

# Building a Finance Data Analytics – Stock Price Prediction Model

By Anshika Goel, Sai Chaitanya Munagala, San Jose State University,  
Applied Data Science Department.  
03/06/2025

---

## Abstract

Stock price prediction plays a major role in business sectors to anticipate the market trends and prepare for better tomorrow. This project paper aims to present an end-to-end stock price prediction for the next 7 days using Snowflake and Apache Airflow. The pipeline extracts the stock data of past 180 days from yfinance API and stores them in a snowflake database, and leverages Machine learning capabilities to predict future stock pricing. Two DAGs are implemented for this project, one for data extraction and loading which we popularly called as ETL and other for forecasting and merging results using UNION ALL command. The final dataset integrates actual and predicted prices for complete comprehensive analysis. Although the project demonstrates the integration of cloud-based data warehousing, machine learning, and workflow orchestration to forecast data, the results represent likelihoods rather than guaranteed actual pricing.

---

## 1. Introduction

### A. Background

Predicting stock price is a critical process in finance and investing sector. Lot of the times investors, traders and also data analysts require reliable and proper prediction tools to forecast stocks prices. General traditional forecasting models typically required of some manual intervention at every stage like collecting, storing and predicting. Hence, they are inefficient, error-prone and not scalable. Now, with the advanced cloud computing technology and availability of fully automated data pipelines, financial forecasting is a streamlined process in a much efficient way.

### B. Problem Statement

The highly dynamic stock market requires a continuous data updates and forecasting. Manually retrieving stock prices and analysing one at a time is a real time-consuming and impractical for large scale financial systems especially stock industry where the data gets changed at a real rapid phase. The key challenges include:

1. *In-efficient data processing* – Stock Information needs to be extracted, transformed and stored at given real time.
2. *Scalability Issues* – Traditional go to approaches do not support real time updates.
3. *Machine Learning Integration* – Analysis and predicting part requires external ML tools.

To address these challenges, we propose a fully automated stock price forecasting system using Apache Airflow, Snowflake and yfinance API.

### C. Objectives

The objective of this project are as follows:

1. To develop a fully automated stock data extraction and storage using *Apache Airflow* and *Snowflake*
  2. To Apply machine learning models to forecast stock prices.
  3. To *Optimize performance issues*, scalability and reliability using Airflow scheduling, *SQL transactions* and *error handling mechanisms*.
  4. To merge actual and predicted prices into a final table for analysis purpose.
-

## 2. System Architecture

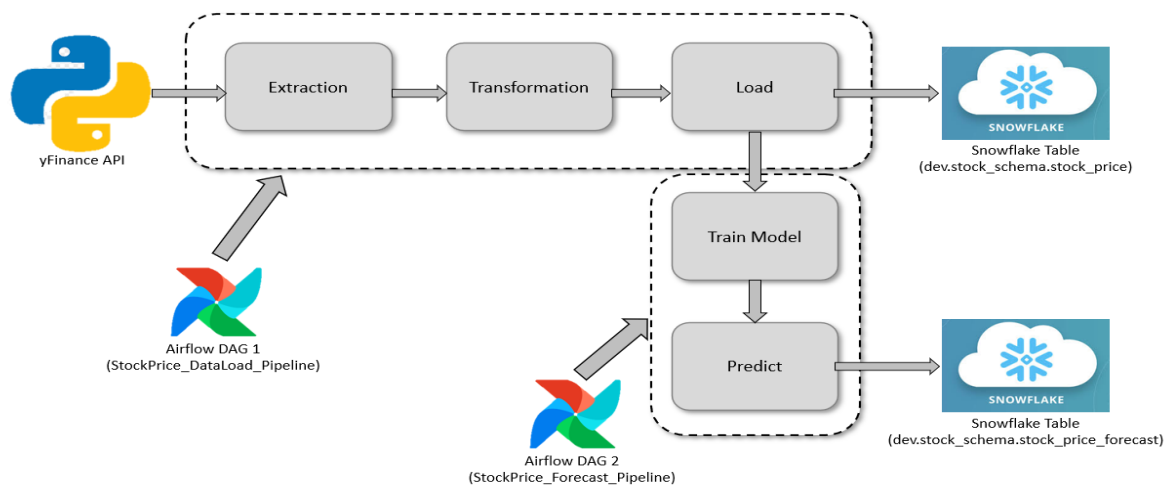
### A. Technology Stack

The project is developed using the listed technologies:

1. *yfinance API* – for fetching real-time and historical stock price data.
2. *Python* – for scripting and integrating API.
3. *Apache Airflow* – for Managing automated workflows to extract and forecast data.
4. *Snowflake* – for primary data warehouse.
5. *SQL* – for data storage, querying and machine learning forecasting within Snowflake.

### B. System Design

Overall Workflow diagram is constructed as follows:



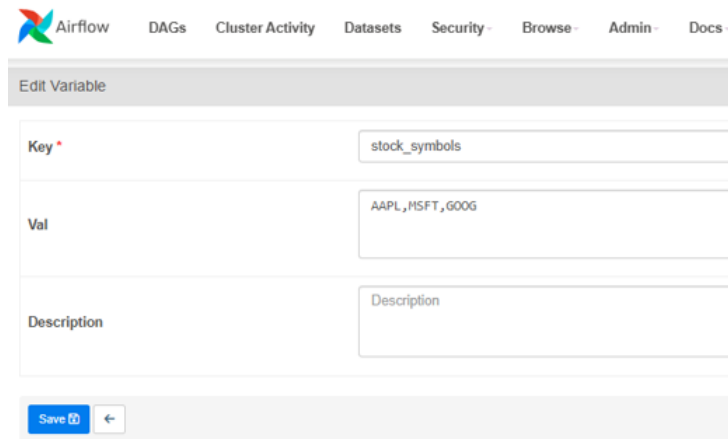
### C. Airflow Connections and Variables

To ensure flexibility and maintainability, Airflow connections and Airflow variables are used to manage configurations dynamically for just one real time. The snowflake connections are securely stored in Airflow under the tab snowflake\_conn identifier. This is a great practise which allows securing snowflake credentials and smooth connection between Airflow and Snowflake.

The screenshot shows the Airflow web interface with the 'Edit Connection' form for a Snowflake connection. The form includes the following fields:

- Connection Id: snowflake\_conn
- Connection Type: Snowflake
- Description: (empty)
- Schema: STOCK\_SCHEMA
- Login: GoelAnshika99
- Password: snowflake password
- Extra: 

```
{
  "account": "aanwwby-jc87555",
  "warehouse": "compute_wh",
  "database": "dev",
  "insecure_mode": false
}
```
- Account: aanwwby-jc87555
- Warehouse: compute\_wh
- Database: dev
- Region: snowflake hosted region
- Role: snowflake role



The image shows the 'Edit Variable' form in the Apache Airflow web interface. It has a header with navigation links: Airflow, DAGs, Cluster Activity, Datasets, Security, Browse, Admin, and Docs. The form contains three input fields: 'Key' with the value 'stock\_symbols', 'Val' with the value 'AAPL,MSFT,GOOG', and 'Description' with the value 'Description'. At the bottom, there is a 'Save' button and a back arrow.

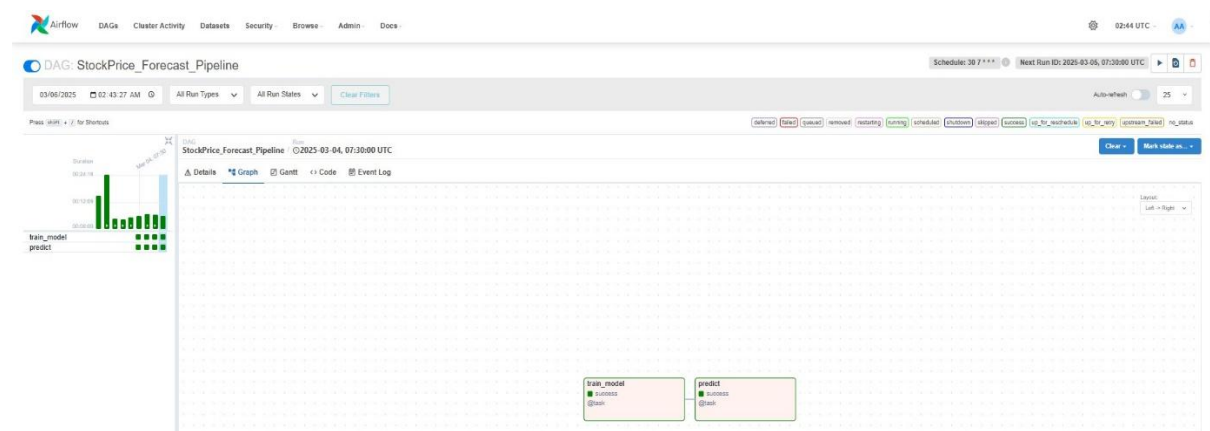
In Airflow variables, stock ticker symbols are stored as an Airflow Variable in the form of Key and Values. This option eliminates the developer to tweak the code every time for stock change. The UI built option In Apache Airflow is perfectly set for the developer user to change it anytime without altering DAG script.

Thus, The Core System Designs after proper connections and *DAG scripting* ideally consists of two primary workflows:

1. *ETL Pipeline* - Extracts stock data and process it before loading into snowflake.
2. *Forecasting Pipeline* – Uses Snowflake ML model to forecast next seven-day pricing.



**Figure: ETL DAG Pipeline with 3 tasks [Extract, Transform and Load]**



**Figure: Forecasting Pipeline with 2 tasks [Train and Predict]**

## D. GitHub Repository for Airflow DAG

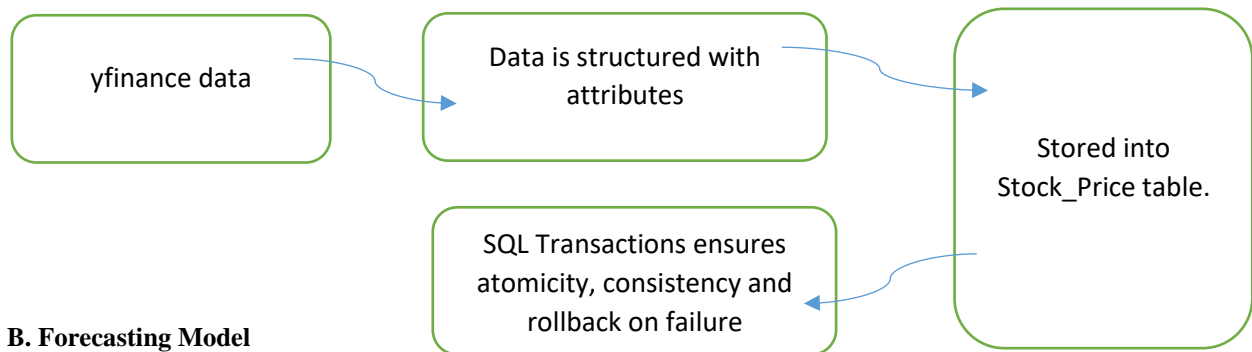
ETL DAG – [https://github.com/GoelAnshika99/Lab-1\\_Group-16/blob/main/ETL%20DAG.py](https://github.com/GoelAnshika99/Lab-1_Group-16/blob/main/ETL%20DAG.py)

Forecast DAG - [https://github.com/GoelAnshika99/Lab-1\\_Group-16/blob/main/Forecast%20DAG.py](https://github.com/GoelAnshika99/Lab-1_Group-16/blob/main/Forecast%20DAG.py)

### 3. Methodology

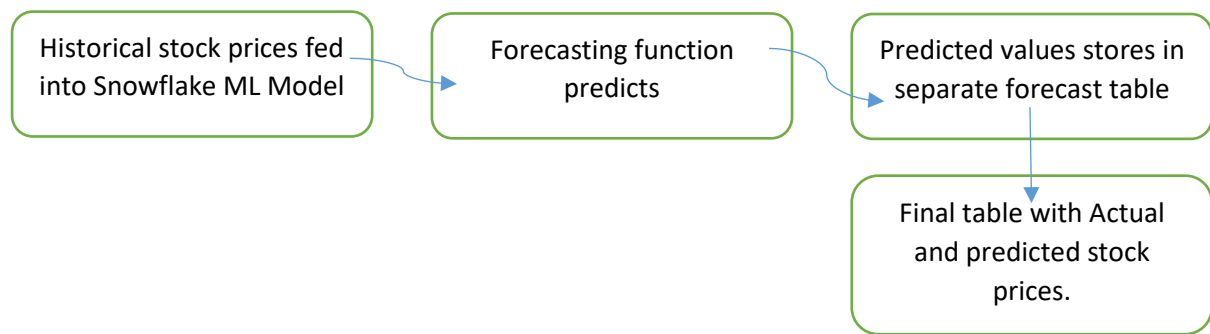
#### A. Data Extraction and Loading

The *ETL Pipeline* extracts stock data using *yfinance API* and it loads into structured table in Snowflake. The process typically follows,



#### B. Forecasting Model

The forecasting pipeline applies built-in *Machine Learning Model* to predict stock prices seamlessly as mentioned for the next seven days. The process typically follows,



### 4. Database Schema and Data Storage

#### A. Stock Price Table

This table contains *historical stock prices* fetched from yfinance.

Field Name	Data Type	Attributes	Constraints
Symbol	Varchar	Unique identifier	Not Null, Primary Key
Date	Date	Represents trade date	Not Null, Primary Key
Open	Float	Numeric, Financial Value	Nullable
High	Float	Numeric, Financial Value	Nullable
Low	Float	Numeric, Financial Value	Nullable
Close	Float	Numeric, Financial Value	Nullable
Volume	Float	Numeric, Financial Value	Nullable

#### B. Forecasted Stock Price Table

This table stores *predicted stock prices* for the next seven days.

Field Name	Data Type	Attributes	Constraints
Symbol	Varchar	Unique identifier	Not Null, Primary Key
Stock_ts	Date	Timestamp of record	Not Null, Primary Key

Open	Float	Numeric, Financial Value	Nullable
High	Float	Numeric, Financial Value	Nullable
Low	Float	Numeric, Financial Value	Nullable
Volume	Float	Numeric, Financial Value	Nullable
Actual_close	Float	Numeric, Financial Value	Nullable
Forecast_close	Float	Numeric, Financial Value	Nullable
Forecast_Lower_Bound	Float	Numeric, Financial Value	Nullable
Forecast_Upper_Bound	Float	Numeric, Financial Value	Nullable

## 5. Results and Discussion

### A. Airflow Execution Results

Both the *Airflow DAGs* are executed successfully, fetching the stock price data, loading them into the tables in snowflake and generating a seven-day forecast.

DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks	Actions	Links
StockPrice_DataLoad_Pipeline	etl	Running	30755	2025-03-05, 06:34:02	2025-03-05, 07:00:00	...	...	...
StockPrice_Forecast_Pipeline	etl	Running	30755	2025-03-05, 03:45:37	2025-03-05, 07:30:00	...	...	...

### B. Final Consolidated Table

The final table is consolidated with actual and predicted stock prices with the help of UNION ALL command and stores them into table, allowing for an easy visualization.

	SYMBOL	STOCK_TS	OPEN	HIGH	LOW	VOLUME	ACTUAL_CLOSE	FORECAST_CLOSE	FORECAST_CLOSE_LOWER_BOUND	FORECAST_CLOSE_UPPER_BOUND	STOCK_TS
1	AAPL	2025-03-14 00:00:00.000	null	null	null	null	null	235.462391296	219.10058471	252.298159801	2025-03-05 00:00:00.000
2	AAPL	2025-03-13 00:00:00.000	null	null	null	null	null	235.499433839	219.35207806	251.102741168	
3	AAPL	2025-03-12 00:00:00.000	null	null	null	null	null	235.588055643	220.958571223	250.906628111	
4	AAPL	2025-03-11 00:00:00.000	null	null	null	null	null	235.699519457	222.374771836	249.423927008	
5	AAPL	2025-03-10 00:00:00.000	null	null	null	null	null	235.514480183	223.053894009	247.855166956	
6	AAPL	2025-03-07 00:00:00.000	null	null	null	null	null	235.640107048	224.772307196	246.583581312	
7	AAPL	2025-03-06 00:00:00.000	null	null	null	null	null	235.70754486	226.185753589	244.886436125	
8	AAPL	2025-03-05 00:00:00.000	235.419998169	236.550003052	229.229995728	47200800	235.740005493	null	null	null	
9	AAPL	2025-03-04 00:00:00.000	237.710006714	240.070007324	234.679992676	53798100	235.929992676	null	null	null	
10	AAPL	2025-03-03 00:00:00.000	241.789993286	244.029998779	236.11000061	47184000	238.029998779	null	null	null	
11	AAPL	2025-02-28 00:00:00.000	236.949996948	242.089996338	230.199996948	56833400	241.839996338	null	null	null	
12	AAPL	2025-02-27 00:00:00.000	239.410003662	242.460006714	237.059997559	41153600	237.300003052	null	null	null	
13	AAPL	2025-02-26 00:00:00.000	244.330001831	244.979995728	239.130004883	44433600	240.36000061	null	null	null	
14	AAPL	2025-02-25 00:00:00.000	248	250	244.910003662	49013300	247.039993286	null	null	null	
15	AAPL	2025-02-24 00:00:00.000	244.929992676	248.86000061	244.419998169	51326400	247.100006104	null	null	null	
16	AAPL	2025-02-21 00:00:00.000	245.949996948	248.690002441	245.220001221	53197400	245.550003052	null	null	null	

10

11

12

13

SELECT \*

FROM STOCK\_PRICE\_FORECAST order by symbol, stock\_ts desc;

Results

Chart

	SYMBOL	STOCK_TS	OPEN	HIGH	LOW	VOLUME	ACTUAL_CLOSE	FORECAST_CLOSE	FORECAST_CLOSE_LOWER_BOUND	FORECAST_CLOSE_UPPER_BOUND
188	GOOG	2025-03-14 00:00:00.000						174.968399235	157.867532044	191.820965817
189	GOOG	2025-03-13 00:00:00.000						174.968399235	160.198326581	189.195365337
190	GOOG	2025-03-12 00:00:00.000						174.968399235	160.471359004	189.086204159
191	GOOG	2025-03-11 00:00:00.000						174.968399235	161.70813236	188.864707766
192	GOOG	2025-03-10 00:00:00.000						174.968399235	163.303845894	185.596715065
193	GOOG	2025-03-07 00:00:00.000						174.968399235	165.543823813	184.047237083
194	GOOG	2025-03-06 00:00:00.000						174.968399235	169.065082086	181.596601446
195	GOOG	2025-03-05 00:00:00.000	172.320007324	175.75	170.929992676	18836300	174.990005493			
196	GOOG	2025-03-04 00:00:00.000	167.940002441	175.164993286	167.539993286	30711400	172.61000061			
197	GOOG	2025-03-03 00:00:00.000	173.729995728	175	167.63999939	24122000	168.680003662			
198	GOOG	2025-02-28 00:00:00.000	170.300003052	172.5	168.38999939	30049800	172.220001221			
199	GOOG	2025-02-27 00:00:00.000	175.940002441	176.589996338	169.751998901	25930500	170.210006714			
200	GOOG	2025-02-26 00:00:00.000	176.945007324	178.080001831	173.589996338	23637200	174.699996948			
201	GOOG	2025-02-25 00:00:00.000	180.154998779	180.759994507	176.770004272	20832500	177.369995117			
202	GOOG	2025-02-24 00:00:00.000	183.800003052	185.089996338	180.880004883	18734200	181.190002441			
203	GOOG	2025-02-21 00:00:00.000	187.289993286	187.470001221	181.130004883	19520800	181.580001831			

Query Details

Query duration

1.2s

Rows

561

Query ID

01bad513-0004-5403-

Show more

SYMBOL

AAPL

187

GOOG

187

MSFT

187

STOCK\_TS

2024-06-14

2025-03-14

OPEN

#

**Note:** The Final table above is generated at a timestamp 05<sup>th</sup> March 2025, there by showing next 7 days results.

## 6. Conclusion

Thus, finally at the end of the project, A fully automated pipeline is developed and built to automate stock price prediction with the help of *Apache Airflow*, *Snowflake* and *yfinance API*. This implementation eliminates manual intervention at every stage of its life cycle and ensures that stock prices are fetched, stored, processed, and forecasted seamlessly on a scheduled basis.

With Airflow orchestrating the workflows, the ETL pipeline reliably pulls stock price data and stores it in Snowflake, while the forecasting pipeline applies machine learning models to generate future stock price predictions. The use of Airflow connections and variables makes the pipeline flexible and easily configurable, allowing updates without modifying the DAG scripting. Therefore, the final output, which combines actual and predicted stock prices, provides valuable insights for financial analysis.

## 7. References

- [1] Snowflake Documentation: <https://docs.snowflake.com/>
- [2] Apache Airflow Documentation: <https://airflow.apache.org/docs/>
- [3] yFinance API Documentation: <https://pypi.org/project/yfinance/>
- [4] IEEE Paper Formatting Guidelines: <https://www.ieee.org/conferences/publishing/templates.html>