Interactive Analytics of Market Sentiment using LLM-Labeled Data

Team 176 – Michael Pucci, Matthew Cheung, Yiming Feng, Xin Lin

1. Introduction

Financial sentiment is crucial to investor behavior, sector narratives, and short-term market movements. Traditional tools focus primarily on structured financial news sites or static sentiment indices, while missing out on fast-moving online communities. Major media platforms such as Reddit's r/stocks and r/wallstreetbets can amplify optimism, fear, or speculation ahead of formal news coverage. Because of this, there is a clear need for systems that can capture sentiment across both financial news and social media in a unified and easy-to-understand format.

Our project addresses this need by building an interactive system that tracks financial sentiment across news and Reddit discussions, linking each sentiment score to specific tickers and sectors. The goal is to identify shifts in market mood that conventional dashboards or static indices often miss. The system integrates large language model-based sentiment labeling, a neural network classifier trained on financial news, and an interactive visualization platform.

The objective of our work is to create a practical and dynamic tool that helps analysts, researchers, and investors understand how market sentiment evolves across multiple sources of information. The system is designed to be reliable, adaptable, and easy to extend in future work.

2. Problem Definition

We aim to build a system that reads text about the stock market, figures out which companies or sectors the text is talking about, decides whether the text sounds positive, negative, or neutral, and then shows how this sentiment changes over time. The purpose is to help people easily see whether the market feels optimistic or pessimistic based on both news articles and social media discussions.

Formally, given a stream of financial text news headlines and Reddit posts, the system must complete three core tasks:

(i) Detect ticker and sector mentions
(ii) Assign sentiment score
(iii) Visualize aggregate sentiment trends over time and across tickers or sectors

The system operates in two stages:

1. **Model Training Phase:** A corpus of financial news headlines is labeled using structured prompts in Google Gemini. These labels form a high-quality training dataset for a neural network sentiment model that is specifically tailored to financial language.

2. **Deployment and Visualization Phase:** The trained model is applied to Reddit data to produce sentiment labels. Predictions are written to a PostgreSQL database, aggregated into time series and sector-level summaries, and visualized through an interactive dashboard that highlights patterns and shifts in market sentiment.

This structure allows the system to learn from reliable news data while capturing rapid changes in mood found in social media, providing a combined view of sentiment across multiple information sources.

3. Literature Survey

3.1 Financial Sentiment & Market Impact

Foundational research establishes the predictive power of textual sentiment in financial market, for example, Tetlock (2007) demonstrates that negative media tone predicts short-term stock returns and trading volume, providing empirical justification for using news sentiment as a market signal, though his work focused exclusively on Wall Street Journal articles. Extending this, Tetlock, Saar-Tsechansky, and Macskassy (2008) show that firm-specific news helps explain earnings and returns through textual tone analysis, supporting the importance of entity-level ticker linking in sentiment systems. Loughran and McDonald (2011) address the inadequacy of general-purpose sentiment dictionaries for financial texts by developing finance-specific lexicons tailored to 10-K filings. Bollen, Mao, and Zeng (2011) broaden the sentiment landscape by demonstrating that aggregate Twitter mood correlates with stock market movements, motivating the inclusion of retail investor sentiment from platforms like Reddit. Shiller (2017) provides theoretical grounding through "narrative economics," arguing that contagious narratives shape economic outcomes—a concept that frames the importance of visual storytelling around sentiment flows and market catalysts.

3.2 NLP Models, Event/Ticker Linking

The evolution from dictionary methods to neural architectures marks significant advances in financial sentiment analysis. Devlin et al. (2019) introduce BERT's bidirectional transformer pretraining, which Araci (2019) adapts to financial corpora with FinBERT, creating a strong domain-specific baseline that outperforms general sentiment models on finance tasks despite computational intensity. Park, Lee, and Cho (2021) address dictionary limitations through Senti-DD, a context-aware lexicon pairing direction-dependent words with target terms to reduce classification errors, though its reliance on simple syntactic cues limits generalization to complex social media text.

Ong et al. (2023) propose FinXABSA, moving beyond document-level sentiment to aspect-based analysis that assigns sentiment to specific dimensions like performance and risk, demonstrating improved interpretability and predictive power through statistical linkage to market movements. Sun et al. (2025) present EnhancedFinSentiBERT, bridging dictionary methods and transformers through domain adaptation and achieving robust improvements over baseline approaches, albeit with substantial computational requirements. Kirtac and Germano (2025) synthesize these approaches, advocating for hybrid architectures that integrate lexicon-derived features with LLM embeddings, finding that such combinations often outperform either method alone. Supporting event extraction capabilities, Ding et al. (2015) guide structured event extraction from news for forecasting, while Liu et al. (2020) cast event extraction as machine reading comprehension to improve precision in identifying market-relevant information.

3.3 Visualization for Temporal/Narrative Insight

Effective visualization transforms sentiment data into actionable insights. Havre, Hetzler, and Nowell (2002) establish the ThemeRiver streamgraph metaphor for visualizing evolving themes over time, providing the conceptual basis for temporal sentiment flow visualization from global markets down to individual tickers. Heer and Shneiderman (2012) formalize design principles for interactive visual analytics—overview first, zoom and filter, details on demand—that inform the architecture of linked sentiment heatmaps, timelines, and detail panels essential for exploring multi-scale market narratives.

4. Proposed Method

Our system integrates two data sources: structured financial news headlines from Webz.io, a Financial News Dataset Repository used for model training and unstructured Reddit discussions used for live deployment and visualization. The core idea is to train a sentiment classifier on high-quality, LLM-labeled financial news, then apply this classifier to fast-moving social media text in order to track sentiment at the ticker and sector levels. The full pipeline operates within a cloud environment and supports end-to-end functionality from data ingestion to visualization.

4.1 Intuition

Current financial sentiment systems rely on static dictionaries or transformer-based model that are computationally substantial. Dictionary methods often struggle with context, sarcasm, and rapidly evolving market language. The large transformer models like FinBert or domain-adapted BERT variants offer better accuracy but are often too computationally expensive for continuous usage and inference on real-time social media streams.

Our method adopts a hybrid strategy, pulling from the strengths of both approaches. We use a large language model to create high-quality sentiment labels for financial news headlines, then train a feed forward neural network on this dataset. In doing so, this creates a model more efficient than full transformer architectures yet benefits from the quality of LLM-generated labels. This structure allows fast inference on unstructured Reddit text without sacrificing domain-specific accuracy. The pipeline provides an efficient and scalable solution that captures both stable news sentiment and rapidly changing Reddit sentiment.

By storing predictions in a PostgreSQL database and connecting them to an interactive dashboard, we create a system that gives users control over how they view and interpret market sentiment trends. This design emphasizes transparency, interpretability, and ease of use, while still incorporating modern NLP and cloud infrastructure techniques.

4.2 Detailed Description of the Approach

First, the data was sourced from multiple outlets. The Financial News Dataset served as our Training Data. For this, we used a collection of more than two hundred thousand financial news headlines sourced from Webz.io, which provides structured metadata, timestamps, and clean text suitable for model training. The remaining data was gathered using Reddit's API. This involved scraping unstructured Reddit posts and comments continuously from finance-related communities including r/stocks, r/investing, and r/wallstreetbets. These posts contain substantial retail sentiment signals, ticker mentions, slang, humor, and other features that differ significantly from formal news text. The Reddit Dataset was use as our Deployment Data for our visualizations.

Next, to create a labeled dataset for supervised learning, we generate sentiment labels for the news headlines using Google Gemini. A structured prompt template classifies each headline into positive, neutral, or negative categories. Batch prompting is used to efficiently label the entire dataset.

This approach allows us to avoid manual annotation while still generating a high-quality dataset. Because the labels are produced by a consistent LLM prompting framework, they maintain uniformity across the entire compilation of texts. This provides a stable training signal for the neural network.

After label generation, we were able to proceed with training a feed forward neural network to classify sentiment using the Gemini-labeled headline dataset. The architecture was selected to provide a balance between accuracy and inference speed, which is critical for our deployment.

The model training process includes:

1. Cleaning and normalizing headline text.
2. Converting text to numerical embeddings.
3. Training the model with cross-validation.
4. Exporting the final model to the GCP environment.

This classifier learns sentiment patterns that are specific to financial language, such as earnings signals, mergers, analyst ratings, and risk terminology.

Following the Neural Network Sentiment Model, was the development of the Reddit Ingestion Pipeline. We implemented a Python-based ingestion pipeline using PRAW to collect Reddit posts in near real time. The pipeline retrieves relevant fields such as post text, timestamp, author, and subreddit name. All data flows directly into a PostgreSQL database hosted in our GCP environment.

A finance-aware text cleaning routine is applied that retains ticker symbols, percentages, dollar amounts, and emojis. These elements often carry sentiment information. For example, “AAPL to the moon” contains emojis and slang that a standard cleaning process might remove.

Ticker extraction is performed using pattern matching rules adapted for typical Reddit formats such as $TSLA, TSLA, and tickers embedded in text blocks.

Each Reddit post is passed through the trained sentiment model. The output includes:

1. A continuous sentiment score
2. A categorical sentiment label
3. Extracted tickers
4. Timestamp and metadata

All predictions were written to the PostgreSQL database. Aggregation queries compute daily averages, rolling ninety-day statistics, and sector-level summaries based on each company’s industry classification. These aggregated results serve as the data source for interactive visualizations.

The final component of the system is an interactive financial sentiment dashboard built with Dash and Plotly. The dashboard connects directly to the PostgreSQL database running in our GCP environment, allowing all visual elements to update dynamically based on user input. The visualization layer is designed to give users flexibility in how they explore sentiment across different companies, industries, and time periods.

The interface begins with a settings panel that lets users choose between two data sources: Reddit-only sentiment or the merged dataset that combines Reddit sentiment with financial news sentiment. Users may also select whether they want to analyze market behavior at the ticker level or the sector level. These

selections determine which PostgreSQL tables the dashboard queries and how the returned data is aggregated and displayed.

The main visualization consists of two tightly integrated components:

1. **Sentiment Time-Series Panel:** This panel allows users to select one or more tickers (or sectors) and view daily sentiment scores together with rolling ninety-day statistics. The system plots positive, neutral, and negative sentiment patterns using color-coded traces. Users can also optionally overlay the corresponding stock price for the first selected ticker, allowing comparison between sentiment shifts and price movements.

2. **Industry Heatmap Panel:** In addition to time-series views, the dashboard provides an industry-level heatmap showing aggregated sentiment across sectors and time. Users may filter by sentiment type (positive, neutral, or negative) and choose any date range to generate a new view. The heatmap uses a diverging color scale to highlight variations in sentiment intensity across industries, making it easy to identify which parts of the market are driving optimism or pessimism on any given day.

All dashboard elements rely on SQLAlchemy queries embedded in the callback functions. These callbacks ensure that only the necessary rows are retrieved from the database, improving responsiveness even when working with large datasets.

The design of the visualization interface prioritizes clarity and interpretability. Users can move smoothly from a high-level industry overview to detailed ticker-specific sentiment trends. This allows analysts, researchers, and investors to explore the data from multiple perspectives and identify market patterns that might not be visible through static reports or traditional sentiment indices.

4.3 Innovations and Technical Contributions

Ultimately, our approach introduces several innovations that make the system both technically robust and analytically novel:

1. LLM-Assisted Labeling Pipeline: Automatic sentiment labeling using Google Gemini removes the need for manual annotation and provides consistent, high-quality training data.
2. Cloud-Native Architecture: The entire workflow, from ingestion to model inference to data storage, operates within a unified GCP environment. This allows scalability and near real-time updates.
3. Finance-Aware Text Processing: The preprocessing pipeline preserves ticker symbols, numerical values, and emojis, which are essential for interpreting informal financial discussions.
4. Interactive Multi-Level Visualization: The dashboard presents sentiment at multiple levels of granularity. Users can explore market-wide patterns, sector flows, or individual ticker sentiment with full interactivity.

5. Evaluation

The evaluation focused on three components: the feed-forward neural network sentiment classifier, the Reddit ingestion pipeline, and the accuracy of the results shown in the visualization interface. The classifier was trained on the Gemini-labeled financial news dataset using an 80/20 split, and training and validation loss curves both stabilized around the same point, indicating good generalization. On the validation set, the model reached an accuracy of 85 percent. Precision scores were 0.82 for Negative, 0.83 for Neutral, and

0.87 for Positive, while recall values were 0.84, 0.82, and 0.86 for the same classes. The confusion matrix showed that the model handled clear sentiment well, with most mistakes appearing in neutral samples that leaned slightly positive or negative.

To test how well the model transfers to real-world data, we applied it to a separate batch of Reddit posts. These posts differ from headlines because they use slang, emojis, abbreviations, and shorter phrasing. Manual review confirmed that the model correctly identified sentiment in the majority of cases. The most common errors were sarcastic comments and posts that required broader context. The ingestion pipeline itself was also checked to confirm consistent cleaning, tokenization, and ticker extraction. Both standard tickers, such as TSLA, and dollar-sign formats were recognized reliably. All predictions, timestamps, and extracted tickers were stored correctly in PostgreSQL, and SQL checks confirmed that daily averages, rolling ninety-day statistics, and sector summaries were calculated accurately.

The visualization interface was evaluated by comparing dashboard outputs directly with SQL queries. Every tested sample matched, confirming that the interface accurately reflects the underlying data. Sentiment time-series charts captured daily swings clearly, and the rolling ninety-day averages helped reveal longer-term sentiment patterns. The sector heatmap also highlighted differences across industries in a consistent and interpretable way. A key limitation observed during evaluation is that less-popular tickers had weaker sentiment coverage and less stable scoring. These tickers appeared rarely in both the training set and the Reddit stream, while high-traffic names, especially the large technology stocks, appeared far more often. As a result, the model produced more reliable predictions for heavily discussed companies.

6. Conclusions and Discussion

This project demonstrates that combining LLM-labeled financial news with a lightweight neural network can produce an effective and efficient sentiment classifier for both structured and unstructured financial text. By integrating this model with a cloud-native Reddit ingestion pipeline and an interactive visualization dashboard, we created a system capable of tracking sentiment at the ticker and sector levels in a clear and interpretable way.

Our evaluation indicates that the classifier performs reliably on formal financial news and generalizes sufficiently to Reddit posts, even with informal language and slang. The visualization interface highlights meaningful sentiment shifts that align with real-world market narratives, showing the value of merging news sentiment with retail investor sentiment.

There remain several opportunities for future improvement. While the pipeline is designed to support continuous real-time ingestion, the current implementation processes a recent batch of scraped Reddit data rather than streaming live updates, which represents a natural next step for expansion. Additional enhancements could include integrating transformer-based sentiment models for improved nuance detection, expanding subreddit coverage to capture a broader sentiment signal, or combining sentiment data with live market feeds to explore predictive relationships.

Team Contribution Statement: All team members contributed a similar amount of effort to the project.

Bibliography

- Araci, D. (2019). FinBERT: Financial sentiment analysis with pre-trained language models. arXiv preprint arXiv:1908.10063. https://arxiv.org/abs/1908.10063
- Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. Journal of Computational Science, 2(1), 1–8.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of NAACL-HLT.
- Ding, X., Zhang, Y., Liu, T., & Duan, J. (2015). Deep learning for event-driven stock prediction. Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.
- Havre, S., Hetzler, E., & Nowell, L. (2002). ThemeRiver: Visualizing thematic changes in large document collections. IEEE Transactions on Visualization and Computer Graphics, 8(1), 9–20.
- Heer, J., & Shneiderman, B. (2012). Interactive dynamics for visual analysis. Communications of the ACM, 55(4), 45–54.
- Kirtac, K., & Germano, G. (2025). Large language models in finance: What is financial sentiment? doi: 10.2139/ssrn.5166656.
- Liu, T., et al. (2020). Event extraction as machine reading comprehension. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.
- Loughran, T., & McDonald, B. (2011). When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks. The Journal of Finance, 66(1), 35–65. https://doi.org/10.1111/j.1540-6261.2010.01625.x
- Ong, K., Van Der Heever, W., Satapathy, R., Cambria, E., & Mengaldo, G. (2023). FinXABSA: Explainable finance through aspect-based sentiment analysis. IEEE International Conference on Data Mining Workshops (ICDMW), 773–782. doi: 10.1109/ICDMW60847.2023.00105.
- Park, J., Lee, H. J., & Cho, S. (2021). Automatic construction of context-aware sentiment lexicon in the financial domain using direction-dependent words. arXiv preprint arXiv:2106.05723. http://arxiv.org/abs/2106.05723
- Shiller, R. J. (2017). Narrative economics. American Economic Review: Papers & Proceedings, 107(4), 967–1004.
- Sun, Y., Yuan, H., & Xu, F. (2025). Financial sentiment analysis for pre-trained language models incorporating dictionary knowledge and neutral features. Natural Language Processing Journal, 11, 100148. doi: 10.1016/j.nlp.2025.100148.
- Tetlock, P. C. (2007). Giving content to investor sentiment: The role of media in the stock market. The Journal of Finance, 62(3), 1139–1168. https://doi.org/10.1111/j.1540-6261.2007.01232.x
- Tetlock, P. C., Saar-Tsechansky, M., & Macskassy, S. (2008). More than words: Quantifying language to measure firms' fundamentals. The Review of Financial Studies, 21(2), 1–37.