**DEEP LEARNING PROJECT**

HARDIK GOEL B22BB048

HRITIN RAJ B22ES015

AYUSH GOEL B22ES013

VINIT THAKUR B22ES06

**Introduction**

Stock market prediction is a challenging task due to its dynamic nature and sensitivity to external factors. This project focuses on predicting Netflix (NFLX) stock prices using historical stock data and Twitter sentiment analysis. The goal is to evaluate whether incorporating sentiment data improves prediction accuracy.

**Data Overview**

The datasets used in this project include:

1. **Netflix Stock Data (2018–2022)**: Daily stock prices (Open, High, Low, Close, Adjusted Close) and trading volume.
2. **Twitter Financial News Sentiment Data**: Daily aggregated sentiment scores (P\_mean, P\_sum) and tweet counts (twt\_count).

**Files Overview :**

* **NFLX.csv :** Netflix Stock Data
* **Final\_nflx\_data\_2018-2022.csv :** Cleaned and preprocessed Data
* **Scraping&Sentiment analysis.ipynb :** Scaraping of Financial Twitter News Data and Sentimental Analysis of Headlines
* **Lstm\_Sentimental\_analysis.ipynb :** Training model on sentimental anlysis for predicting stock prices
* **Arima\_pred.ipynb :** Prediction based on arima model
* **Stock\_model\_with\_twitter.pt :** Downloaded model from Lstm\_Sentimental\_analsyis.ipynb
* **Final\_pred\_result.ipynb :** Final calculated stock price

**Key Insights from Data Analysis**:

* Sentiment Distribution: 93% negative, 6% positive, 1% neutral.
* Stock price trends show significant volatility during major events like the COVID-19 pandemic.

**Preprocessing Steps**

1. Cleaned missing values and outliers.
2. Normalized features using MinMaxScaler.
3. Created time-lagged features for time-series modeling.
4. Classified sentiments into positive, neutral, and negative categories.

**Methodology**

**Modeling Approaches**

Two approaches were implemented for stock price prediction:

1. **CNN-LSTM Models**:
   * Hybrid deep learning model combining convolutional layers (for feature extraction) and LSTM layers (for capturing temporal dependencies).
   * Features: Stock data (with/without Twitter sentiment).
   * Architecture:
     + Two CNN layers with max pooling.
     + Two bidirectional LSTM layers with 256 hidden units.
     + Dense layers for regression output.
2. **ARIMA Models**:
   * Traditional statistical model for time series forecasting.
   * Features: Stock data (with/without Twitter sentiment).
   * Order: ARIMA(2,1,3) with seasonal components.

**Training Details**

* CNN-LSTM models were trained for 50 epochs with a batch size of 64 using the Adam optimizer and Mean Squared Error loss function.
* ARIMA models were trained using maximum likelihood estimation.
* Data split: 80% training, 10% validation, 10% testing.

**Results**

**ARIMA Model Results**

* **Open Price Prediction**:
  + Without Twitter: Training MSE = 63.91; Testing MSE = 83.10
  + With Twitter: Training MSE = 59.88; Testing MSE = 83.10
* **Adjusted Close Price Prediction**:
  + Without Twitter: Training MSE = 158.32; Testing MSE = 212.77
  + With Twitter: Training MSE = 158.42; Testing MSE = 212.77

*Observation*: Twitter sentiment slightly improved training accuracy for Open prices but had minimal impact on Adjusted Close predictions.

**CNN-LSTM Model Results**

* **Open Price Prediction**:
  + Without Twitter: Training MSE = 1260.21; Testing MSE = 1924.76
  + With Twitter: Training MSE = 177.30; Testing MSE = 2729.50
* **Adjusted Close Price Prediction**:
  + Without Twitter: Training MSE = 1280.14; Testing MSE = 2061.25
  + With Twitter: Training MSE = 303.78; Testing MSE = 2775.36

*Observation*: While CNN-LSTM models achieved better training performance with sentiment data, they suffered from overfitting during testing.

**Comparative Analysis**

1. ARIMA models exhibited more stable performance across training and testing datasets.
2. CNN-LSTM models captured complex patterns but required better regularization to avoid overfitting.
3. Sentiment data showed potential for improving predictions during volatile periods.

**Conclusion and Team Contributions**

**Conclusion**

This project demonstrates that integrating social media sentiment data can enhance stock price predictions under certain conditions:

1. ARIMA models benefited marginally from sentiment data for Open prices but not for Adjusted Close prices.
2. CNN-LSTM models showed significant improvements in training accuracy with sentiment data but faced overfitting challenges during testing.

Future work could explore:

* Advanced sentiment analysis techniques (e.g., aspect-based analysis).
* Incorporation of news sentiment or other social media platforms.
* Ensemble methods combining statistical and deep learning approaches.

**Team Contributions**

1. **HARDIK GOEL** :
   * Project planning and coordination.
   * ARIMA model implementation and evaluation.
   * Final report preparation.
2. **VINIT THAKUR**:
   * Data cleaning, preprocessing, and feature engineering.
   * Exploratory data analysis and visualization.
   * Model evaluation metrics implementation.
3. **AYUSH GOEL**:
   * Design and implementation of CNN-LSTM architecture.
   * Hyperparameter tuning and optimization of deep learning models.
   * Development of training/testing pipelines.
4. **HRITIN RAJ**:
   * Twitter sentiment analysis using polarity scores (P\_mean).
   * Integration of sentiment features into predictive models.
   * Sentiment classification into positive/neutral/negative categories.

**Github Link --**[**Github Link**](https://github.com/Hardikgoel070/Stock-Market-Prediction)