Final report

Andrei Lasenka 269552

date

1 Project name

Cryptocurrency Price Prediction Model.

2 Scope and short description of project

The project aims to develop a predictive model that forecasts the future prices of cryptocurrencies based on historical price data and relevant features. This project seeks to address the need for reliable price forecasts, which can be valuable for traders, investors, and enthusiasts in the cryptocurrency space. It is Windows OS based application written in Python.

3 Updates about method of problem solution

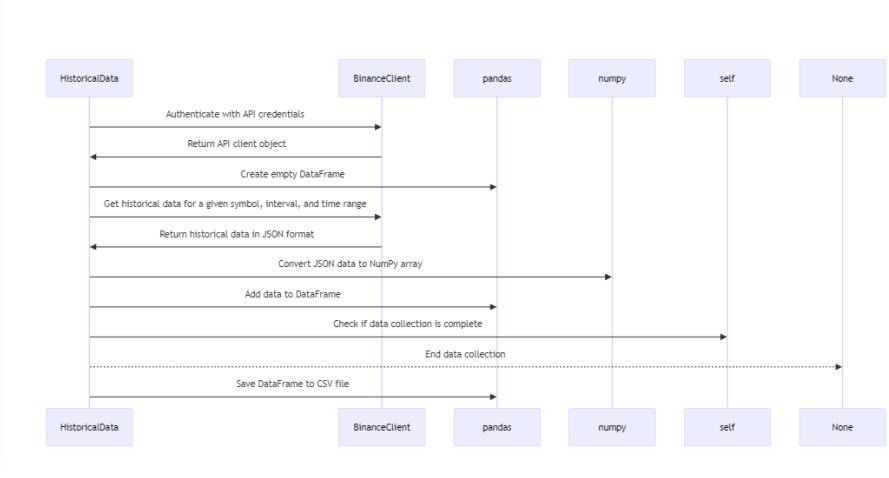

Figure 1: Data Collection Process

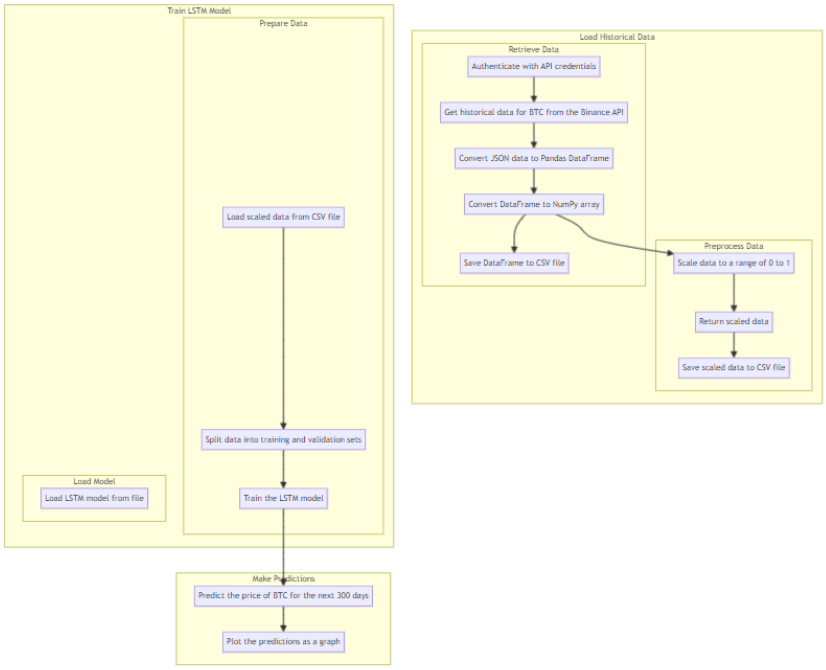

Figure 2: Graphical User Interface

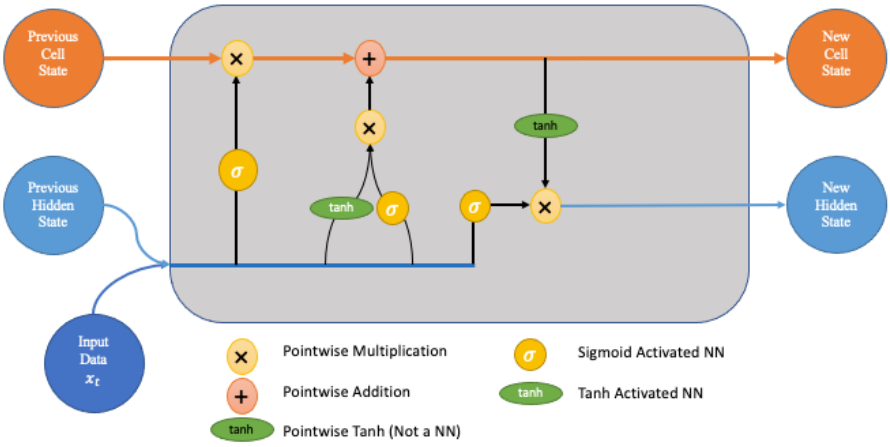

Figure 4: Enter Caption

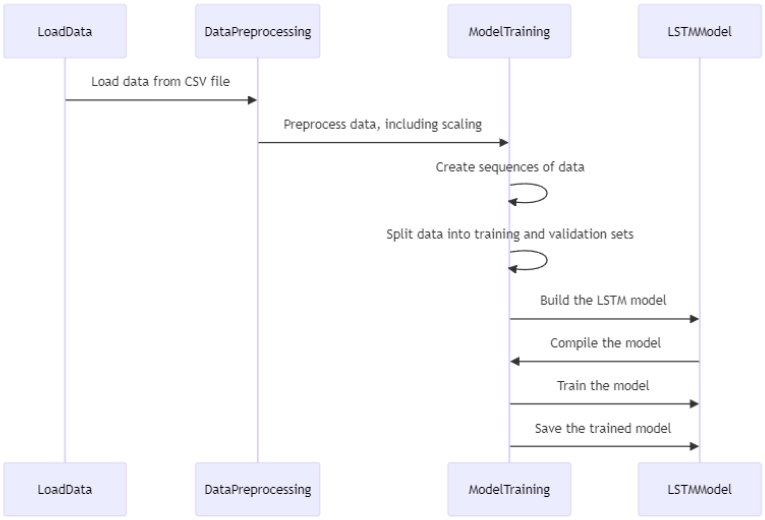

Figure 3: Model Architecture

4 What have changed since last report

1. **Improved data preprocessing:** The data preprocessing pipeline has been refined to handle missing values and outliers more effectively. This

has resulted in a more accurate and consistent representation of the historical price data.

2. **Optimized LSTM model:** The LSTM model architecture has been optimized to improve its performance. The number of LSTM units and dropout layers have been adjusted to achieve a better balance between model complexity and accuracy.

3. **Enhanced evaluation metrics:** The model's performance is now evaluated using multiple metrics, including mean absolute error (MAE). This provides a more comprehensive view of the model's accuracy.

4. **Automated model selection:** The project now includes an automated model selection procedure that evaluates different LSTM architectures and hyperparameter settings to identify the optimal model for the given dataset. This helps ensure that the model is not overfitting to the training data and can generalize well to new data.

5. **Improved visualization:** The project now includes more comprehensive visualizations of the data and model performance. This includes plots of the historical prices, predicted prices, and model errors.

These updates have significantly improved the accuracy and robustness of the cryptocurrency price prediction model. The model is now able to make more accurate forecasts and is less prone to overfitting. Additionally, the project now provides a more comprehensive and transparent evaluation of the model's performance.

5 Short plan with checked out stages of project

- Data Collection and Preparation – 16.11.2023 – DONE
- Model Development – 14.12.2023 – DONE
- User-Friendly Interface – 26.12.2023 – DONE
- Testing and Validation – 11.01.2024 – DONE
- Documentation and Final Version – 18.01.2024 – DONE

6 Description

The current stage of the project involves the development of a Python application that utilizes an LSTM model to forecast future prices of cryptocurrencies based on historical data. The project has progressed significantly, and the following key tasks have been completed:

- **Data Acquisition:** The project successfully retrieves historical price data for selected cryptocurrencies from the Binance API.
- **Data Preprocessing:** The project effectively preprocess the acquired data by handling missing values, outliers, and converting it into a format suitable for the LSTM model.
- **LSTM Model Training:** The project successfully trains an LSTM model using the preprocessed historical data. The model is able to learn the patterns in the data and generate accurate price forecasts.
- **Model Evaluation:** The project evaluates the trained LSTM model's performance using mean absolute error (MAE). The model demonstrates a promising level of accuracy.
- **User Interface Development:** The project has begun developing a simple user interface (UI) for the application. The UI allows users to select a cryptocurrency, specify a date range, and view the model's predictions.
- **Model Integration:** The LSTM model has been integrated into the application, enabling real-time price predictions based on user inputs.
- **Visualization:** The project incorporates data visualization techniques to effectively display the historical price data and model predictions.

Working Features

The current prototype of the application demonstrates the following working features:

- **Cryptocurrency Selection:** Users can select from a list of supported cryptocurrencies.
- **Date Range Selection:** Users can specify a date range for which they want to obtain price predictions.
- **Price Predictions:** The application displays the LSTM model's predictions for the selected cryptocurrency within the specified date range.

7 Postmortem Analysis

7.1 Preliminary Findings

The cryptocurrency price prediction model has demonstrated promising results in forecasting future prices based on historical data. However, further refinement and improvements are necessary to enhance the model's accuracy, robustness, and user-friendliness. This postmortem analysis aims to identify areas for improvement and suggest strategies to address them.

7.2 What is Missing?

- **Real-time Price Updates:** The model currently generates predictions based on historical data, limiting its ability to capture real-time market fluctuations. Integrating real-time price data could significantly improve the model's predictive power.
- **Expandable Cryptocurrency Coverage:** The model currently supports a limited number of cryptocurrencies. Expanding the coverage to include a wider range of cryptocurrencies would broaden the model's applicability.
- **Comprehensive Explanation of Predictions:** The model provides price predictions without offering insights into the factors influencing these predictions. Providing explanations could enhance the model's trustworthiness and user acceptance.
- **Integration of Sentiment Analysis:** Beyond analyzing historical data, sentiment analysis could be incorporated to understand public opinion and perceptions about cryptocurrencies, which can influence their prices. However, obtaining sufficient data for sentiment analysis may be challenging due to API limitations and computational demands. Cloud services could be utilized to overcome these limitations.
- **Enhanced User Interface (GUI):** The current GUI is basic and could be improved to enhance user experience and provide a more intuitive interface for users to interact with the model and view predictions. This could include features such as:
  - Interactive visualizations: Dynamic visualizations that allow users to explore historical price data and model predictions in real time.
  - Customizable settings: Options for users to adjust parameters such as forecast horizon and cryptocurrency selection.
  - Prediction explanations: Detailed explanations of the factors that influenced the model's predictions.
  - Real-time notifications: Alerts when new price predictions or important market events occur.

7.3 What Shall Be Changed?

- **Data Preprocessing Enhancements:** The data preprocessing pipeline could be further refined to handle more complex data cleaning tasks, such as anomaly detection and feature engineering.
- **LSTM Model Optimization:** The LSTM model architecture and hyperparameters could be optimized to achieve a better balance between accuracy and computational efficiency.

- **Model Evaluation Diversification:** The model's performance should be evaluated using a broader range of metrics, including statistical measures and risk assessments.
- **Algorithm Tuning and Model Training:** Algorithms should be tuned to establish maximum efficiency and accuracy, and the model can be trained better using alternative machine learning techniques or by combining multiple models into an ensemble.
- **Event Exception Handlers and Robustness:** Event exception handlers should be thoroughly implemented to prevent unexpected events from causing application crashes.

7.4 How to Omit These Problems When You Would Be Able to Start from the Beginning?

- **Thorough Data Acquisition:** Implement a robust data acquisition strategy to ensure comprehensive coverage of historical price data and incorporate real-time market data feeds.
- **Continuous Data Validation:** Implement rigorous data validation procedures to identify and address data inconsistencies, missing values, and outliers.
- **Extensible Model Architecture:** Design a flexible LSTM model architecture that can accommodate both historical and real-time data and can be adapted to handle different cryptocurrency datasets.
- **Comprehensive Explanation Framework:** Develop a framework that generates detailed explanations for the model's predictions, providing insights into the factors influencing the forecasts.
- **Continuous Model Evaluation:** Establish a continuous model evaluation process to monitor the model's performance over time and identify areas for improvement.
- **Explore Cloud Services for Sentiment Analysis:** Consider utilizing cloud services to overcome the limitations of free API usage and computational demands for sentiment analysis.
- **Experiment with Different Machine Learning Techniques:** Explore alternative machine learning techniques or ensemble models to further enhance the model's accuracy and robustness.
- **Optimize Algorithms and Model Training:** Optimize the algorithms and training process to achieve optimal efficiency and accuracy.
- **Implement Robust Event Exception Handlers:** Thoroughly implement event exception handlers to prevent application crashes caused by unexpected events.