COSE474-2024F: Final Project Proposal
"Bitcoin Price Prediction Using Sentiment Analysis: A Comparison of LSTM and LSTM-RF Models"

Muhammad Nazim Fu

1. Introduction

Bitcoin, the world's first cryptocurrency, is known for its decentralized nature and high volatility, making price prediction both challenging and essential for investors and researchers. Market sentiment significantly impacts Bitcoin prices, highlighting the value of integrating sentiment analysis into predictive models.

Machine learning, particularly Long Short-Term Memory (LSTM) networks, has proven effective for time-series prediction. However, standalone LSTM models often struggle with the complex, non-linear dynamics of cryptocurrency markets. This study explores a hybrid approach, combining LSTM for sequential data processing with Random Forest (RF) for refining predictions, to enhance accuracy and reliability.

The key objectives are to compare the performance of standalone LSTM and hybrid LSTM-RF models, identify their strengths and limitations, and determine their suitability for Bitcoin price forecasting. This report presents the motivation, methodology, and findings, contributing to advancements in combining deep learning and ensemble techniques for financial predictions.

2. Problem definition & challenges

Bitcoin's price is highly volatile, influenced by market sentiment, economic events, and historical trends. Accurately predicting its price requires models that can handle complex, non-linear relationships. While LSTM networks excel at sequential data modeling, they often struggle with noise and overfitting. This study explores the effectiveness of combining LSTM with Random Forest to improve prediction accuracy and robustness, comparing the standalone LSTM model with the hybrid approach.

Challenges:

- High Volatility: Sudden price changes make predictions difficult.
- Noisy Sentiment Data: Extracting meaningful insights from biased or irrelevant data is complex.
- Non-linear Dependencies: Modeling the intricate relationships between sentiment and prices is challenging.
- Overfitting in LSTM Models: LSTM models, while powerful, are prone to overfitting, especially when the dataset is small or lacks diversity.
- Hybrid Model Complexity: Combining LSTM with Random Forest introduces additional computational and implementation challenges. Aligning the outputs of LSTM with the inputs required by Random Forest requires careful feature engineering and data transformation.

This study focuses on building and comparing models to see if a combined approach can better predict Bitcoin prices and provide more reliable results for users.

3. Concise Description of Contribution

This study introduces a hybrid approach combining Long Short-Term Memory (LSTM) networks with Random Forest (RF) for Bitcoin price prediction. The contribution lies in evaluating and comparing the performance of standalone LSTM models against the hybrid LSTM-RF model, aiming to improve prediction accuracy and robustness in the volatile cryptocurrency market. By integrating sequential data processing (LSTM) with ensemble learning (RF), the research offers insights into enhancing predictive models for financial time-series forecasting, with potential applications in cryptocurrency market analysis.

4. Methods

1. Significance/Novelty:
The novelty lies in combining deep learning with ensemble learning. LSTM models are adept at modeling sequential data but may struggle with noise and non-linearities. Random Forest complements LSTM by refining predictions through its robustness to noise and ability to model complex relationships.

This hybrid approach addresses the limitations of standalone models, offering a potentially more accurate and reliable

forecasting system.

3. Reproducibility:
The methodology involves the following steps:

1. Data Collection: Gather historical Bitcoin prices (open, high, low, close) and sentiment scores from social media/news.
2. Data Preprocessing: Normalize price data using MinMaxScaler, generate sequences, and split data into training and testing sets.
3. LSTM Model: Train an LSTM model with parameters (e.g., 50 units, 32 batch size, 50 epochs) to predict Bitcoin's price.
4. LSTM-RF Model: Use LSTM predictions as input features for Random Forest, which learns the residual errors and refines the predictions.
5. Evaluation Metrics: Compare models using Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).

.

5. Datasets

This project uses two datasets for Bitcoin price prediction:

1. Tweets Dataset (`tweets.csv`) This dataset contains tweets related to Bitcoin and their sentiment scores, sourced from social media platforms. It includes fields such as:
   - **Tweet ID**: Unique identifier for each tweet.
   - **Timestamp**: Date and time of the tweet.
   - **Tweet Text**: The content of the tweet.
   - **Sentiment Score**: A numerical value representing the tweet's sentiment (positive, negative, or neutral).
2. Bitcoin Price History Dataset (`BTC-USD.csv`) This dataset provides historical Bitcoin price data from 2014 to 2024. It includes daily details such as:
   - **Date**: Date of the record.
   - **Open, High, Low, Close**: Bitcoin's daily price metrics.
   - **Volume**: Total Bitcoin traded on that day.
   - **Adj Close**: Adjusted closing price, accounting for factors like stock splits.

Both datasets are synchronized by timestamp, allowing sentiment data and price information to be used together for training the predictive models.

6. Computer Resources

Platform: Google Colab Pro **Operating System**: Managed by Google Colab Pro Programming Language: Python 3.7 **Libraries and Frameworks**: Pandas, NumPy, scikit-learn, TensorFlow **Machine Learning Framework**: scikit-learn (for Random Forest) **Deep Learning Framework**: TensorFlow (for LSTM)

7. Experimental Design and Setup

7.1. Data Collection and Preprocessing

7.1.1. Data Sources

The datasets used for this project are sourced from Kaggle:

- **Tweets Dataset (`tweets.csv`)**: Contains Bitcoin-related tweets with sentiment scores.
- **Bitcoin Price History Dataset (`BTC-USD.csv`)**: Provides historical Bitcoin prices from 2014 to 2024.

7.1.2. Data Alignment

The datasets are aligned based on timestamps to ensure that sentiment scores and Bitcoin price data are correctly synchronized.

7.1.3. Data Preprocessing

- **Normalization**: Both Bitcoin prices and sentiment scores are normalized using MinMaxScaler to scale the values between 0 and 1.
- **Sequence Generation**: The data is converted into sequences to capture the temporal dependencies. Each sequence consists of a fixed number of past time steps and their corresponding sentiment scores.
- **Splitting**: The data is split into training and testing sets, typically with an 80/20 or 70/30 ratio, ensuring that the model can generalize to unseen data.

7.2. Model Development

7.2.1. Standalone LSTM Model

An LSTM model is designed to capture temporal dependencies and sequential patterns in the Bitcoin price data. It includes layers such as LSTM units, dropout layers, and Dense layers. The model is trained on the preprocessed Bitcoin price and sentiment data using the training set.

7.2.2. Hybrid LSTM-RF Model

- The LSTM model is used first to predict Bitcoin prices based on historical price and sentiment data.

- The predictions from the LSTM model are then used as input features for training a Random Forest model.
- The Random Forest model refines the LSTM predictions, helping to improve prediction accuracy.

7.3. Training Process

7.3.1. Training the LSTM Model

The LSTM model is trained using the training dataset with the Mean Absolute Error (MAE) loss function and the Adam optimizer. The model is trained for 50 epochs with a batch size of 32. The validation set is used to monitor the performance during training and prevent overfitting.

7.3.2. Training the Random Forest Model

The Random Forest model is trained using the LSTM-generated predictions and any additional features, such as the sentiment scores, from the training set. The Random Forest model uses 100 estimators (trees) and is trained to improve the overall prediction accuracy.

7.4. Evaluation

7.4.1. Performance Metrics

The models are evaluated using two primary metrics:

- **Mean Absolute Error (MAE)**: Measures the average magnitude of errors in the predictions.
- **Root Mean Squared Error (RMSE)**: Provides a measure of the model's prediction error, penalizing larger errors more heavily.

7.4.2. Comparison

The performance of the standalone LSTM model is compared to the hybrid LSTM-RF model. A lower MAE and RMSE indicate better prediction accuracy and model performance.

Model Performance Comparison

Model	MAE	RMSE
LSTM	0.0044	0.066
LSTM-RF	0.0003	0.017

Table 1. Comparison of Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) for LSTM and LSTM-RF models.

7.5. Model Comparison and Analysis

7.5.1. Quantitative Results

MAE and RMSE are computed for both models, and the results are analyzed to determine which model provides better predictive accuracy.

7.5.2. Qualitative Results

Visualizations, such as line plots comparing the predicted vs. actual Bitcoin prices, are used to assess how closely each model follows the actual price trend over time.

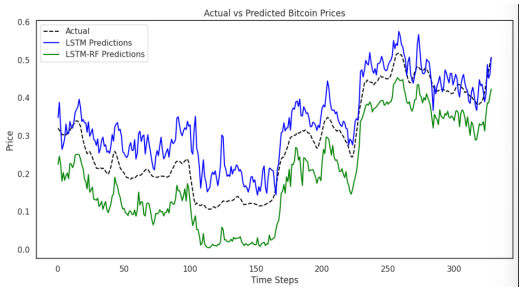

Figure 1. Actual vs Predicted BTC (LSTM LSTM-RF).

7.5.3. Statistical Analysis

Statistical tests, such as paired t-tests, can be used to assess the significance of the performance difference between the LSTM and hybrid LSTM-RF models.

7.6. Discussion

7.6.1. Model Performance

The strengths and weaknesses of both models are discussed, with a focus on accuracy, robustness, and the ability to generalize to unseen data.

7.6.2. Challenges and Limitations

Potential challenges, such as overfitting in the LSTM model or noisy sentiment data, are addressed.

7.7. Future Work

Based on the results, the experimental design may be expanded to incorporate additional features or other advanced models, such as Transformer networks or ensemble methods, to further improve prediction accuracy.

8. State-of-the-art methods and baselines

9. Schedule & Roles

References