Leveraging Transformer-Based Sentiment Analysis for Financial Market Insights

Anany Sachan

November 6, 2025

1 Introduction

Financial markets are profoundly influenced not only by economic fundamentals but also by the sentiments and psychology of investors. In recent years, the proliferation of online financial news, social media platforms, and discussion forums has created an abundance of textual data reflecting the real-time "market mood". This has spurred growing interest in sentiment analysis as a tool to quantify and track these emotions and opinions at scale. Major financial data providers now even offer sentiment indices as part of their analytics services, underscoring the perceived value of such measures [10]. In fact, the sentiment index of market participants has been extensively used for stock market prediction in recent years [10], with evidence that incorporating sentiment can improve forecasting accuracy and investment decisions.

However, harnessing unstructured sentiment data effectively remains challenging due to the sheer volume and velocity of text streams and the nuanced language (including slang, sarcasm, and domain-specific jargon) prevalent in financial discourse. These challenges motivate the development of a real-time sentiment analysis dashboard for finance—a system to "nowcast" market mood by continuously analyzing textual data sources. *Nowcasting*, in this context, refers to the real-time estimation of current market sentiment (as opposed to traditional forecasting which predicts future trends). A reliable nowcasting tool could alert traders and analysts to sudden shifts in investor optimism or fear, potentially offering early indicators of market movements. For example, collective bullish enthusiasm on social media forums was a driving force behind events such as the GameStop short squeeze in early 2021, where coordinated sentiment on Reddit's WallStreetBets forum helped fuel extreme volatility in GameStop's stock price [4]. This project's significance lies in bridging advances in natural language processing with financial analytics to capture such phenomena. Accordingly, this thesis presents the design and implementation of such a nowcasting dashboard, and outlines its anticipated contributions to financial market analysis.

2 Background

2.1 Natural Language Processing and Sentiment Analysis.

Sentiment analysis (or opinion mining) is a subfield of natural language processing (NLP) that focuses on identifying and extracting subjective information from text—typically the polarity (positive, negative, neutral) of opinions or the emotion and attitude expressed. Early approaches to sentiment analysis often relied on lexicon-based methods: using dictionaries of sentiment-laden words to determine a text's overall sentiment. For example, a simple lexicon-based system might count occurrences of "positive" words minus "negative" words to assign a sentiment score. Domain-specific lexicons (such as Loughran and McDonald's finance sentiment word lists) were developed to better handle financial terminology, which differs from everyday language (e.g., words like "bullish," "bearish," or "short" have special meanings in markets). While straightforward and interpretable, lexicon-based methods have inherent limitations—they cannot easily account for context, sarcasm, negation, or shifting word usages, and their accuracy hinges on the completeness of the predefined word list.

As the field matured, **machine learning** techniques quickly supplanted pure lexicon-based systems for sentiment classification tasks. Instead of fixed dictionaries, machine learning approaches learn to infer sentiment from examples of labeled text. A seminal work by Pang et al. (2002) showed that standard machine learning classifiers (Naïve Bayes, maximum entropy, and support vector machines) significantly outperformed human-crafted keyword baselines on movie review sentiment classification [8]. This study also highlighted that sentiment classification is more challenging than topic-based text classification, owing to the need to detect subtle linguistic cues (for instance, negation or sarcasm) rather than just content words [8]. The implication was that more sophisticated features and models were required to capture the nuance in sentiment-bearing language.

2.2 Deep Learning and Transformer Models.

In the past decade, advances in deep learning have dramatically improved the performance of NLP tasks, including sentiment analysis. Neural network architectures like **convolutional neural networks (CNNs)** and **recurrent neural networks (RNNs)** (particularly Long Short-Term Memory networks, LSTMs) enabled models to automatically learn rich feature representations from text data. These approaches surpassed earlier algorithms by capturing word order, semantic nuance, and contextual relationships more effectively than bag-of-words models or manual feature engineering [11]. Zhang et al.'s comprehensive survey (2018) notes that deep learning methods began to consistently outperform traditional classifiers (such as SVM or logistic regression) on sentiment tasks by learning multiple layers of abstract features [11]. The introduction of *word embeddings* (e.g., Word2Vec, GloVe) also boosted sentiment analysis, as words could be represented in vector spaces that encode semantic similarity, helping algorithms generalize beyond exact keyword matches.

The most significant recent breakthrough has been the emergence of **transformer-based language models**, epitomized by **BERT** (Bidirectional Encoder Representations from Transformers) introduced by Devlin et al. [5]. Transformer models use self-attention mechanisms to capture long-range dependencies in text and can be trained on massive corpora to learn contextual language representations. Unlike earlier RNNs, transformers process words in parallel and consider both left and right context simultaneously, enabling a deeper understanding of meaning. Fine-tuned transformer models now achieve state-of-the-art results on a wide range of NLP benchmarks, including sentiment classification. For example, one study found that a transformer-based classifier significantly outperformed an LSTM and other prior models on a large Twitter sentiment dataset, particularly excelling at handling the noisy, informal language of social media [9]. The success of transformers has been so pronounced that they have become the dominant paradigm in NLP, even spurring new directions such as *multimodal* sentiment analysis that combines text with audio or visual cues [2].

Despite these advancements, certain challenges in sentiment analysis persist. Subtle linguistic phenomena like sarcasm, idioms, and context-dependent irony remain difficult for algorithms to fully grasp [7]. There are also concerns about biases in models (e.g., language models learning biased associations from training data) and how well models generalize across domains [7]. Notably, the financial domain presents a unique context: language in analyst reports, news headlines, or trader chats can be very domain-specific, filled with jargon and phrases that are rare in general text corpora. A general-purpose model might misinterpret or simply not understand such domain-specific language. This gap has led researchers to pursue domain adaptation strategies—tailoring NLP models specifically for finance.

2.3 FinBERT and Domain-Specific Modeling

One pivotal development in this regard was the creation of **FinBERT** by Araci (2019) [1]. FinBERT is a variant of the BERT model that was further pre-trained on large volumes of financial text (e.g., news articles, earnings reports, and financial forums) to imbue it with domain-specific knowledge. The motivation for FinBERT was that while generic BERT captures general language patterns, it may struggle with specialized terminology and context found in finance (for instance, interpreting "market rally" or "dead cat bounce"). By continuing BERT's training on a finance corpus and then fine-tuning it for sentiment classification, FinBERT achieved superior performance on financial sentiment tasks compared to off-the-shelf models [1].

In evaluations, FinBERT consistently outperformed general models and earlier deep learning methods when classifying the sentiment of financial news and reports [1]. This demonstrated that domain-specific language models can substantially improve accuracy by accounting for the nuances of industry-specific language. FinBERT and similar finance-focused NLP models have since been widely adopted in both research and industry for tasks like analyzing news sentiment, earnings call transcripts, and social media discussions related to stocks.

In summary, the evolution of sentiment analysis techniques—from lexicon approaches to machine learning, and from simple classifiers to transformers like BERT—has provided an expanding toolkit for tackling the problem of understanding market mood. The challenge now lies in applying these tools effectively to real-time financial data streams, which is the focus of our project.

3 Related Work

Research at the intersection of textual sentiment analysis and finance is rich and multifaceted. Broadly, prior work can be grouped into two themes: (1) developing specialized sentiment analysis models for financial language, and (2) applying sentiment-based indicators to financial forecasting or market analysis.

3.1 Financial Sentiment Models

A cornerstone in this area is the aforementioned **FinBERT** model. Araci's work [1] demonstrated that adapting a transformer to financial text data yields clear benefits for sentiment classification in finance. FinBERT's introduction has spurred further research into domain-specific NLP, and it serves as a foundation for many subsequent studies. For instance, Jiang and Zeng [6] leverage FinBERT to extract sentiment signals from financial news, which they then input into a predictive model for stock movement. In their approach, daily news articles are fed through FinBERT to produce sentiment scores or embeddings, and these features are used alongside an LSTM-based temporal model to forecast stock price trends. They report that the FinBERT-enhanced model significantly outperforms comparable models using a generic BERT or using no text input at all, confirming that finance-tailored language models can improve predictive accuracy in market tasks. This finding aligns with the general intuition that more informative representations of text (in this case, capturing finance-specific context) translate into better downstream predictions.

3.2 Sentiment in Market Prediction

Even before the deep learning era, researchers explored links between public sentiment and market behavior. A variety of textual sources have been studied, including news, financial reports, and social media. Early studies (e.g., 2000s-era works by Tetlock and others) found that negative tone in news or investor forums can predict short-term dips in stock prices, suggesting that sentiment contains predictive signal. More recent work has continued to validate and extend these insights. Xing et al. (2018) provide a clear example by constructing a sentiment index from social media posts and integrating it into an asset allocation framework [10]. They use an ensemble of clustering and LSTM models to process streams of Twitter and forum data, distill a market sentiment time series, and incorporate it as "market views" in a Bayesian portfolio optimization. The result was improved portfolio performance (in terms of stability and returns) compared to strategies that ignore sentiment [10]. This study not only underscores that sentiment can enhance predictive models, but also illustrates a methodology for merging textual signals with traditional financial theories (Modern Portfolio Theory, in that case). Similarly, other works have used sentiment extracted from news headlines or financial blogs to forecast stock returns or volatility, often reporting that sentiment features add incremental predictive power on top of technical or fundamental features. The consensus emerging from these studies is that there is measurable information content in the collective mood of market participants, which, if quantified correctly, can be useful for nowcasting and forecasting financial market dynamics.

3.3 Social Media and Alternative Data

A particularly vibrant strand of recent research focuses on social media sentiment, given the outsized impact platforms like Twitter, Reddit, and StockTwits now have on retail investor behavior. Social media data is noisy and rife with slang, memes, and unstructured narratives, making it challenging for traditional NLP models.

Deng et al. (2023) highlight this challenge in their Reddit sentiment analysis study: they note that understanding content from the r/WallStreetBets community requires both financial knowledge and fluency in internet vernacular, which makes obtaining high-quality labeled data difficult [3]. To tackle this, they employ a **semi-supervised learning** pipeline using a large language model (GPT-3 variant) to generate "weak" sentiment labels for thousands of Reddit posts. These LLM-generated labels (refined through prompt techniques like chain-of-thought reasoning) are then used to train a smaller, deployable model [3]. Remarkably, with only a handful of manual prompts to guide the LLM, the final distilled model achieved accuracy on par with fully supervised models—illustrating the great potential of LLMs to bootstrap sentiment analysis when human-labeled data is scarce.

This approach is especially relevant for finance, where new slang (e.g., "diamond hands," "to the moon") and rapidly evolving topics can quickly outdated static lexicons or past training data. In addition to methodological advances, social media has also been the subject of case studies linking sentiment to market events. For example, Desiderio et al. (2025) examine the dynamics of the Reddit-driven GameStop short squeeze, quantitatively analyzing how collective bullish sentiment online coalesced into a coordinated buying frenzy [4]. Their findings shed light on the feedback loop between viral social-media sentiment and extreme market outcomes, reinforcing why real-time monitoring of such sentiment is important.

4 Methodology

Our primary goal is to create a real-time sentiment analysis dashboard that can nowcast market mood by aggregating textual data from multiple sources. In particular, we focus on two contrasting data streams: (1) financial news articles, which provide formal and vetted information, and (2) social media posts from platforms like Reddit, which capture informal investor opinion. These sources were chosen to balance credibility and timeliness - news reflects the perspectives of journalists and analysts, while Reddit (e.g., posts from communities such as `r/WallStreetBets`) reflects retail investor sentiment in the wild. Both have been identified in prior research as rich sources of market sentiment [10]. The target sentiment categories in our analysis are positive, negative, or neutral, following standard practice in sentiment classification. Each incoming text will be labeled with one of these polarities. We employ the transformer-based FinBERT model for this purpose, as it has been pre-trained and fine-tuned specifically for financial language [1], making it well-suited to detect nuances in finance-specific text (e.g., distinguishing bullish vs. bearish context).

At a high level, our system's architecture follows a pipeline from user input to sentiment output. Figure 1 illustrates this overall architecture. The process can be summarized in several key stages:

1. **User Input & Query Formation:** The user selects or inputs a target of interest

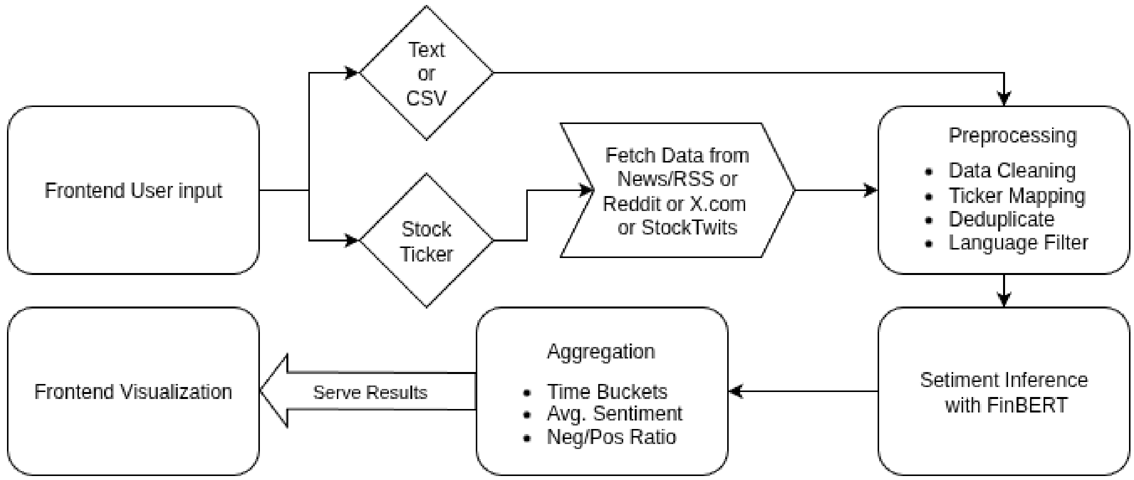

Figure 1: System architecture for the Dashboard

(such as a specific stock ticker, company name, or market topic). This query defines the context for sentiment analysis, ensuring that only relevant data is collected (e.g., news articles about that company, or Reddit posts mentioning the ticker).

2. **Data Ingestion from Multiple Sources:** For each query, the system concurrently fetches new textual data from the chosen sources. A news adapter retrieves the latest relevant news articles (e.g., headlines or snippets from financial news APIs or RSS feeds), and a social media adapter pulls recent Reddit posts or comments from finance-related subreddits that mention the query. We throttle and schedule these requests to effectively capture real-time information without overloading any single source.

3. **Text Preprocessing:** The collected texts are then cleaned and preprocessed. This step includes removing irrelevant content (such as HTML tags, URLs, or special characters), normalizing text (lowercasing, handling common slang or abbreviations), and potentially filtering out very short or non-English messages. Domain-specific tokens (e.g., ticker symbols like GME, or slang like "HODL") are preserved since they carry information. The goal is to format each text into a clean input string suitable for the language model.

4. **Sentiment Classification with FinBERT:** Each preprocessed text is fed into the FinBERT model to predict its sentiment category. FinBERT's output includes probabilities for the three classes, which we convert into a discrete label (positive/negative/neutral) for simplicity. Because FinBERT is trained on financial corpora, it can interpret context that generic models might misclassify; for example, it understands that a sentence like "Dow falls on fears of rate hike" should be read as a negative sentiment regarding market outlook. We do not further fine-tune FinBERT in this project (to avoid expensive training), relying on the model's off-the-shelf capabilities [1]. Each piece of text now has an associated sentiment score or label.

5. **Aggregation and Sentiment Index Computation:** Finally, the system aggregates individual sentiments into an overall sentiment indicator. Depending on the use case,

this could be a simple aggregate (such as the percentage of positive vs. negative items in the last hour) or a weighted sentiment score. In our implementation, we compute a sentiment index by assigning +1 to positive, -1 to negative (and 0 to neutral) classifications and averaging over all texts within a given time window. This yields a continuous sentiment signal that can be tracked over time. The index can be visualized as a time series to show sentiment trends (e.g., a sharp drop in the index might indicate a surge of pessimistic news and posts). In addition, separate sentiment tallies for each source are maintained so that the dashboard can display source-specific sentiments side by side.

Through this pipeline, the system effectively transforms unstructured text streams into quantifiable sentiment metrics in real-time. The methodology ensures that our sentiment analysis is context-aware (thanks to FinBERT's domain knowledge) and comprehensive in coverage (by drawing from both news media and social media). In the next section, we discuss how this methodology was realized in practice via our system's implementation.

5 Implementation

To implement the above approach, we developed a modular web-based application consisting of a Python backend and a JavaScript frontend. The backend is built with the lightweight Flask framework, which exposes a RESTful API for data retrieval and sentiment analysis. The frontend is built in React, creating an interactive single-page application that displays the sentiment dashboard. This separation of concerns allows the heavy NLP processing to occur server-side, while users interact with a smooth, responsive interface in their web browser. The overall software architecture is service-oriented: the Flask API endpoints serve JSON data that the React frontend fetches and renders. Figure 2 shows a prototype of the user interface, including charts for sentiment trends and controls for the user to select different stock tickers or time ranges.

On the backend, we implemented dedicated ingestion adapters for each data source. *Current plan, not implemented yet:* Specifically, one adapter handles retrieving relevant news articles while another collects data from social media. For news, we connect to a financial news API or RSS feed to fetch the latest headlines and snippets related to the user's query. For social media, we use Reddit's API (via a Python wrapper) to gather recent posts from finance-focused communities (e.g., `r/WallStreetBets`) that mention the query. These adapters run as separate asynchronous tasks, allowing concurrent data collection from both sources in near real-time. We also apply basic rate-limiting (e.g., fetching updates every minute) to avoid overwhelming any single source.

Both adapters feed incoming text to a preprocessing pipeline. Implemented in Python, this pipeline cleans and normalizes text as described in the methodology. Specifically, we remove URLs, user mentions, and markup from Reddit text (while preserving salient tokens like cashtags or tickers that start with $). We also strip common stop words and punctuation from news snippets to reduce clutter for the language model. In this stage, we pay attention to case normalization (since FinBERT is cased, we generally preserve case), and we ensure that each text does not exceed BERT's maximum token length (512 tokens). If a news article is too long (which is uncommon for headlines/snippets), it is truncated or split into multiple segments for analysis.

After preprocessing, the core sentiment classification step is handled by FinBERT. We integrate FinBERT using the Hugging Face Transformers library, loading a pre-trained FinBERT model fine-tuned for sentiment analysis. The model is loaded once when the server starts and remains in memory to serve predictions quickly on demand. For each cleaned text, we call the model's inference pipeline (tokenizing the text and obtaining the softmax probabilities for each sentiment class). We then assign a sentiment label based on the highest probability. This entire process is encapsulated in a Flask route (for example, a `/analyze` endpoint) so that the front-end can trigger sentiment analysis via an HTTP request.

On the frontend side, the React dashboard polls these API endpoints to get the newest data. The interface is designed to present key information at a glance. It includes: (a) a time-series chart that plots the sentiment index over time, updating in near real-time (e.g., a new point every minute), (b) a bar chart or pie chart showing the proportion of positive/negative/neutral posts in the latest window (with different colors for news vs. social media contributions), and (c) controls for the user to change the query or adjust settings (such as selecting a different subreddit or news source, or pausing the live updates). Figure 2 provides a snapshot of the dashboard prototype, where the sentiment index trend line is shown alongside a table of recent high-sentiment headlines. The UI emphasizes clarity and immediacy, using visual cues (green for positive, red for negative) to highlight sentiment changes. We also implement basic error handling and status indicators (for example, if an API call fails or if no data is available for a given query, the UI displays a message so the user is aware of the system state).

Throughout the implementation, a priority was placed on modularity and extensibility. New data sources (for example, Twitter or StockTwits) could be added by writing a new adapter that conforms to the same interface. Likewise, the sentiment model can be swapped out or upgraded (for instance, if a newer finance-specific LLM becomes available) with minimal changes to the surrounding code, thanks to the abstraction of the `/analyze` API route. The use of Flask and React for the prototype proved sufficient for our needs; Flask can comfortably handle the moderate request load of our polling mechanism, and React enables dynamic updates and interactive visualizations in the browser.

Placeholder figure to be replaced with an actual screenshot

Figure 2: Prototype of the web-based dashboard UI. The interface includes (left) a line chart showing the computed sentiment index over time for a selected stock, and (right) a summary of recent posts/news with their individual sentiment classifications (green for positive, red for negative, gray for neutral). Users can input different tickers or topics to refresh the analysis.

6 Results

At this stage of the project, data collection and integration are ongoing, so we present an outline of the expected results and how we plan to analyze them. The evaluation of our system will be primarily qualitative and exploratory, given that we do not have a labeled "ground truth" for real-time market sentiment. Key results will revolve around visualizing sentiment trends, comparing sources, and assessing the FinBERT model's outputs. We anticipate the following findings and will include corresponding charts and tables in the final report:

1. **Sentiment Trend Visualization:** We will present time-series plots of the computed sentiment index for a particular financial instrument or topic over various time windows (e.g., intraday, daily, weekly). For example, Figure 3 will show the sentiment index for a chosen stock (say, Tesla) over a multi-day period. These charts will illustrate how sentiment fluctuates in response to market events, news releases, or social media activity. For example, we expect to see spikes in negative sentiment during market sell-offs or positive sentiment following favorable earnings reports. By examining these patterns, we can demonstrate the system's ability to capture market mood dynamics. In the final thesis, we plan to highlight a couple of specific case studies where the sentiment timeline clearly reflected an important market narrative, thereby showcasing the dashboard's practical relevance.

Placeholder for sentiment index timeline visualization

Figure 3: Example sentiment index timeline for a selected stock over one week. The y-axis represents the aggregated sentiment score (positive or negative tilt), and the x-axis is time (days). Notable external events (earnings release, major news) are annotated to show alignment with sentiment spikes.

2. **Source Comparison Analysis:** We will analyze and compare sentiment distributions between news articles and social media posts. Table 1 illustrates the kind of summary we will provide. We hypothesize that social media may exhibit more extreme sentiment swings (due to retail investor enthusiasm or panic) compared to the more measured tone of news media. Statistical summaries (e.g., mean sentiment scores, variance) will quantify these differences.

Placeholder for source-wise sentiment breakdown table

Table 1: This table will list, for a given period, the number of items from each source (News vs. Reddit), and the percentage of those items labeled as Positive, Neutral, or Negative by FinBERT. It will also include the average sentiment score per source. An example (for illustration only) might be: News: 40 articles (15% pos, 70% neutral, 15% neg, avg score = 0.0); Reddit: 120 posts (25% pos, 50% neutral, 25% neg, avg score = 0.0).

3. **Model Output Assessment:** We will examine the reliability and characteristics of the FinBERT model's outputs on our collected data. Although we cannot directly measure accuracy without labeled test data, we can analyze proxies for reliability. One analysis will involve checking the distribution of predicted labels. If we find, for instance, that an overwhelmingly high percentage of items are labeled "Neutral," it might indicate that FinBERT is defaulting to neutral in uncertain cases (which could

dampen the sensitivity of our index). Conversely, a balanced or contextually appropriate distribution (e.g., more negatives on a day of bad news) would build confidence in the model's usefulness. We may include a chart showing the proportion of each sentiment class in a volatile period versus a calm period to illustrate that the model's output varies with market conditions (which it should if it's capturing real sentiment). Additionally, we plan a qualitative assessment: manually reading a sample of texts alongside the model's label to subjectively judge correctness. Early informal tests suggest FinBERT is quite adept with straightforward cases (it correctly labels headlines like "Stocks soar after earnings beat" as positive, and "Company X faces fraud allegations" as negative). However, it can be challenged by sarcasm or figurative language in social posts - a known difficulty for sentiment models [7]. We will document any systematic errors observed (for example, finance slang that confuses the model, or instances where context was missed) and may update our preprocessing or handling of model confidence accordingly.

7 Discussion

With results still preliminary, we discuss how the expected findings relate to prior research and the broader goals of the project. Several key aspects of the analysis are considered:

1. **Relationship to prior work:** The outcomes of our project align with several trends documented in prior literature. Our use of FinBERT, for instance, leverages Araci's (2019) insight that a domain-specific language model can outperform generic models in financial sentiment tasks [1]. Consistent with Jiang and Zeng's findings [6], we expect that FinBERT's specialized understanding of finance terminology will yield more relevant sentiment signals than a one-size-fits-all approach. Moreover, the concept of aggregating sentiment from multiple sources resonates with studies that have emphasized the value of alternative data in market analysis. For example, Xing et al. [10] successfully incorporated a sentiment index into an investment strategy, supporting the notion that textual sentiment can provide unique predictive insights. Our real-time dashboard builds on these foundations, translating prior research findings into an integrated, practical tool.

2. **System Strengths:** Our system offers several notable strengths. First, by fusing multiple data sources (news and social media) in real time, it captures a more holistic view of market sentiment than single-source analyses. This dual-source approach allows us to detect signals that might appear in one channel but not the other, giving a broader situational awareness. Second, the use of FinBERT imbues the system with domain-specific intelligence—financial jargon and context that might confound a generic model are accurately interpreted, which improves classification reliability. The system's modular, service-oriented architecture is also a strength: it ensures scalability and flexibility. New information sources (e.g., other forums or data streams) can be integrated with minimal changes, and the sentiment analysis component can be upgraded (for instance, swapping in a newer model) without rebuilding the entire

pipeline. Finally, the dashboard interface emphasizes clarity, translating complex analytics into intuitive visualizations that allow users to grasp sentiment trends at a glance. These strengths collectively make the platform robust for continuous deployment in a dynamic environment like finance.

3. **Implications for investors and analysts:** Myriad practical applications emerge from our sentiment nowcasting tool. For investors and market analysts, the dashboard can serve as an early warning system or a confirmatory signal for market moves. A sudden surge in negative sentiment across news and social channels, for instance, might precede a broader sell-off or indicate growing fear in the market—information that could prompt a portfolio manager to adopt a more defensive stance. Conversely, an uptick in optimistic chatter around a particular stock or sector might flag an emerging opportunity (or a risk of a hype-fueled bubble) that merits closer investigation. By quantifying and visualizing crowd psychology, the system provides an additional layer of insight beyond traditional market indicators. This aligns with observations from events like the GameStop short squeeze, where monitoring online forums such as `r/WallStreetBets` could have revealed the brewing momentum behind the price surge [4]. In essence, our tool bridges qualitative sentiment and quantitative analysis, helping decision-makers incorporate the human factor into their strategies.

4. **Limitations of the current system:** Our prototype has some clear limitations. One is the scope of data: the analysis is currently restricted to Reddit and a single news source, leaving out other influential platforms (e.g., Twitter, financial blogs) and limiting the breadth of market coverage. This means the sentiment index might not fully represent the entire investor community at any given time. Another limitation is the reliance on FinBERT without further tuning; while effective generally, the model may miss new slang or nuanced sarcasm in social media posts—a known challenge for sentiment models [7]. This can lead to misclassifications or a bias toward "neutral" labels in uncertain cases. Additionally, because we lack ground-truth sentiment labels for real market data, it is difficult to quantitatively assess the accuracy of our classifications; our evaluation remains largely qualitative. The real-time nature of the system also presents practical challenges: network delays, API rate limits, or data outages could disrupt the analysis stream. Finally, as a proof-of-concept, the current implementation is not optimized for large-scale deployment; significant engineering effort would be needed to handle higher data volumes or to integrate the tool into live trading workflows. These limitations should be kept in mind when interpreting the results and point to avenues for improvement.

5. **Potential improvements and future directions:** Building on the above limitations, there are several directions for future enhancement. First, expanding the range of data sources would make the sentiment index more comprehensive. Integrating additional platforms—such as Twitter, other finance forums, or even transcripts of earnings calls—could capture sentiment signals that our current Reddit-and-news focus might miss. Second, the sentiment analysis model can be refined. Future work could involve fine-tuning FinBERT on more recent and diverse financial texts to keep up with evolving language, or experimenting with state-of-the-art models (including large language

models in a prompt-based setup) to improve classification nuance. Another important step is to rigorously evaluate the system's predictive value. For example, we could analyze how changes in our sentiment index correlate with subsequent market movements (e.g., next-day stock volatility or returns) to determine if the sentiment signals have forecasting power. If such correlations prove strong, the tool could be extended beyond nowcasting to actually aid in market forecasting. Finally, user-facing improvements like alert mechanisms for extreme sentiment shifts or more granular filtering options (by sector, asset type, etc.) could increase the tool's practical utility. Pursuing these improvements would significantly enhance the system's accuracy, robustness, and real-world applicability.

8 Conclusion

In summary, this study has developed a framework for leveraging transformer-based sentiment analysis in the context of financial markets. We designed and implemented a real-time dashboard that aggregates sentiment from both news media and social media, providing a nuanced "market mood" indicator for investors and analysts. By utilizing FinBERT, a language model tailored to finance, our system can understand domain-specific chatter and news with greater fidelity than generic sentiment tools. The expected outcome is a tool that not only visualizes sentiment trends in an intuitive way but also bridges a gap between cutting-edge NLP techniques and practical financial analytics.

This work contributes an integrated approach that marries multiple data sources and advanced AI in a real-time application. The modular design means it can be continually updated: new data feeds and improved models can be incorporated as the landscape evolves. In its current form, the dashboard serves as a proof-of-concept for how unstructured textual data might be harnessed for market insight, aligning with a growing industry trend of using alternative data in finance.

For future work, there are several promising directions. One immediate extension would be to expand the coverage of the system - for example, including Twitter data or transcripts of earnings calls to capture even more facets of market sentiment. Another avenue is to refine the sentiment analysis itself, perhaps by fine-tuning the model on a larger, more recent corpus of financial discussions (or employing prompt-based large language models for greater nuance). There is also potential to quantitatively evaluate the predictive value of the sentiment index by correlating it with subsequent market movements, thus moving from nowcasting towards forecasting. Ultimately, we envision that efforts like this project can evolve into robust tools that assist investors by flagging early signs of optimism or fear in the market. In conclusion, the project underscores the value of interdisciplinary approaches - combining NLP, data engineering, and finance - to innovate in understanding and visualizing the complex emotional undercurrents that drive economic behavior. We hope this study lays groundwork for more sophisticated sentiment-driven analytics in the financial domain, and we have outlined several improvements that can be pursued to enhance both the accuracy and utility of such systems in the future.

References

[1] Araci, D. FinBERT: Financial Sentiment Analysis with Pre-trained Language Models. *arXiv preprint arXiv:1908.10063* (2019).

[2] Das, R., and Singh, T. D. Multimodal Sentiment Analysis: A Survey of Methods, Trends, and Challenges. *ACM Comput. Surv. 55*, 13s (July 2023).

[3] Deng, X., Bashlovkina, V., Han, F., Baumgartner, S., and Bendersky, M. LLMs to the Moon? Reddit Market Sentiment Analysis with Large Language Models. In *Companion Proceedings of the ACM Web Conference 2023* (New York, NY, USA, 2023), WWW '23 Companion, Association for Computing Machinery, pp. 1014–1019.

[4] Desiderio, A., Aiello, L. M., Cimini, G., and Alessandretti, L. The dynamics of the Reddit collective action leading to the GameStop short squeeze. *npj Complexity 2*, 5 (Feb. 2025).

[5] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (Minneapolis, Minnesota, June 2019), J. Burstein, C. Doran, and T. Solorio, Eds., Association for Computational Linguistics, pp. 4171–4186.

[6] Jiang, T., and Zeng, Q. Financial sentiment analysis using FinBERT with application in predicting stock movement. *arXiv preprint arXiv:2306.02136* (2025).

[7] Mohammad, S. M. Sentiment Analysis: Automatically Detecting Valence, Emotions, and Other Affectual States from Text. In *Emotion Measurement (Second Edition)*, H. Meiselman, Ed. Elsevier, 2021.

[8] Pang, B., Lee, L., and Vaithyanathan, S. Thumbs up? Sentiment Classification using Machine Learning Techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10* (USA, 2002), EMNLP '02, Association for Computational Linguistics, pp. 79–86.

[9] Tang, J. Leveraging Transformer Neural Networks for Enhanced Sentiment Analysis on Online Platform Comments. In *Proceedings of the 2023 4th International Conference on Machine Learning and Computer Application* (New York, NY, USA, 2024), ICMLCA '23, Association for Computing Machinery, pp. 256–260.

[10] Xing, F. Z., Cambria, E., and Welsch, R. E. Intelligent Asset Allocation via Market Sentiment Views. *Comp. Intell. Mag. 13*, 4 (Nov. 2018), 25–34.

[11] Zhang, L., Wang, S., and Liu, B. Deep learning for sentiment analysis: A survey. *WIREs Data Mining and Knowledge Discovery 8*, 4 (2018), e1253.