Team7 DS598 Project

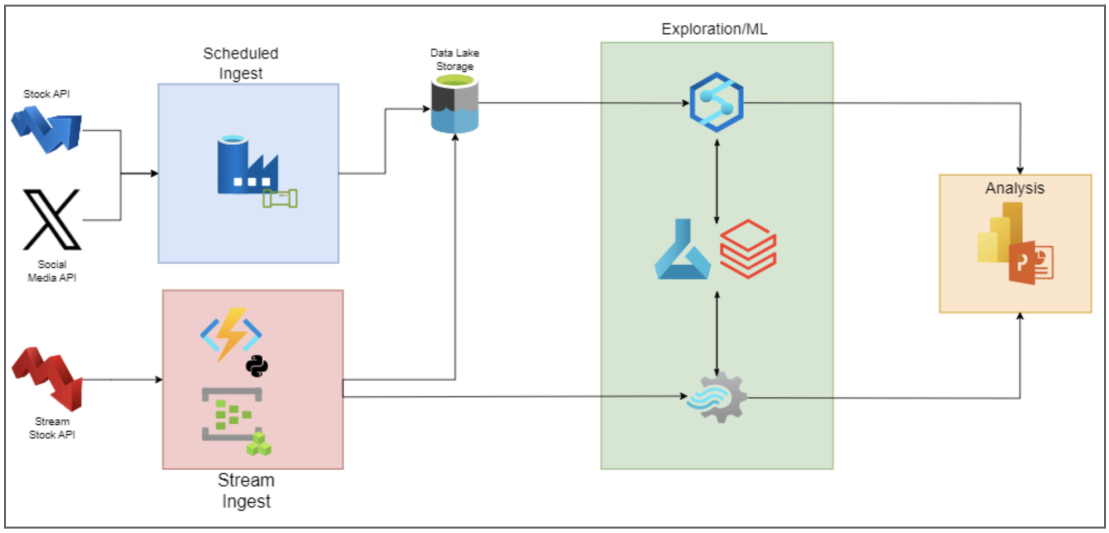
Zach Pao, Scoot Lyu, Caslow Chien

Overview & Introduction

We developed a pipeline using Azure tools (Data Factory, Event Hubs, Functions, Storage, Databricks, and Power BI) to predict the popularity of "AI" by analyzing data from Reddit, stock trends, and Google Trends. The pipeline cleans and processes data, applies an LSTM-based model for forecasting, and visualizes insights in Power BI. This project helps stakeholders understand the connection between stock performance, public sentiment, and AI news, offering actionable predictions and trend analysis.

The rapid growth of AI has reshaped global markets, yet predicting its sustained impact remains challenging. Stakeholders often struggle to evaluate AI's market influence in real-time, leading to inefficiencies. This pipeline addresses that gap by analyzing data from the Stock API and Reddit API over three years, supplemented by real-time data. It forecasts trends and provides historical analysis to deliver actionable insights, enabling better decision-making.

Methodology



In our project's architecture, we streamlined the data processing pipeline by bypassing Azure Stream Analytics and directly integrating our data into Power BI for visualization and analysis. This decision was guided by our need for a more straightforward workflow and the capability of Power BI to handle both batch and real-time data efficiently. By feeding cleaned and processed data directly from Azure Databricks into Power BI, we were able to leverage its dynamic data modeling and visualization tools immediately. This approach not only reduced the complexity of our data pipeline by eliminating an intermediary processing layer but also expedited the

availability of data insights through Power BI's robust analytical dashboards, enhancing our ability to quickly visualize trends and derive actionable intelligence from the combined stock and Reddit data streams.

Datasets

Data Ingestion

In the data ingestion phase of our project, we employed a dual-source approach, leveraging the Alpha Vantage API and Reddit data to analyze AI trends. For stock data, we integrated the Alpha Vantage API within our Azure Data Factory pipeline to systematically retrieve daily stock information starting from the year 2021. This integration ensured a reliable and continuous inflow of stock performance metrics, which are critical for correlating financial trends with public sentiment derived from social media. Our focus was particularly on key tech companies such as Microsoft (MSFT), AMD (AMD), Alphabet (GOOGL), Meta Platforms (META), and NVIDIA (NVDA), which are at the forefront of AI technology development. This selection allowed us to closely monitor and analyze how advancements in AI are influencing their stock performance and market behavior. Initially, our project aimed to utilize Twitter data; however, due to significant changes in Twitter's API access policies at the commencement of our project, we redirected our focus to Reddit, sourcing from subreddits such as r/stocks and the misspelled r/artificialinteligence. Each day, the top 500 posts from these subreddits were collected to serve as our historical dataset.

The data collection from Reddit presented unique challenges, notably the platform's rate limit of fetching only 50 posts per minute. This constraint complicated our efforts to amass large volumes of data swiftly. To manage this, we developed an automated system using Azure Event Hubs and Azure Functions, which facilitated the continuous and automated ingestion of streaming data. This setup not only avoided the limitations imposed by Reddit's API but also enhanced the robustness of our data collection framework, allowing for the real-time analysis of user interactions and sentiment shifts. The transition from Twitter to Reddit required adjustments in our data ingestion strategy but ultimately enriched our dataset by tapping into a broader and more diverse discourse surrounding AI trends. This comprehensive approach to data integration provided a robust foundation for understanding the evolving landscape of AI as these major technology firms continue to expand their AI capabilities.

Data Cleaning

For the data cleaning process, we utilized Databricks to access and preprocess the datasets retrieved from the Stock API and Reddit API, which were stored in our team's container. These datasets were combined into two unified datasets for analysis. To ensure the data was well-structured and ready for accurate insights, we implemented the following steps:

- **Handling Missing Data**: We thoroughly examined both datasets for any missing or incomplete entries. Any rows or columns with missing values were either filled using appropriate methods (e.g., mean or mode imputation) or removed, depending on their significance to the analysis. This step ensured the integrity and reliability of our data.
- **Timestamp Normalization**: To standardize the datasets, we normalized all timestamps to a consistent format: “YYYY-MM-DD HH:MM:SS” This ensured uniformity across the data, enabling seamless sorting, filtering, and time-based operations.
- **Weekly Categorization**: To facilitate temporal analysis, we created a new column named “week_range”. This column categorizes the data into weekly intervals, grouping entries into seven-day periods (Monday to Sunday). For instance, data entries with timestamps from "2021-01-04" to "2021-01-10" were categorized under the week range "2021-01-04 -- 2021-01-10." This grouping allowed for easier trend analysis over time.

Model

Feature Selection

We aggregate data using the average value for each week (Monday to Sunday).

- Diff_percent: Percentage change in stock's closing value compared to the previous row.

$$\text{Diff\_percent} = \frac{\text{Close}_{\text{current}} - \text{Close}_{\text{previous}}}{\text{Close}_{\text{previous}}} \times 100$$

- Volume: Average stock trading volume for the week.
- AI_previous_x: AI trend index from x week ago (x from 1 to 3)

Why “AI_previous”?
They allow models to explicitly capture the influence of historical AI trends and patterns on the current target value.

Structure

Overview:

➢ LSTM followed by fully connected layers.

- LSTM
  - Processes sequential data by learning patterns over time, capturing both short-term and long-term dependencies.
  - Applies a Dropout to randomly disables neurons during training to prevent overfitting and improve generalization.
- Fully Connected layer
  - Applies a ReLU activation function, introducing a non-linear transformation that helps the model learn more complex patterns in the data.

- Normalizes the data, stabilizing the learning process and allowing the model to train more effectively.
- Reduces the size of the data and generates the final prediction.

Why LSTM Works:

- Sequential Memory: LSTMs are designed to capture long-term dependencies in sequential data, making them ideal for time-dependent trends.
- Non-Linear Relationships: LSTMs model complex, non-linear relationships, which is crucial as trends like the popularity of "AI" are often not linear.

Additional Features:

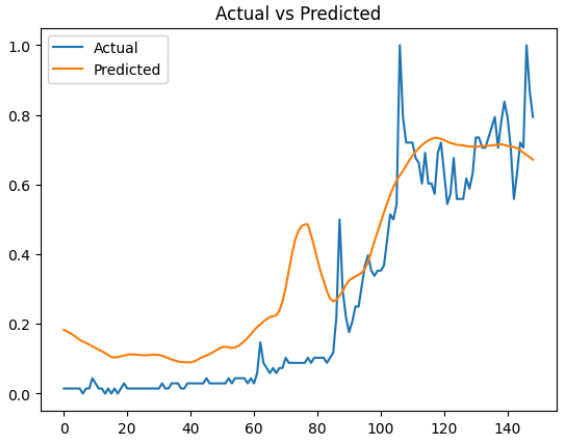
- Optimizer: Adam adapts the learning rate based on gradients, optimizing efficiently.
- Early Stopping: Stops training if validation performance doesn't improve after a set number of epochs, preventing overfitting.
- Learning Rate Scheduler: `ReduceLROnPlateau` to lower the learning rate when validation loss plateaus, aiding in finer model tuning.

Params:

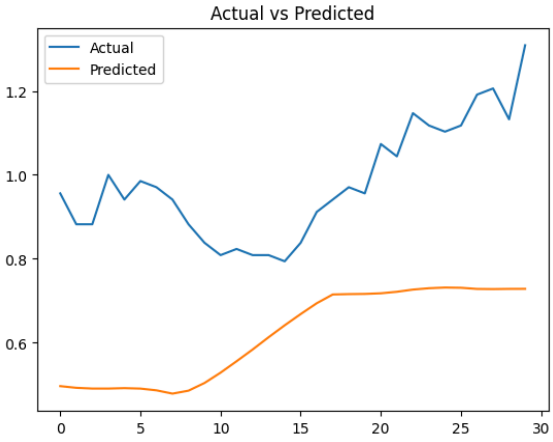
```
window_size = 10
batch_size = 2
epochs = 100
early_stopping_patience = 20
learning_rate = 0.0001
dropout = 0.4
```

Results

Actual vs Predicted - Training Data



Actual vs Predicted - Testing Data



The Training Data graph (on the left) clearly indicates that the predicted trend consistently leads the actual trend.

The Testing Data graph (right) shows the predicted and actual trends for the term "AI" from index 0 to 30. Between indices 5 and 15, the predicted model rises, forecasting an increase in trendiness for "AI," while the actual trend remains flat. From indices 15 to 30, the actual trend sees a sharp rise, matching the earlier prediction. This indicates that the model successfully predicted the upward trend before it happened, proving its usefulness in forecasting.

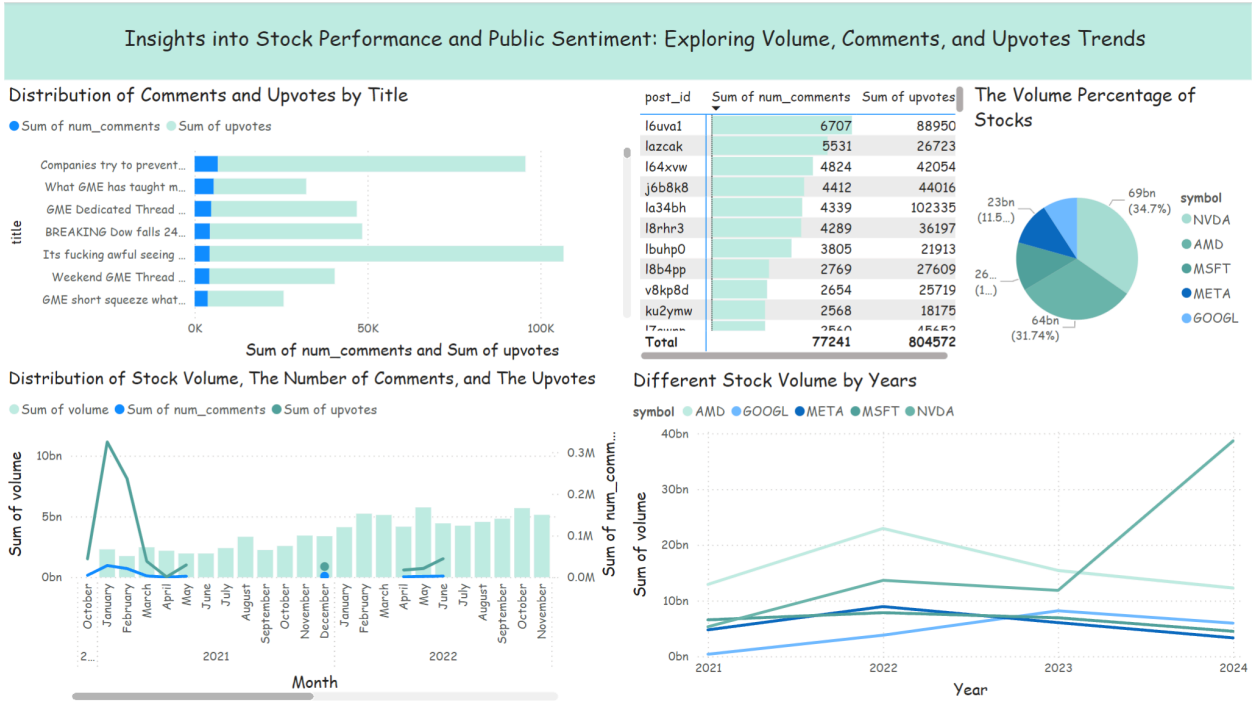
Evaluation

The evaluation results suggest that the model's performance is suboptimal, with a MAE of 0.36 indicating moderate prediction errors. While improvements are needed, the predictions are still intuitively useful.

Insights

Our model demonstrates strong predictive power in forecasting trends, as evidenced by its ability to predict the upward trend of "AI" before it occurred. The predicted trend consistently leads the actual trend, especially during periods where the model anticipates future increases in popularity. This makes the model highly useful for applications such as trend forecasting, market analysis, and strategic planning, where anticipating future movements ahead of time can provide a competitive advantage.

Analysis



The visualization shows that Reddit comments and upvotes are closely tied to stock performance. Newsworthy or controversial posts, like “Companies try to prevent…” and “Weekend GME Thread,” drive high engagement, reflecting and amplifying public interest in specific stocks.

The pie chart shows that NVIDIA (NVDA) holds the largest share of stock trading volume at 34.7%, followed by Microsoft (MSFT) at 31.7%, highlighting the prominence of stocks tied to AI and other technologies. The time-series chart reveals peaks in Reddit activity and trading volumes in early 2021, likely driven by events like the GameStop (GME) short squeeze, demonstrating the significant impact of market-moving events on public sentiment and trading behavior, with steady growth observed through 2023.

Recommendation / Future Work

1. Develop an automated scheduling system to periodically retrain the model whenever sufficient new data becomes available.
2. Prepare for potential scalability issues by implementing Synapse in case the dataset grows significantly larger in the future.
3. Expanding the dataset with additional social media data could greatly enhance the model's performance. However, as obtaining such data is often costly, consider exploring alternative social platforms.
4. Enrich the dataset with more diverse and complex data to improve the model's predictive accuracy. For example, include updates and news from leading AI companies.

Conclusion

This project used Azure tools to build a pipeline for predicting AI popularity by analyzing data from Reddit, stock trends, and Google Trends. The LSTM-based model and Power BI visualizations provide actionable insights, linking public sentiment, market performance, and AI trends. Future work will enhance scalability, dataset richness, and model retraining.