Abstract

This project developed a machine learning web application to forecast gold prices. Using the Facebook Prophet model and historical data, the application successfully generated plausible future price trends. The results validate that this approach is feasible, though future work incorporating more external factors could further enhance predictive accuracy.

Objectives

To accomplish this goal, the project is broken down into the following specific aims:

- **Conduct background research** on the factors influencing gold price trends and examine previous studies on gold price prediction methods.
- **Collect and preprocess historical gold price data** from trusted sources to ensure it is clean, complete, and ready for analysis.
- **Explore and implement multiple prediction models**, including Facebook Prophet, Random Forest, and Long Short Term Memory (LSTM), to determine which provides the most accurate forecasts.
- **Compare the performance of each model** using suitable evaluation metrics to select the best-performing algorithm for integration into the final application.
- **Train and fine-tune the selected model** using the processed historical data to improve prediction accuracy and reliability.

- **Develop a user-friendly web application** that allows users to enter a future date and receive a gold price prediction based on the trained model.
- **Incorporate visualizations** to display historical trends and predicted future prices for better user understanding.
- **Evaluate the final system's performance** and usability, and provide insights into potential limitations and areas for future improvement.

Dataset Overview

The dataset aggregates gold prices (1979-2025) from Yahoo Finance and the World Gold Council. After evaluating various economic indicators, the US Dollar Index (USDX) was selected as an external regressor due to its strong inverse correlation with gold and high feature importance. This was seamlessly integrated into the Facebook Prophet model to enhance forecast accuracy.

Model Selection & Development

To select the best model for gold price prediction, several options were evaluated. Random Forest, XGBoost, and Linear Regression proved inadequate for capturing temporal dependencies without extensive feature engineering. RNNs were excluded due to high computational demands.

Facebook Prophet was chosen as it is specifically designed for time series data, automatically modeling trends and seasonality, and allows for the inclusion of external regressors like the US Dollar Index. The model achieved a 97% R^2 score, demonstrating a strong predictive fit.

This web application was built using a **Python Flask** backend with an **HTML/CSS** frontend and **Plotly** for **interactive charts**. It features a simple static login page leading to a homepage.

The core of the application consists of two main pages:

1. **Dataset Overview**: Displays the dataset's head and tail and an interactive graph comparing actual gold prices against the model's predictions.
2. **Prediction Page**: Allows users to select a specific date and receive the predicted gold price per troy ounce.

Designed for usability, the app provides a straightforward interface for users to view data insights and obtain forecasts.

Limitations

The project faced limitations, primarily computational constraints that prevented the use of more complex models like LSTMs, which might capture deeper patterns. While gold prices are highly volatile, the chosen Facebook Prophet model offered an optimal balance for this application, prioritizing strong performance, ease of deployment, and user accessibility over potential marginal gains in accuracy from a more complex system.

Summary of Key Findings

Facebook Prophet proved to be the most practical model, balancing interpretability, **accuracy (97% R^2),** and **ease of deployment**.

The **US Dollar Index** was the only consistent and significant regressor across a large historical timeline.

Other machine learning models **(Random Forest, XG Boost, RNN** were considered, but either lacked temporal awareness, required complex engineering, or exceeded resource constraints.

The web application was successfully built and tested, with all primary features working as intended.

While only a **one-year prediction window** was implemented, the **results** were promising and **matched real trends** reasonably well.

Design and Implementation

System Architecture

The architecture of the system follows a straightforward and modular web-based structure, consisting of a backend powered by Python and Flask, and a frontend built using HTML, CSS, and JavaScript. The design prioritizes simplicity, responsiveness, and ease of integration with machine learning tools.

```python
@app.route("/predict", methods=["GET", "POST"])
def index():
    prediction = None
    selected_date = None
    min_date = preds['ds'].min().date()
    max_date = preds['ds'].max().date()

    if request.method == "POST":
        selected_date = request.form.get("date")
        selected_date = pd.to_datetime(selected_date)

        result = preds[preds['ds'] == selected_date]
        if not result.empty:
            prediction = round(result.iloc[0]['yhat'], 2)
        else:
            prediction = "No prediction available for this date."

    return render_template("index.html", prediction=prediction, selected_date=selected_date,
                           min_date=min_date, max_date=max_date)
```

Flask code snippet

This web application uses a Flask backend to run the Facebook Prophet forecasting model and process data. The frontend, built with HTML, CSS, and JavaScript, provides an interactive interface for users to request predictions and view results on dynamic graphs. This efficient, lightweight architecture ensures smooth performance and easy maintenance.

The prediction pipeline used a minimal **DataFrame**, **future_gold**, containing only **future dates (ds)** and corresponding US Dollar Index values (**usdx**). This structure allowed the **Prophet model** to generate context-aware forecasts by incorporating the essential external economic regressor.

```
future_gold = gold_model.make_future_dataframe(periods=360)
future_gold = future_gold.merge(usd_future, on="ds", how="left")
```

Code snippet of creating future predictions

```
[34] future_gold
```

	ds	usdx
0	1979-12-26	80.975975
1	1979-12-27	81.001904
2	1979-12-28	81.080060
3	1979-12-31	81.170437
4	1980-01-02	81.257176
...	...	...
12060	2026-04-15	104.555236
12061	2026-04-16	104.526871
12062	2026-04-17	104.547946
12063	2026-04-18	106.717426
12064	2026-04-19	106.696083

12065 rows × 2 columns

Future dates of the predictions and regressor values

Model Implementation

The model implementation relied on Facebook Prophet, which had already been selected for its suitability in time series forecasting. The model was trained using the cleaned and prepared dataset of historical gold prices, with an additional regressor—the US Dollar Index (USDX)—included to improve forecasting accuracy.

The **USDX** was chosen after feature analysis revealed it had the strongest long-term correlation with gold prices across the dataset spanning from **1979** to **2025**. Prophet's built-in support for external regressors allowed seamless integration, helping the model capture the influence of currency fluctuations on gold prices more effectively.

The training process was straightforward: Prophet was fitted on the dataset with **ds**, **y**, and the **usdx** regressor columns. After training, the model was evaluated using the **R^2 score**, which resulted in a strong accuracy of approximately 97%, confirming the model's reliability for one-year-ahead forecasting.

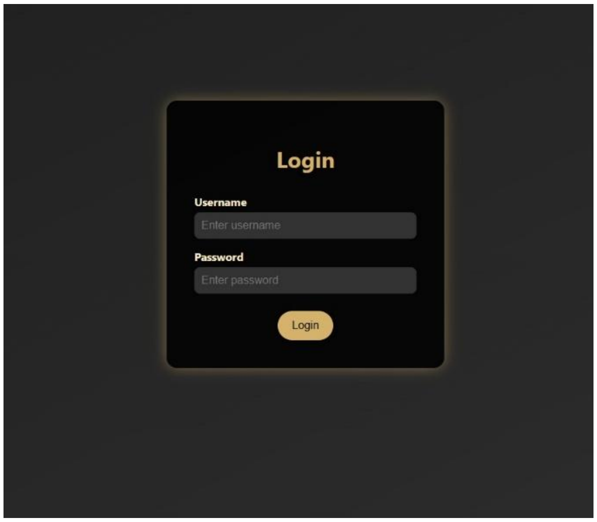
```
gold_model = Prophet()
gold_model.add_regressor("usdx")
```

```
<prophet.forecaster.Prophet at 0x24158d629d0>
```

Fitting the model

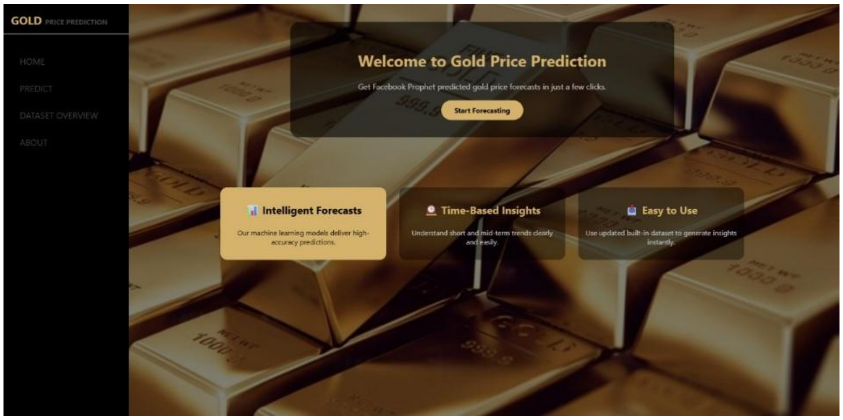
Web Application Functionality

- **Log in Page:**
  This serves as the log in interface for the user when they open the website.



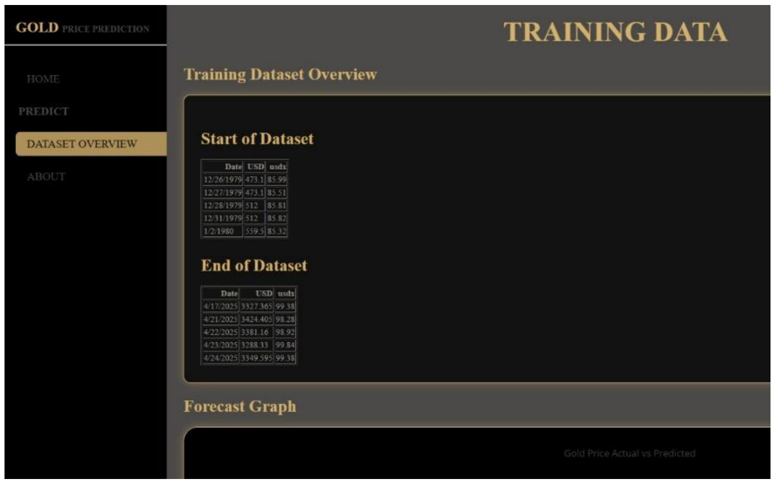
Log in page

- **Home Page:**
  This serves as the landing page of the application, offering users a brief overview of the project, its purpose, and what they can expect from the tool. It sets the context and introduces users to the functionality of the prediction system.

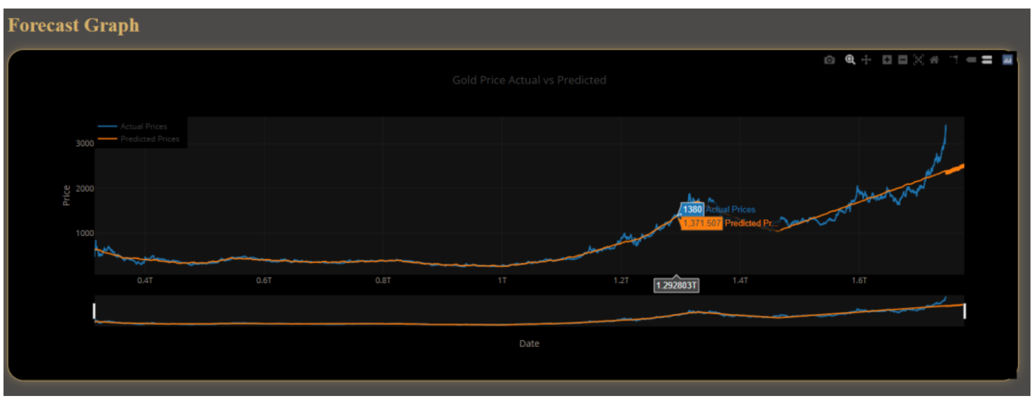


Homepage

- **Dataset Overview Page:**
  This page provides transparency into the model's training data by displaying its head and tail. Users can also explore an interactive graph comparing actual versus predicted prices for clearer insight.

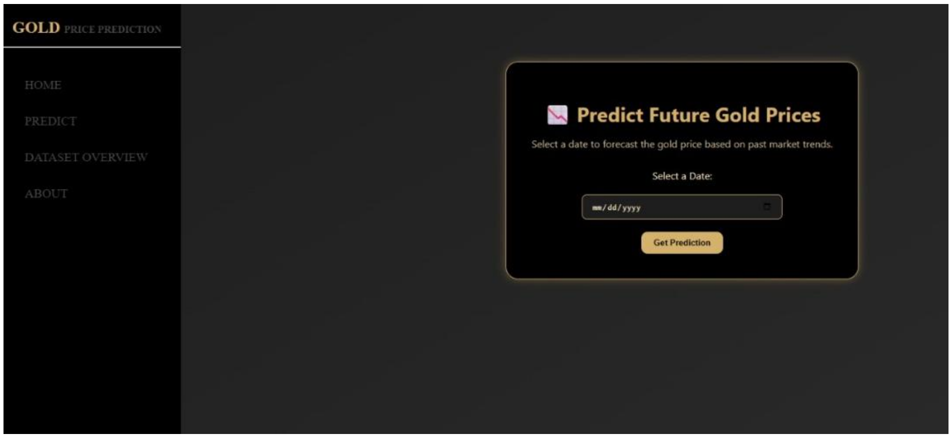
- 

Training overview



Actual prices vs predicted prices

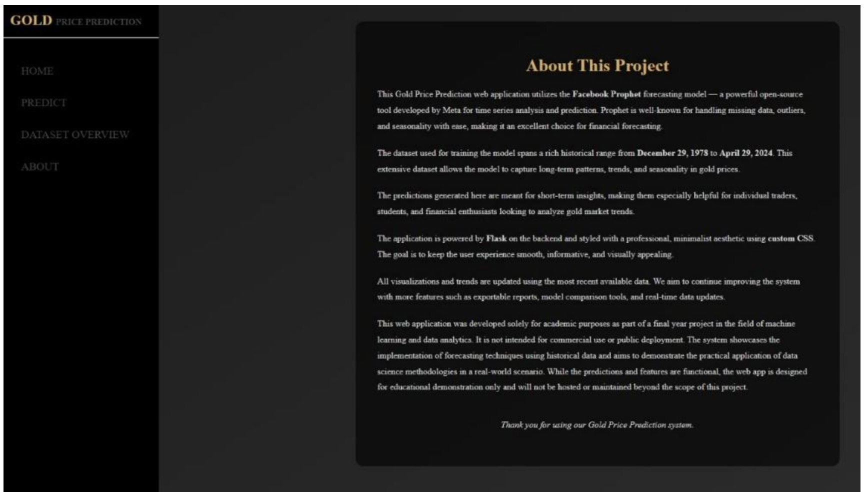
- **Prediction Page:**
  This is the core feature of the application. Users are prompted to input a specific date for which they want to predict the gold price. Upon submission, the Facebook Prophet model generates and displays the forecasted price per troy ounce. This page is designed to be straightforward and informative, ensuring a smooth user experience.



Prediction page

Prediction block

- **About Page:**
  The About section provides background information on the project, including a summary of the dataset, the choice of model (Facebook Prophet), and a high-level explanation of how the prediction is made. It helps users understand the methodology behind the tool and builds trust in the results.



About page

Web Interface Design with HTML and CSS

A clean and user-friendly web interface was built with HTML and CSS to display the gold price forecasts. The design focuses on clarity, featuring a homepage, a dedicated forecast chart, and a minimal, professional layout to make the predictions easily interpretable for all users.

```html
<div class="main-content">
  <div class="hero-section">
    <h1>Welcome to Gold Price Prediction</h1>
    <p>
      Get Facebook Prophet predicted gold price forecasts in just a few
      clicks.
    </p>
    <a href="/predict" class="cta-button">Start Forecasting</a>
  </div>

  <div class="features-section">
    <div class="feature-box">
      <h2>📊 Intelligent Forecasts</h2>
      <p>
        Our machine learning models deliver high-accuracy predictions.
      </p>
    </div>
    <div class="feature-box">
      <h2>🕰️ Time-Based Insights</h2>
      <p>Understand short and mid-term trends clearly and easily.</p>
    </div>
    <div class="feature-box">
      <h2>📥 Easy to Use</h2>
      <p>Use updated built-in dataset to generate insights instantly.</p>
    </div>
  </div>
```

Code snippet of homepage html

```css
.main-content {
    margin-top: -30px;
    margin-left: 280px; /* Default margin to accommodate the sidebar */
    flex: 1; /* Take up the remaining space */
    padding: 20px; /* Padding inside the main content */
    position: relative; /* Position relative for the background */
    transition: margin-left 0.3s ease; /* Smooth transition for margin */
}
```

Code snippet of css page

Result and Analysis

PREDICTING THE usdx FIRST TO USE AS A REGRESSOR

```
df_usd = df[["Date", "usdx"]].rename(columns={"Date": "ds", "usdx": "y"})
usd_model = Prophet()
usd_model.fit(df_usd)
future_usd = usd_model.make_future_dataframe(periods=360)
forecast_usd = usd_model.predict(future_usd)

usd_future = forecast_usd[["ds", "yhat"]].rename(columns={"yhat": "usdx"})
```

Code snippet of predicting the future usdx (regressor) first

Forecast Output Table:

Date	Actual Price	Predicted Price	Error
06-08-2017	1273.10	1207.934	-65.166
01-03-2022	1805.90	1888.782	+82.882
04-11-2023	2002.70	2101.414	+98.714
04-18-2024	2379.3	2251.259	-128.041
02-11-2025	2897.58	2367.51	-530

Actual Price vs Predicted Price Table

Conclusion

The project successfully developed a web-based gold price forecasting tool using Facebook Prophet. The model achieved a **97% R^2 score,** significantly aided by including the US Dollar Index. This demonstrates that a simple, resource-efficient model can yield highly accurate short-term predictions.

Limitations & Future Work

The project had limitations, including **computational constraints**, a limited feature set, and a basic **web app login**. Future improvements could involve adding more **features**, a **dynamic user system**, and **model comparison tools**.

In summary, the project provides a strong, practical foundation for an accessible gold price prediction system.

Bibliography

Masud, S.R., Imtiaz, A., Hossain, S.M. and Miah, M.S.U., 2025, March. A Comprehensive Approach to Gold Price Prediction Using Machine Learning and Time Series Models. In *2025 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)* (Vol. 3, pp. 1-6). IEEE. Available at: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10984927&isnumber=10984438 [Accessed 10 April 2025].

Manjula, K.A. and Karthikeyan, P., 2019, April. Gold price prediction using ensemble based machine learning techniques. In *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)* (pp. 1360-1364). IEEE. Available at: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8862557&isnumber=8862508 [Accessed 14 April 2025].

Landge, U., Phokmare, O., Borane, N. and Shelke, P., 2024, June. Gold Price Prediction using Random Forest Algorithm. In *2024 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)* (pp. 1287-1292). IEEE. Available at: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10575316&isnumber=10574894 [Accessed 2 April 2025].

Zainal, N.A. and Mustaffa, Z., 2015, August. A literature review on gold price predictive techniques. In *2015 4th International Conference on Software Engineering and Computer Systems (ICSECS)* (pp. 39-44). IEEE. Available at: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7333120&isnumber=7333073 [Accessed 27 March 2025].

Tripurana, N., Kar, B., Chakravarty, S., Paikaray, B. and Satpathy, S., 2022, May. Gold Price Prediction using Machine Learning Techniques. In *ACI@ ISIC* (pp. 274-281). [online] Available at: https://ceur-ws.org/Vol-3283/Paper108.pdf [Accessed 2 May 2025].