Chapter 1

INTRODUCTION

Gold, a precious metal with a long history of fascination and value, continues to be a cornerstone of global financial markets. Its volatile price fluctuations present both opportunities and challenges for investors and traders alike. Accurately predicting future gold prices can be a crucial factor in making informed investment decisions and navigating market uncertainties.

This application aims to tackle the challenge of gold price forecasting by utilizing two powerful statistical models: Long Short-Term Memory (LSTM) and Prophet. Developed within the interactive framework of Streamlit, this web app provides a user-friendly platform for analyzing historical gold price data, comparing different forecasting models, and visualizing future price trends.

1.1 Overview

- The project aims to employ various machine learning techniques to forecast commodity prices, with a specific focus on predicting gold prices in Indian Rupees (INR).
- The code integrates traditional time-series analysis methods, such as Facebook's Prophet, and more advanced techniques like Long Short-Term Memory (LSTM) networks.

1.2 Problem Statement

Predicting future gold prices is a complex task due to the numerous factors influencing the market, including economic conditions, geopolitical events, and investor sentiment. Traditional forecasting methods often struggle to capture the intricate dynamics of financial markets, leading to inaccurate predictions. Additionally, the lack of user-friendly tools for visualizing and comparing different forecasting models can hinder effective analysis and decision-making.

- Forecasting commodity prices is a challenging task due to the inherent volatility and complexity of financial markets. Accurate predictions are crucial for various stakeholders, including investors, traders, and policymakers.

- The code addresses the challenge of predicting gold prices, a significant commodity, using both traditional and modern machine learning approaches.

This application addresses these challenges by:

- Leveraging the power of LSTM and Prophet: Both models have proven effective in capturing complex non-linear relationships within data, making them suitable for gold price forecasting.
- Providing interactive data visualization: Streamlit allows for dynamic exploration of historical gold price data through interactive charts and graphs, enabling users to identify trends and patterns.
- Facilitating model comparison: The application allows users to select and compare the performance of LSTM and Prophet models, providing valuable insights into their strengths and weaknesses in gold price forecasting.

1.3 Objective

This application aims to achieve the following objectives:

- Develop an accurate and user-friendly platform for forecasting gold prices using LSTM and Prophet models.
- Provide interactive visualizations of historical gold price data to identify trends and patterns.
- Enable users to compare the performance of LSTM and Prophet models for informed decision-making.
- Offer valuable insights for investors, traders, and financial analysts in the gold market.

1.4 Dataset Description

The application utilizes a dataset of historical gold prices, including:

- Date: Date of the price recording.
- Price: Gold price in a specific currency in INR.

```
RangeIndex: 11379 entries, 0 to 11378
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Date    11379 non-null  object
 1   INR     11379 non-null  float64
dtypes: float64(1), object(1)
memory usage: 177.9+ KB
```

Fig 1.1 Dataset description

	Date	INR
11372	04/08/2022	1158.7
11373	05/08/2022	1149.4
11374	08/08/2022	1161.8
11375	09/08/2022	1169.1
11376	10/08/2022	1167.1
11377	11/08/2022	1169.7
11378	12/08/2022	1167.2

Fig 1.2 Snip of dataset tail

Chapter 2

SYSTEM REQUIREMENTS

2.1 Software and Hardware Requirements

Software Requirements:

- Python 3.x: The entire codebase is written in Python, leveraging the capabilities of various libraries and frameworks for data analysis, visualization, and machine learning. Ensure Python 3.x is installed on the system.
- Streamlit: Used for creating an interactive web application to visualize data and model results.
- Pandas: Essential for data manipulation and analysis.
- NumPy: Utilized for numerical operations and array manipulations.
- Plotly (Express and Graph Objects): Employed for creating interactive and aesthetically pleasing visualizations.
- Prophet: Implemented for time-series forecasting.
- Scikit-learn: Utilized for traditional machine learning tasks, such as train/test splitting and linear regression.
- Keras (with TensorFlow backend): Applied for implementing LSTM neural networks.
- Integrated Development Environment (IDE): Any preferred Python-compatible IDE, such as Jupyter Notebooks or Visual Studio Code, can be used to run and modify the code.

Hardware Requirements:

- Processor: A multi-core processor is recommended for faster data processing and model training.
- Memory (RAM): Adequate RAM, depending on the size of the dataset and complexity of the machine learning models.
- Storage: Sufficient storage space for storing datasets and model checkpoints.

Chapter 3

SYSTEM ARCHITECTURE AND DESIGN

3.1 Architecture

The system architecture of the gold price forecasting project involves a modular and layered design, incorporating different components for data processing, model training, and visualization. The key architectural elements include:

Data Loading and Preprocessing:

- The system starts by loading historical gold price data from the provided CSV file using the Pandas library.
- Data preprocessing steps, such as handling missing values and normalizing the data, are performed to ensure consistency and facilitate model training.

Data Visualization with Streamlit:

- The Streamlit library is employed to create an interactive web application.
- Users can visualize historical gold prices, explore train/test splits, and interact with forecasting results through the Streamlit interface.

Time-Series Forecasting with Prophet:

- Facebook's Prophet is integrated for time-series forecasting using a traditional approach.
- The dataset is split into training and testing sets, and the Prophet model is trained on historical data.
- Results are visualized using Plotly to compare actual and predicted gold prices.

Gold Price Forecasting with LSTM:

- Long Short-Term Memory (LSTM) neural networks, implemented with Keras, are utilized for more advanced forecasting.
- Data is preprocessed and normalized before being split into training and testing sets.
- The LSTM model is constructed with multiple layers and trained on historical gold prices.

- Predictions are inverse-transformed to the original scale for result interpretation.

Interactive Forecasting and Evaluation:

- Users can dynamically adjust training data percentages, the number of years for forecasting, and other parameters using Streamlit widgets.
- The system provides visualizations comparing actual vs predicted values and a forecast plot for LSTM predictions.

3.2 Design

Modular Code Structure:

- The code is structured in a modular fashion, with functions and classes for specific tasks such as data loading, preprocessing, model training, and evaluation.
- This design facilitates code readability, maintainability, and scalability.

User Interface Design with Streamlit:

- Streamlit is utilized to design an intuitive and user-friendly interface for data exploration and forecasting.
- Widgets like sliders and charts are employed to allow users to interact with the data and models dynamically.

Model Training and Evaluation:

- The code separates the training and evaluation of the Prophet and LSTM models, making it clear and manageable.
- Evaluation metrics for the LSTM model, including Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared (R2), are calculated and displayed.

Forecasting Future Prices:

- The LSTM model is employed to forecast future gold prices for a user-defined number of years.
- The results are presented both in tabular form and through visualizations using Plotly.

Chapter 4

IMPLEMENTATION

4.1 Machine Learning Algorithm Selection

LSTM Neural Network for Gold Price Forecasting:

- Long Short-Term Memory (LSTM) neural networks are selected for their capability to capture complex temporal dependencies in time-series data.
- LSTMs excel at learning patterns and trends in sequential data, making them suitable for forecasting tasks.

4.2 Machine Learning Model Building & Evaluation

LSTM Neural Network:

- The LSTM model is constructed using the Keras library, with a sequential architecture.
- Data is preprocessed, normalized, and split into training and testing sets.
- The model is trained on the training set, and predictions are made on the testing set.
- Performance metrics such as MAE, MSE, and R2 are calculated and displayed for the LSTM model.

```
# Build LSTM model
model = Sequential()
model.add(LSTM(units=100, return_sequences=True, input_shape=(x_train.shape[1], 1)))
model.add(LSTM(units=100, return_sequences=False))
model.add(Dense(units=25))
model.add(Dense(units=1))

model.compile(optimizer='adam', loss='mean_squared_error')

# Train the model
model.fit(x_train, y_train, batch_size=110, epochs=40)
```

Fig 4.1 LSTM model snip

4.3 Frontend Development

Streamlit Interface:

- Streamlit is employed for frontend development, creating an interactive and user-friendly interface.
- Various Streamlit widgets, such as sliders, are used for dynamic user input, allowing users to customize the training data percentage and forecast duration.
- Streamlit charts visualize historical prices, train/test splits, and actual vs predicted values.

4.4 Backend Development

Data Loading and Preprocessing:

- Pandas is utilized for efficient loading and preprocessing of the gold price dataset.
- Missing values are handled, and the data is normalized for LSTM model training.

Machine Learning Model Integration:

- The Prophet and LSTM models are seamlessly integrated into the backend, allowing users to switch between forecasting approaches.
- Model training, prediction, and evaluation are handled within the backend components.

Results Visualization:

- Plotly is employed for creating visually appealing charts that convey the results of the forecasting models.
- The interactive charts allow users to explore and analyze the model predictions easily.

Chapter 5

RESULTS

Fig 5.1 Graphical visualization gold price history

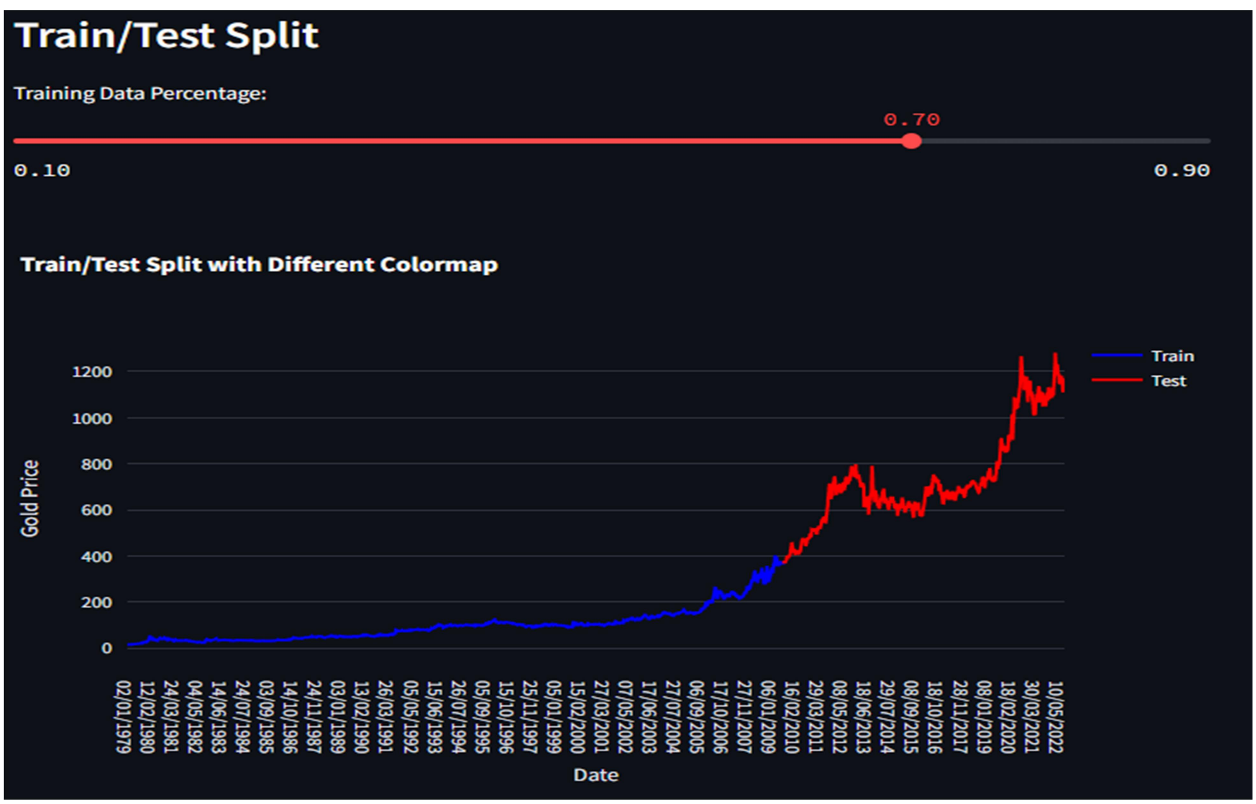

Fig 5.2 Graphical visualization of train test split

Fig 5.3 Slider for years of prediction

Fig 5.4 Forecasted graph for gold price

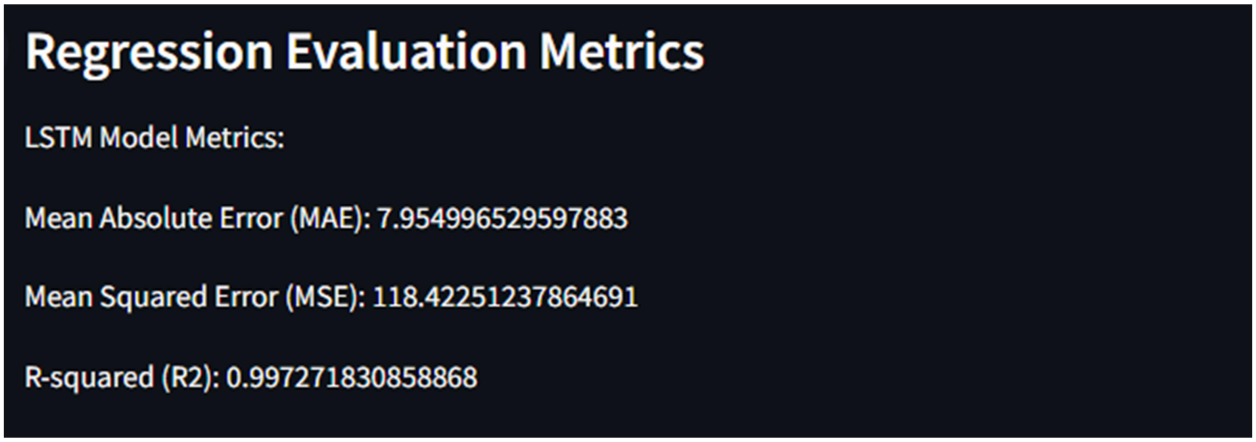

Fig 5.5 Metrics evaluation results for LSTM

Chapter 6

CONCLUSION & FUTURE ENHANCEMENTS

Conclusion

The gold price forecasting project successfully achieves its objectives, offering users a comprehensive platform to:

- Explore historical gold price trends.
- Evaluate forecasting performance using traditional time-series analysis (Prophet) and advanced deep learning techniques (LSTM).
- Interact with the models through an intuitive Streamlit interface.
- Using machine learning algorithms, interactive visualization, and user-friendly design provides a valuable tool for stakeholders interested in predicting gold prices.

Future Enhancements

Hyperparameter Tuning:

- Conducting a more extensive search for optimal hyperparameters for the LSTM model may improve forecasting accuracy.

Ensemble Methods:

- Exploring ensemble methods that combine the predictions from multiple models, such as combining Prophet and LSTM outputs, could potentially enhance overall forecasting performance.

Additional Features:

- Incorporating additional relevant features, such as economic indicators or geopolitical events, might improve model accuracy by considering external factors that influence gold prices.

Real-Time Data Integration:

- Enhancing the system to handle real-time data updates would provide users with the latest information, making the forecasting tool more relevant for dynamic market conditions.