**IBM NAAN MUDHALVAN**

**KINGS ENGINEERING COLLEGE-2108**

**Domine:APPLIED DATA SCIENCE-PHASE4**

**TEAM MEMBERS**

K MANIMARAN :manimaran31072004@gmail.com

D RISWIN :riswinfdo@gmail.com

SHARONJEBASTEEVE:sharonjebasteeve46@gmail.com

P RAMESH :pmramesh03@gmail.com

**STOCK PRICE PREDICTION**

**Introduction:**

such as cryptocurrency price predictionIn this article, we will study the applications of neural networks on time series forecasting to accomplish stock market price prediction. You can find the full code for this tutorial and run it on a free GPU from the ML Showcase.

**Feature engineering :**

Feature engineering is a crucial step in a stock price prediction project. It involves creating, selecting, and transforming relevant features from your raw data to enhance the predictive power of your model. Here are some feature engineering techniques and ideas for a stock price prediction project:

**1. Lagged Features:**

Create lagged features, which are values of the target variable (stock price) from previous time steps. For example, use the stock price from the previous day, week, or month.

**2. Moving Averages:**

Calculate various moving averages, such as simple moving averages (SMA) and exponential moving averages (EMA). These can help capture trends and smooth out noise in the data.

**3. Volatility Indicators:**

Compute measures of volatility, such as historical volatility, average true range (ATR), or Bollinger Bands, to capture the stock's price fluctuations.

**4. Relative Strength Index (RSI):**

Calculate the RSI, a momentum oscillator that indicates whether a stock is overbought or oversold.

**5. Volume Features:**

Incorporate trading volume-related features, such as moving averages of trading volume or volume indicators like the on-balance volume (OBV).

**6. Price Rate of Change:**

Compute the rate of change of stock prices over a specified period to capture momentum.

**7. Seasonal and Time-Based Features:**

Incorporate features that capture seasonality and time-based patterns, such as day of the week, month, or year.

**8. News Sentiment Analysis:**

Analyze news articles related to the stock and extract sentiment scores (e.g., positive, negative, neutral) to gauge the market sentiment's impact on stock prices.

**9. Market Indices and Economic Indicators:**

Include data on relevant market indices (e.g., S&P 500) and economic indicators (e.g., GDP, interest rates) that may influence stock prices.

**10. Fundamental Ratios:**

Utilize fundamental financial ratios, such as price-to-earnings (P/E) ratios, price-to-book (P/B) ratios, and dividend yields, if available.

**11. Technical Indicators:**

Calculate various technical indicators like MACD (Moving Average Convergence Divergence), Stochastic Oscillator, and the Average Directional Index (ADX).

**12. Event Features:**

Incorporate features that represent significant events related to the stock or the company, such as earnings reports, product launches, or mergers and acquisitions.

**13. Correlations and Cross-Correlations:**

Compute correlations between the stock and other related assets, market indices, or macroeconomic variables. Cross-correlations can provide insights into interdependencies.

**14. Market Sentiment Features:**

Include sentiment scores from social media and online forums (e.g., Twitter, Reddit) to gauge public sentiment and its potential impact on stock prices.

**15. Custom Features:**

Create domain-specific features based on your understanding of the stock market and the specific stock you are predicting.

**16. Data Transformation:**

Apply data transformations like differencing, log transformation, or percentage change to make the data more stationary and suitable for data analysis.

**1. Choose the Right Model:**

Select a machine learning or deep learning model that is appropriate for time series forecasting. Some common models for stock price prediction include:

Linear regression

Time series models (e.g., ARIMA, LSTM, GRU)

Random Forest

XGBoost

Model training is a critical step in a stock price prediction project. After you've collected and preprocessed your data and engineered relevant features, you can proceed with training your predictive model. Here's a step-by-step guide on how to train a model for stock price prediction:

**Model training:**

**data set link:** [**https://www.kaggle.com/datasets/prasoonkottarathil/microsoft-lifetime-stocks-dataset**](https://www.kaggle.com/datasets/prasoonkottarathil/microsoft-lifetime-stocks-dataset)

**1. Choose the Right Model:**

Select a machine learning or deep learning model that is appropriate for time series forecasting. Some common models for stock price prediction include:

Linear regression

Time series models (e.g., ARIMA, LSTM, GRU)

Random Forest

XGBoost

**2. Data Splitting:**

Split your dataset into training, validation, and test sets. Common splits are 70-15-15 or 80-10-10. The training set is used to train the model, the validation set is used for hyperparameter tuning and early stopping, and the test set is reserved for the final evaluation.

**3. Model Configuration:**

Define the architecture of your chosen model. Specify the number of layers, units, activation functions, and other hyperparameters. For neural networks, define the architecture of the network, including the input and output layers.

**4. Model Training:**

Train the model on the training data using appropriate loss functions and optimization algorithms. For example, for a deep learning model, you might use mean squared error (MSE) as the loss function and stochastic gradient descent (SGD) or Adam as the optimizer.

**5. Hyperparameter Tuning:**

Experiment with different hyperparameter settings to find the optimal configuration. This includes learning rates, batch sizes, the number of epochs, and regularization techniques. Utilize the validation set to monitor model performance during training.

**6. Early Stopping:**

Implement early stopping to prevent overfitting. Stop training if the model's performance on the validation set starts to degrade.

**7. Model Evaluation:**

After training, evaluate the model's performance on the test set using appropriate metrics such as:

Mean Absolute Error (MAE)

Mean Squared Error (MSE)

Root Mean Squared Error (RMSE)

Mean Absolute Percentage Error (MAPE)

R-squared (R2) for regression models.

**8. Visualize Results:**

Plot the predicted vs. actual stock prices on the test set to visually assess the model's performance. You can also create time series plots to analyze how well the model captures trends and fluctuations.

**Conclusion:**

We have succesfully finish the Stock market prediction has been features,model training and evaluation in- Machine Learning.