**DATANIFTY: UNVEILING STOCK TRENDS**

**SPECIAL TOPICS IN BUSINESS INTELLIGENCE**

**(DATA INTEGRATION)**

**SUBMITTED BY**

**SAISREEJA YALAVARTHI**

**(yalavarthi.sa@northeastern.edu)**

**SRIKAR RAMBHATLA**

**(rambhatla.s@northeastern.edu)**

**Problem Statement:** The financial market, particularly in the domain of stock trading, is witnessing significant growth and evolution, with the Nifty stock index being a prominent player in the Indian stock market. With the advent of technology and increased access to financial data, there's a flourishing interest in leveraging data analytics and machine learning techniques for understanding and predicting stock price movements. The availability of granular data at 5-minute intervals from January 2015 to February 2022 provides a rich resource for conducting comprehensive analysis.

**Business Flow Proposed:** This project aims to construct a robust data pipeline for Nifty stock price analysis, leveraging both traditional OHLCV (Open, High, Low, Close, and Volume) data along with a suite of 55 technical indicators derived from this data. The data pipeline will facilitate the integration of diverse data sources, ensuring seamless access and processing of historical stock price information. By incorporating advanced analytics techniques, the pipeline will enable the extraction of actionable insights to inform trading strategies and investment decisions.

**Analysis Dimensions:**

* Trend Analysis: We can identify long-term, intermediate, and short-term trends in Nifty stock prices using moving averages, trendlines, and trend reversal indicators.
* Volatility Analysis: Also measure the volatility of Nifty stock prices using indicators such as Bollinger Bands, Average True Range (ATR), and historical volatility.
* Momentum Analysis: Can assess the momentum of Nifty stock prices using indicators like the Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), and Stochastic Oscillator.
* Volume Analysis: We can analyze trading volume patterns to identify periods of accumulation or distribution, using volume-based indicators such as On-Balance Volume (OBV) and Chaikin Money Flow.
* Pattern Recognition: We can also detect chart patterns such as triangles, flags, and head and shoulders patterns using pattern recognition algorithms.
* Correlation Analysis: Correlations can be explored between Nifty stock prices and various macroeconomic indicators, sectoral indices, and global market trends.

**Strategic Insights:** The analysis would be helpful for investors to understand the KPIs that are influencing the market trends. By establishing a comprehensive data pipeline for Nifty stock price analysis, this project aims to empower investors, traders, and financial institutions with actionable insights derived from robust data analytics techniques, ultimately enhancing decision-making processes, and maximizing returns on investment in the dynamic Indian stock market landscape. Also, we can identify optimal entry and exit points for trading positions based on trend, momentum, and volatility indicators. Market anomalies and abnormal trading patterns can be monitored for potential arbitrage opportunities or risk mitigation strategies. It will also help to adapt trading strategies accordingly and mitigate risk exposure.

**Data Sources:**

* https://www.kaggle.com/datasets/debashis74017/stock-market-data-nifty-100-stocks-5-min-data?select=BHARTIARTL\_with\_indicators\_.csv
* https://www.kaggle.com/datasets/setseries/nifty50-stocks-dataset20102021/data

**ETL CLOUD ARCHITECTURE**  
A diagram of data pipeline

Description automatically generated

The pipeline can be broadly divided into 3 layers namely Landing Layer, Staging Layer, and Production Layer. We wanted to analyse the trends over the timeline of years, quarters and month.

Before creating the pipeline, we analysed our dataset. We had 50 CSV individual files separately for each stock over 7 years, we used PYTHON to do the initial pre-processing steps, by merging the data into 3 files: Yearly, Quarterly, and Monthly.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**LANDING LAYER:**

Initially, we created S3 buckets.

One for Source: ***source-stocks*** & One for Target: ***target-stocks***.

After creating S3 buckets, the CSV files were uploaded to specific folders within the source bucket. Following this, to perform ETL and query the transformed data for analysis, we created source and target AWS Glue Databases using Amazon Athena. Then the data was loaded into the respective database with the help of AWS Crawlers. Crawlers automatically detect the schema of the files and create tables to store data in the database. The following below screenshots show the above-described process:

A screenshot of a computer

Description automatically generated

**Source and Target Buckets Creation**

A screenshot of a computer

Description automatically generated

**Uploaded the csv files of our data into the respective folders in the source S3 bucket**

**A screenshot of a computer

Description automatically generated**

We create a lambda function to send an email whenever a new file is added in the source bucket:  
A screenshot of a computer

Description automatically generated  
**Glimpse of Lambda Function**

A screenshot of a computer

Description automatically generated  
**Successful Log Generation**

A screenshot of a computer

Description automatically generated

**Source and Target Databases Creation in AWS Glue**

**A screenshot of a computer

Description automatically generated**

A screenshot of a computer

Description automatically generated  
**Crawlers’ creation to load data into the source Database**

**Preview of the data in Athena**  
A screenshot of a computer

Description automatically generated

**Monthly Table Preview**

A screenshot of a computer

Description automatically generated  
**Quarterly Table Preview**

A screenshot of a computer

Description automatically generated  
**Yearly Table Preview**

**STAGING LAYER:**

We create a target database in AWS Glue, for the data to be stored after ETL jobs completion.

Now we create ETL jobs in GLUE as follows:

A screenshot of a computer

Description automatically generated

**Yearly Table ETL Job Flow**

A screenshot of a computer

Description automatically generated

**Monthly Table ETL Job Flow**

A screenshot of a computer

Description automatically generated  
**Quarterly Table ETL Job Flow**

A screenshot of a computer

Description automatically generated  
**Glimpse of ETL Jobs**

After running the ETL jobs successfully, we now run the Crawlers to migrate the data into our target database. We created 3 different crawlers for our data to be migrated. The jobs ran successfully.

A screenshot of a computer

Description automatically generated

**PRODUCTION LAYER:**

Once the crawlers run successfully, we will be now able to query our tables in Athena. It is shown as below: The data has been successfully propagated.  
A screenshot of a computer

Description automatically generated

**Yearly Table Preview**

A screenshot of a computer

Description automatically generated  
**Monthly Table Preview**

A screenshot of a computer

Description automatically generated  
**Quarterly Table Preview**

**Few Analytical Queries :**

**A screenshot of a computer

Description automatically generated**

The above query focused on retrieving the `open`, `close`, and `volume` values for the earliest profitable trading day for each company. This helps to provide insights into profitable trading days, trading volumes, and performance comparisons across different companies.

A screenshot of a computer

Description automatically generated

This query identifies the top 10 stocks with the highest momentum using the 10-day momentum indicator (MOM10).

A screenshot of a computer

Description automatically generated

This query identifies stocks that are trading near the upper Bollinger Band (within 1% of the upper band), which may indicate overbought conditions.

**DASHBOARD:**