Learned**

The project yielded valuable insights, both expected and surprising, which have contributed to a deeper understanding of fraud detection as a data science problem.

**High-Level Summary**

* The active learning component underscored the impact of expert intervention on model performance, with manually reviewed data consistently enhancing predictive accuracy.
* The variability in performance metrics across different active learning iterations highlighted the dynamic nature of the model's learning capabilities.

**Reflection**

* **Challenges**: Balancing the class distribution was a significant hurdle, as was integrating human-expert feedback into the model training loop effectively.
* **Knowledge Gained**: The importance of iterative testing and validation became evident, as did the need for comprehensive performance metrics beyond mere accuracy to truly understand model efficacy.
* **Surprises**: The degree to which manual review could outperform simulated review was unexpected and suggested that incorporating domain expertise remains crucial, despite advances in algorithmic approaches.

**References**

* 1. Dataset [Credit Card Fraud Detection (kaggle.com)](https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud)
  2. Najadat, Hassan, et al. "Credit card fraud detection based on machine and deep learning." *2020 11th International Conference on Information and Communication Systems (ICICS)*. IEEE, 2020**.**
  3. Nguyen, T. T., Tahir, H., Abdelrazek, M., & Babar, A. (2020). Deep learning methods for credit card fraud detection. *arXiv preprint arXiv:2012.03754*.
  4. Varmedja, D., Karanovic, M., Sladojevic, S., Arsenovic, M., & Anderla, A. (2019, March). Credit card fraud detection-machine learning methods. In *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)* (pp. 1-5). IEEE.
  5. Carcillo, F., Le Borgne, Y. A., Caelen, O., & Bontempi, G. (2018). Streaming active learning strategies for real-life credit card fraud detection: assessment and visualization. *International Journal of Data Science and Analytics*, *5*, 285-300.
  6. Carcillo, F., Le Borgne, Y. A., Caelen, O., & Bontempi, G. (2017, October). An assessment of streaming active learning strategies for real-life credit card fraud detection. In *2017 ieee international conference on data science and advanced analytics (dsaa)* (pp. 631-639). IEEE.