

<b>Project Title</b>	<b>Apple Stock Price Prediction</b>
<b>Skills take away From This Project</b>	<b>Basic Python, Data Visualization, Data Cleaning, EDA, Deep Learning (SimpleRNN and LSTM)</b>
<b>Domain</b>	<b>Financial Services</b>

### Problem Statement:

Create a predictive DL model to predict the stock price of Apple with following key points -

- **Deep Learning Models:**
  - Stock Price data is sequential in nature and as you know Recurrent Neural Networks have worked very well on Sequential Data. Hence, explore deep learning approaches like **Recurrent Neural Networks (RNNs)**, **Long Short-Term Memory (LSTM)** networks for the data and compare the performance.
  - You have to perform the analysis on the closing price. We are expecting you to create deep learning models (SimpleRNN and LSTM) that can predict 1 day, 5 days and 10 days behaviour of stock's adjusted closing price.
  - Analyze and handle missing values in the dataset (if any). How would you handle them differently for training the model having a time component in it ?
  - Use **GridSearchCV** to tune hyperparameters like the number of LSTM units, dropout rate, and learning rate.

### Business Use Cases:

#### 1. Stock Market Trading & Investment Strategies

##### ◆ Automated Trading:

- Use the model's predictions to develop an algorithmic trading strategy.
- Automate buying/selling stocks based on predicted price trends.

#### ◆ **Risk Management & Portfolio Optimization:**

- Investors can assess potential future price movements to adjust their portfolio allocations.
  - Predicting stock volatility helps in hedging risks with options and futures trading.
- 

## **2. Financial Forecasting & Time-Series Analysis**

#### ◆ **Long-Term Investment Planning:**

- Predict future stock trends for retirement funds, ETFs, or mutual funds.
- Helps in making data-driven decisions on holding or selling assets.

#### ◆ **Macroeconomic Analysis:**

- Compare Apple's stock trends with economic indicators like interest rates, inflation, and industry trends.
- 

## **3. Business & Corporate Use Cases**

#### ◆ **Company Valuation & Earnings Prediction:**

- Apple can use similar models internally to forecast revenue and profit trends.
- Helps in financial reporting and investor guidance.

#### ◆ **Competitor Analysis:**

- Apply the model to other stocks (e.g., Rivian, NIO, Lucid Motors) to compare Apple's growth with competitors.
- 

## **4. Deep Learning & Research Use Cases**

#### ◆ **Comparing Time-Series Models:**

- Extend the project by comparing LSTMs with GRU, Transformer models, or ARIMA.
- Evaluate which model works best for stock predictions.

#### ◆ **Feature Engineering & Alternative Data:**

- Enhance the model by adding **news sentiment analysis, social media trends, or macroeconomic indicators**.

# Approach for Apple Stock Price Prediction using SimpleRNN and LSTMs

## 1. Problem Understanding

- The goal is to predict **Apple stock prices** using historical data.
- We will use a **deep learning-based approach (SimpleRNN and LSTM)** to model stock price trends.
- The dataset consists of features: **Date, Open, High, Low, close, Adj adj close, and Volume**.

## 2. Data Preprocessing

### 2.1. Load Dataset

- Read the **Apple stock price dataset (AAPL.csv)** using **pandas**.
- Explore the dataset to understand key features.

### 2.2. Feature Selection

- Use the **adj close** price as the target variable for training the model.
- Convert the **Date** column to a **datetime** format if necessary and set it as the index.

### 2.3. Scaling the Data (Optional)

- Apply **MinMaxScaler** (normalization) to scale stock prices between **0 and 1** for better model convergence.

### 2.4. Creating Time-Series Sequences

- Prepare the data for LSTM by creating input-output sequences.
- Use a **window of past n days** to predict the next stock price.

## 3. Model Development

### 3.1. Define SimpleRNN & LSTM Architecture

- Use **Sequential** from **tensorflow.keras** to build a SimpleRNN/LSTM model.

- Layers used:
  - **SimpleRNN/LSTM layer** for learning sequential dependencies.
  - **Dropout layer** to prevent overfitting.
  - **Dense layer** to output the predicted stock price.

### 3.2. Compile the Model

- Use `mean_squared_error` (MSE) as the loss function.
- Optimize using `Adam` optimizer (any other optimizer is also fine).

### 3.3. Model Training

- Train the model with **early stopping** to avoid overfitting.
- Use **ModelCheckpoint** to save the best model.

## 4. Model Evaluation & Prediction

- Use the trained SimpleRNN/LSTM model to **predict stock prices** on the test set.
- Compare actual vs predicted stock prices using visualization (`matplotlib`).
- Calculate **Mean Squared Error (MSE)** to evaluate model performance.

## 5. Insights & Conclusion

- Analyze whether the model effectively captures stock price trends.
- Discuss limitations (e.g., sensitivity to market fluctuations).
- Suggest possible improvements (e.g., adding more features like **news sentiment, trading volume trends, or macroeconomic indicators**).

**Please remember you have to create both SimpleRNN and LSTM models and compare their performance.**

### Results:

We are expecting a detailed report explaining your approach about the project, along with the Live Coding Jupyter Notebook.

### Project Evaluation metrics:

Data Cleaning (20%), Data Pre-Processing (20%), Data Visualization (10%), Feature Engineering (10%), DL Modelling (30%), Model Evaluation and Model Optimization (10%)

### Technical Tags:

#banking, #finance

### **Data Set:**

[Data](#)

### **Data Set Explanation:**

You have to perform the analysis on the adjusted closing price. All the column names are self explanatory. Feel free to make any assumptions that you want.

### **Project Deliverables:**

We are expecting a **detailed report** explaining your approach about the project, along with the **Live Coding Jupyter Notebook**.

### **Project Guidelines:**

Use the latest version of Python for Model Building

### **Timeline:**

Define the project timeline, including milestones and deadlines.

## **PROJECT DOUBT CLARIFICATION SESSION ( PROJECT AND CLASS DOUBTS)**

**About Session:** The Project Doubt Clarification Session is a helpful resource for resolving questions and concerns about projects and class topics. It provides support in understanding project requirements, addressing code issues, and clarifying class concepts. The session aims to enhance comprehension and provide guidance to overcome challenges effectively.

**Note: Book the slot at least before 12:00 Pm on the same day**

**Timing: Tuesday, Thursday, Saturday (5:00PM to 7:00PM)**

**Booking link :** <https://forms.gle/XC553oSbMJ2Gcfug9>

## **LIVE EVALUATION SESSION (CAPSTONE AND FINAL PROJECT)**

**About Session:** The Live Evaluation Session for Capstone and Final Projects allows participants to showcase their projects and receive real-time feedback for improvement. It assesses project quality and provides an opportunity for discussion and evaluation.

**Note: This form will Open on Saturday and Sunday Only on Every Week**

**Timing: Monday-Saturday (11:30PM to 12:30PM)**

**Booking link :** <https://forms.gle/1m2Gsro41fLtZurRA>