Fraud Detection in Ethereum Transactions Using Deep Learning MLP and LSTM Models

**Introduction**

In the evolving landscape of digital currencies, Ethereum stands out as a platform not just for cryptocurrency transactions but also for executing smart contracts. As with any financial system, fraud detection is paramount to ensure security and trust. This report details the application of two deep learning models, Multilayer Perceptron (MLP) and Long Short-Term Memory (LSTM), in identifying fraudulent transactions within Ethereum.

**Data Preprocessing**

The dataset comprises various attributes of Ethereum transactions, including but not limited to transaction amounts, timestamps, and contract details. The target variable categorizes transactions into 'fraudulent' or 'non-fraudulent'.

Steps Undertaken:

**Missing Value Treatment**: Ensuring data quality by addressing missing values appropriately.

**Normalization**: Standardizing the numerical features to bring them onto a comparable scale.

**Data Splitting**: Segregating the data into training and testing sets to validate the model's performance on unseen data.

**Class Imbalance Handling with SMOTE**

Given the inherent class imbalance in fraud detection (fraudulent transactions being less frequent), we utilized the Synthetic Minority Over-sampling Technique (SMOTE). This technique helps balance the dataset by synthesizing new examples in the minority class, thereby aiding in better model training and generalization. Here is the visualization plot before and applying SMOTE:



**Information Gain for Feature Selection**

To reduce the operational and time complexity of the model, a feature selection technique has been adopted. Here the optimum feature subset has been selected by obtaining the IG value of each feature with respect to the label data. After addressing the class imbalance with SMOTE, we utilized Information Gain (IG) to select the most relevant features for our models. Information Gain, a concept borrowed from information theory, measures the reduction in entropy or uncertainty in the target variable brought about by the knowledge of a feature.

**MLP Model Implementation**

Architecture

The MLP model is a dense, feedforward neural network consisting of multiple layers. Each layer learns different representations of the data, making the model capable of understanding complex patterns.

**Key Components:**

**ReLU Activation**: Used in hidden layers to introduce non-linearity.

**Sigmoid Activation**: In the output layer for binary classification.

binary\_crossentropy: As the loss function, optimal for binary classification tasks.

**Adam Optimizer**: Chosen for its efficiency and adaptive learning rate properties.

LSTM Model Implementation

Architecture

LSTM networks are well-suited for sequential data analysis. In this model, LSTM layers are used to capture temporal dependencies and patterns in transaction data, which are crucial for fraud detection.

**Key Components:**

**Sequential Processing**: LSTM layers are adept at processing time-series data.

**Binary Classification Output**: Sigmoid activation in the output layer.

**Loss and Optimization**: binary\_crossentropy for loss calculation and Adam optimizer for optimization.

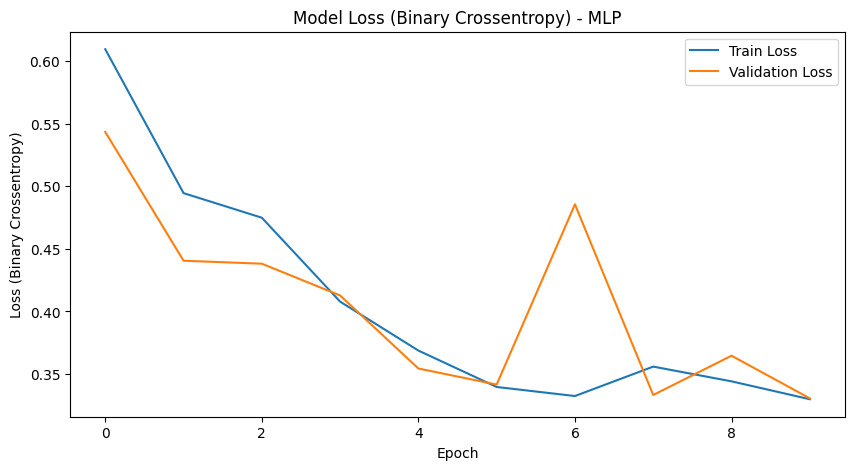
**Training and Validation**

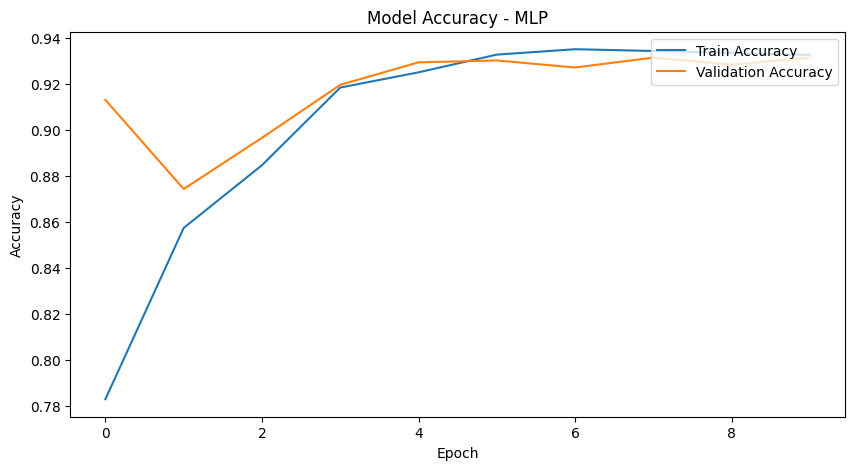
Both models were trained in the prepared dataset, with a portion of the data reserved for validation. This approach enables the evaluation of the models' predictive capabilities and generalization to new, unseen data.

**Analysis of Results**

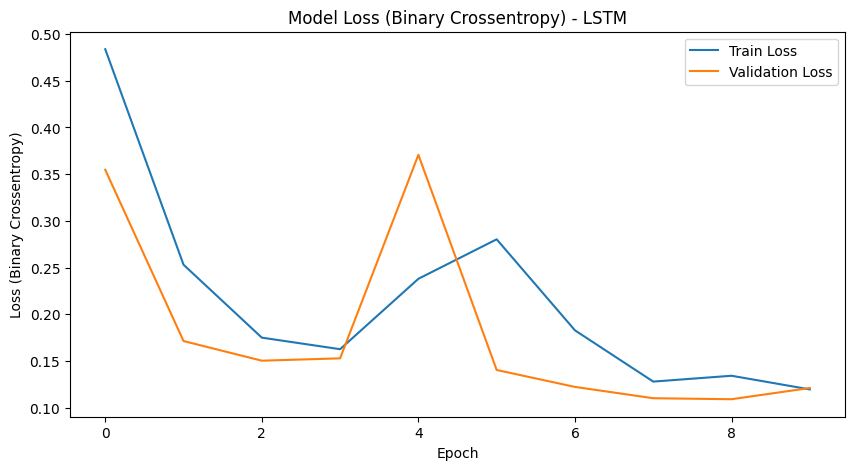
Performance Metrics

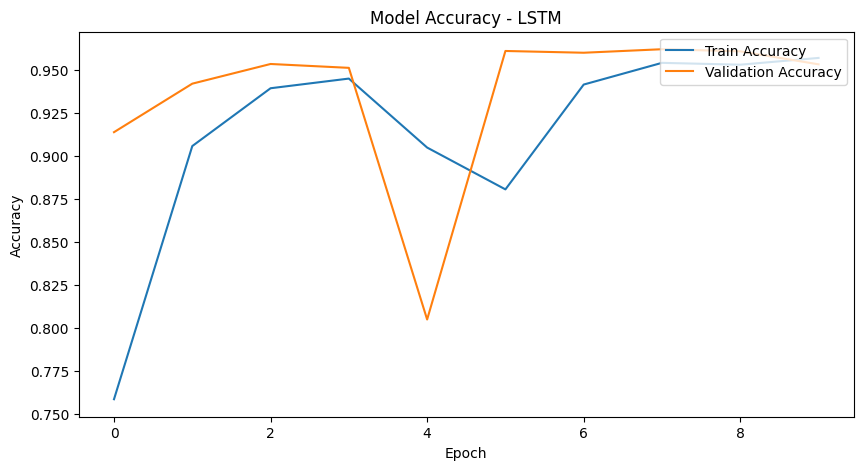
**For MLP:**





**For LSTM:**





Performance Metrics

The models were evaluated based on accuracy.

A comparison between MLP and LSTM performance highlighted the strengths of each model in the context of the given dataset.

**Interpretability**

SHAP (SHapley Additive exPlanations) was employed to interpret the models' decisions. This provided insights into which features were most influential in predicting fraud.

**Conclusion**

The application of MLP and LSTM models has demonstrated significant potential in detecting fraudulent activities in Ethereum transactions. The LSTM model, with its ability to process sequences, showed a nuanced understanding of temporal patterns, which is crucial in the context of transaction data.

**Future Directions**

Further advancements could include exploring more complex neural network architectures, incorporating additional features, and continuously updating the model with new transaction data for sustained performance and relevance.