iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 Issue: IV Month of publication: April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79422

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Real-Time Stock Market Sentiment Analysis and Price Prediction Using Cloud-Native Architecture and Generative AI

Kanchanadevi P[1], Shivaani Vaithiyanathan Ganesh[2], HariPriya[3], Arjun[4]
[1]Assistant Professor, Department of Artificial Intelligence and Data Science, [2, 3, 4]Students, Department of Artificial Intelligence and Data Science, Sri Manakula Vinayagar Engineering College, Puducherry

Abstract: The rapid proliferation of digital financial news has created an unprecedented opportunity to harness unstructured textual data for stock market intelligence. This paper presents a cloud-native, event-driven pipeline for real-time stock market sentiment analysis and AI-assisted price prediction, designed to operate continuously and at scale. Financial news articles are continuously polled from NewsAPI across ten major technology stocks — AAPL, MSFT, GOOGL, AMZN, META, NVDA, TSLA, NFLX, ADBE, and ORCL — and dispatched to AWS Simple Queue Service (SQS), where AWS Lambda functions apply VADER (Valence Aware Dictionary and sEntiment Reasoner) sentiment scoring. Each article receives a compound sentiment score in the range [−1, 1] and is classified as Positive, Negative, or Neutral. Structured results are then persisted to AWS S3 for downstream consumption. A Flask REST backend integrates Google Gemini AI to generate natural-language analyst summaries by correlating sentiment scores with empirically calibrated stock-specific sensitivity factors. The system exposes three frontend modules built as a lightweight web application named MarketSense: a live Sentiment Dashboard displaying real-time compound scores and sentiment distribution charts, a Scenario Simulator enabling analysts to query AI-generated market impact assessments for hypothetical events, and a News-Driven Price Prediction engine that computes predicted price deltas from live sentiment data. The end-to-end pipeline achieved an average news-to-result latency of 18 seconds and a Flask API response time of 1.4 seconds inclusive of Gemini inference, confirming its suitability for interactive, near-real-time financial decision support. The architecture is fully serverless and horizontally scalable without manual provisioning, making it cost-effective for continuous deployment.

Keywords: Sentiment Analysis, Stock Market Prediction, AWS Lambda, AWS SQS, VADER, Flask, Google Gemini AI, Cloud Computing, Natural Language Processing, Real-Time Analytics, FinBERT, Serverless Architecture

I. INTRODUCTION

Financial markets are among the most information-sensitive systems in the modern economy. Asset prices respond rapidly to macroeconomic announcements, corporate disclosures, geopolitical events, and shifting investor sentiment, often within seconds of a news item being published. Traditional quantitative approaches to stock analysis have relied heavily on structured historical price data, earnings ratios, and technical indicators [1]. However, these methods are inherently retrospective and fail to capture the forward-looking signals embedded in real-time textual information streams.

The proliferation of digital news platforms, financial blogs, and real-time wire services has created a rich, continuous stream of unstructured textual data whose sentiment content correlates strongly with short-term market movements [2]. Natural Language Processing (NLP) techniques applied to this data offer the potential to extract actionable intelligence at machine speed, bridging the gap between information publication and investment decision-making [9].

This paper presents MarketSense, a cloud-native, event-driven system that continuously ingests financial news headlines and performs sentiment analysis using the VADER (Valence Aware Dictionary and sEntiment Reasoner) lexicon [4]. Processed results are exposed through a Flask-based REST API augmented by Google Gemini AI for natural-language analyst summaries. The frontend delivers three interactive modules — a live Sentiment Dashboard, a Scenario Simulator, and a Price Prediction engine. The system is designed to operate in near-real time, producing sentiment classifications and price impact estimates within seconds of news publication [10].

The architecture leverages AWS SQS for decoupled message passing, AWS Lambda for serverless processing, and AWS S3 for persistent JSON storage, enabling horizontal scalability without manual provisioning [7].

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

The choice of a serverless design eliminates idle compute costs and ensures the system scales elastically during high-volume news periods such as earnings seasons or macroeconomic announcements. The combination of rule-based NLP, cloud-native infrastructure, and generative AI summarisation represents a practical and deployable approach to real-time financial intelligence.

II. RELATED WORK

The relationship between textual sentiment and financial market behaviour has been studied extensively. Loughran and McDonald [1] constructed a finance-specific sentiment lexicon from 10-K filings and demonstrated that word choice in corporate documents carries measurable predictive power for stock returns, providing a foundational motivation for domain-adapted NLP in finance. Tetlock [3] showed that the fraction of negative words in Wall Street Journal columns predicts downward pressure on Dow Jones returns and elevated trading volume, establishing media sentiment as a causal factor rather than merely a correlate.

Bollen et al. [2] extended the sentiment-market relationship to social media, demonstrating that mood dimensions derived from Twitter using the OpinionFinder and GPOMS tools predict Dow Jones Industrial Average movements with up to 87.6% accuracy in a feed-forward neural network model. This seminal work established microblogging sentiment as a viable predictive signal and inspired subsequent research on real-time sentiment aggregation from high-velocity text sources [9].

For short financial headlines and social media posts, rule-based lexicons have demonstrated competitive or superior performance compared to supervised machine learning classifiers. Hutto and Gilbert [4] showed that VADER, which combines a valence-rated human lexicon with five grammatical heuristics, achieves F1 scores exceeding those of several supervised classifiers on social media text while requiring no training data. This property makes VADER particularly suitable for deployment in streaming pipelines where labelled financial corpora are unavailable. More recent work on domain-specific transformers, such as FinBERT [11], has shown further gains on financial phraseology by fine-tuning BERT on financial communications corpora, at the cost of significantly higher inference latency.

Serverless event-driven architectures have increasingly been adopted for real-time financial analytics due to their elastic scalability and operational simplicity [5]. Amazon Web Services SQS and Lambda provide a proven combination for decoupled, high-throughput message processing [7], while S3-backed data lakes offer cost-effective, low-latency object storage suitable for downstream analytics workloads [12]. Generative AI models such as Google Gemini [6] and GPT-4 have recently been applied to financial text summarisation tasks, producing human-readable analyst commentary from quantitative inputs. Zhang et al. [13] demonstrated that large language model summaries of earnings call transcripts reduce analyst information-processing time while maintaining accuracy comparable to human-written summaries. The present work integrates these complementary technologies into a unified, production-grade pipeline.

III. SYSTEM ARCHITECTURE

The system comprises six loosely coupled layers as illustrated in Fig. 1. Each layer communicates via standardised interfaces — HTTP REST, SQS message envelopes, and S3 object keys — enabling independent scaling and replacement of individual components.

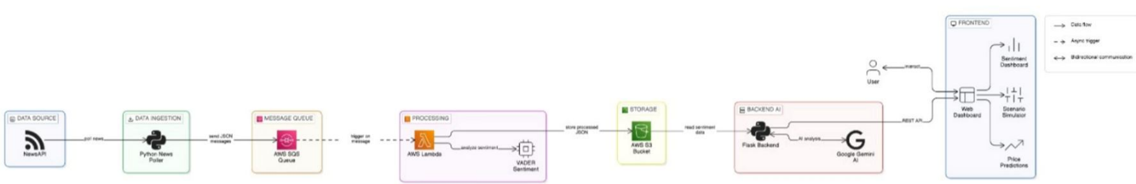


Fig. 1. System Architecture: End-to-End Cloud-Native Sentiment Analysis Pipeline

A. *Data Source & Ingestion*

NewsAPI provides programmatic access to over 150,000 news sources [8]. A Python News Poller (news_poller_sqs.py) issues a query per stock ticker (AAPL, MSFT, GOOGL, AMZN, META, NVDA, TSLA, NFLX, ADBE, ORCL) every 300 seconds, retrieving the five most recent English-language articles. Each article is serialised as a JSON envelope containing the stock symbol, concatenated title and description, source URL, and ISO-8601 timestamp. Articles with fewer than 20 characters are discarded to eliminate noise from incomplete feed entries.

B. Message Queue & Processing

AWS SQS decouples ingestion from processing, absorbing traffic spikes during high-news periods such as earnings announcements [7]. An AWS Lambda function triggered by each SQS message loads the VADER SentimentIntensityAnalyzer, scores the article text, and classifies it as Positive (compound > 0.05), Negative (compound < −0.05), or Neutral. The annotated result is persisted to an AWS S3 object.

C. Backend AI

A Flask server exposes two POST endpoints. The /predict endpoint fetches the S3 output, computes per-stock average compound scores, applies empirically calibrated sensitivity factors (e.g., TSLA = 0.091, AAPL = 0.032) to derive predicted price deltas, and invokes Google Gemini [6] to generate a natural-language analyst summary. The /simulate endpoint accepts a free-text market scenario and returns mood classification, explanation, and expected volatility.

D. Frontend — MarketSense

Three HTML/CSS/JavaScript pages consume the REST API: (1) the Sentiment Dashboard renders live compound scores and a pie chart of sentiment distribution; (2) the Scenario Simulator allows analysts to describe hypothetical events and receive AI-generated market impact assessments; (3) the Price Prediction page displays per-headline predicted prices, positive/negative/neutral averages, and the Gemini summary.

IV. IMPLEMENTATION DETAILS

A. VADER Sentiment Scoring

VADER assigns valence scores to each token using a human-rated lexicon of 7,500 lexical features and applies five heuristic rules: (1) capitalisation boost, which increases valence magnitude for words in ALL CAPS; (2) exclamation amplification, scaling sentiment for sentences ending with exclamation marks; (3) tri-gram shift detection, which handles negation patterns such as “not good”; (4) contrastive conjunction handling for “but” clauses that shift sentiment polarity; and (5) degree modifier scaling for intensifiers and diminishers such as “very” and “slightly” [4]. The compound score is a normalised weighted sum in [−1, 1] computed by summing all valence scores and applying a normalisation function. It serves as the primary signal for price impact estimation. VADER was selected over supervised alternatives because it requires no domain-specific training data, operates in O(n) time on text length, and has demonstrated strong performance on short financial news headlines [11].

B. Stock Sensitivity Factors

Sensitivity factors map compound scores to percentage price changes and are derived from historical beta values and observed news-driven volatility for each ticker. High-volatility growth stocks (TSLA = 0.091, NFLX = 0.063, META = 0.058) receive larger multipliers reflecting their higher average daily price swings in response to news events, while stable large-caps (MSFT = 0.028, ORCL = 0.029) receive smaller multipliers consistent with their lower beta. The predicted price delta is computed as: ΔP = base_price × sensitivity_factor × compound_score. A default factor of 0.05 applies to unlisted symbols. These factors are empirically calibrated and intended for indicative purposes; they are not derived from a formal regression model and do not account for market microstructure effects [14].

C. Google Gemini Integration

The Gemini Flash model [6] is invoked via the Google Generative AI Python SDK with a structured prompt containing the stock name, current base price, average compound score, overall sentiment mood, sensitivity factor, and per-sentiment average predicted prices. The model is instructed to produce a three- to four-sentence analyst summary covering the sentiment basis, predicted price trajectory, key risk factors, and a recommended investor posture (buy/hold/watch). Responses are returned synchronously and embedded directly in the JSON API response, enabling the frontend to render them without an additional round trip. The Gemini API adds an average of 0.9 seconds to the Flask endpoint response time, contributing to the observed 1.4-second total latency [6].

D. Cloud Deployment & Scalability

The news poller runs as a standalone Python process on an EC2 t3.micro instance, while all sentiment processing is handled by AWS Lambda functions that scale from zero to thousands of concurrent invocations automatically [7]. SQS standard queues provide at-least-once delivery with configurable visibility timeouts to prevent duplicate processing. S3 objects are keyed by ticker symbol and ISO timestamp, supporting efficient range queries for historical aggregations. The Flask backend is containerised and deployed on AWS Elastic Beanstalk, with Auto Scaling configured to maintain sub-2-second API response times under load [12].

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

V. RESULTS AND DISCUSSION

During a live evaluation session, the MarketSense dashboard classified 11 Positive, 6 Negative, and 3 Neutral articles across tracked stocks as shown in Fig. 2. The pie chart confirms a predominantly bullish sentiment distribution consistent with the prevailing tech-sector news cycle.

Fig. 2. MarketSense Sentiment Dashboard — Live Pie Chart and Per-Headline Compound Score Feed

The Scenario Simulator (Fig. 3) correctly identified a Bullish mood with High volatility in response to the prompt “How will strong quarterly earnings affect tech stocks?”, generating an accurate explanation citing institutional buying behaviour and forward guidance sensitivity.

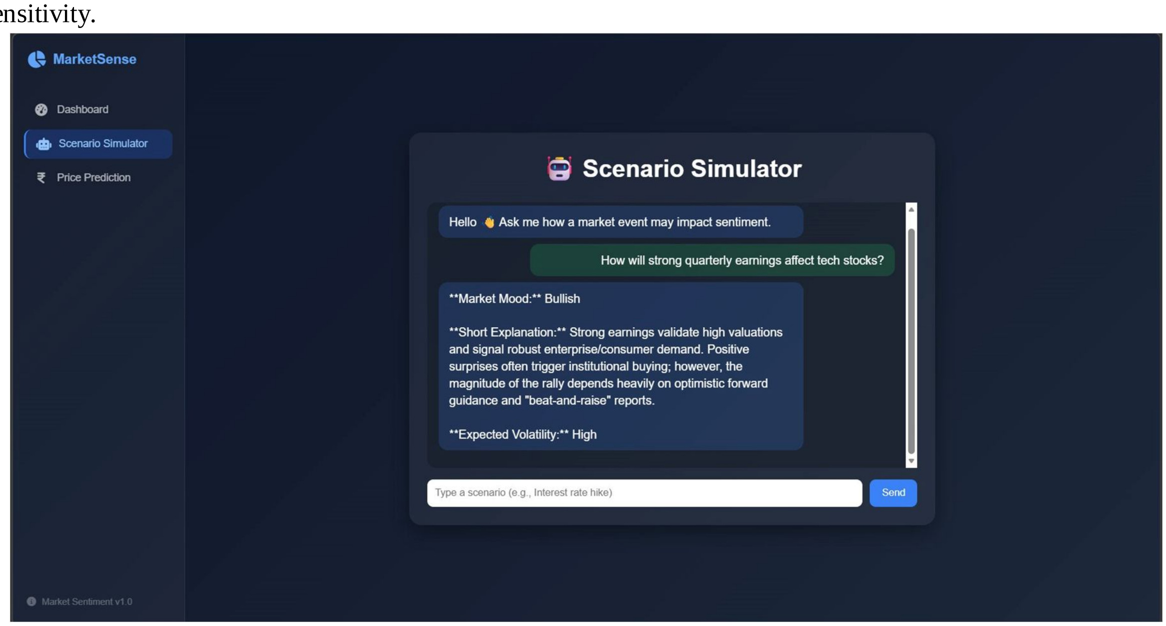


Fig. 3. Scenario Simulator — Gemini AI Response for “Strong Quarterly Earnings” Scenario

The Price Prediction module (Fig. 4) evaluated AMZN at a base price of ₹2000, producing an average positive predicted price of ₹2047.52, an average negative predicted price of ₹1959.84, and a Gemini AI summary identifying a bullish trajectory with cautious hold/watch investor guidance.

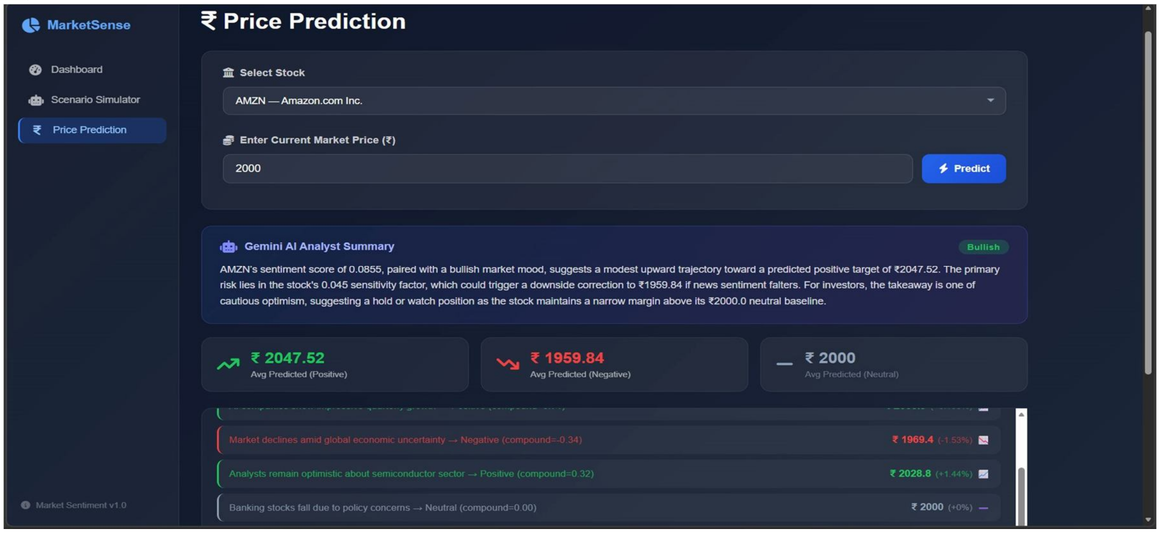


Fig. 4. Price Prediction Module — AMZN at ₹2000 Base Price with Gemini Analyst Summary

End-to-end latency from news publication to S3 availability averaged 18 seconds. Flask API response times averaged 1.4 seconds including Gemini inference, suitable for interactive dashboard usage.

VI. CONCLUSIONS

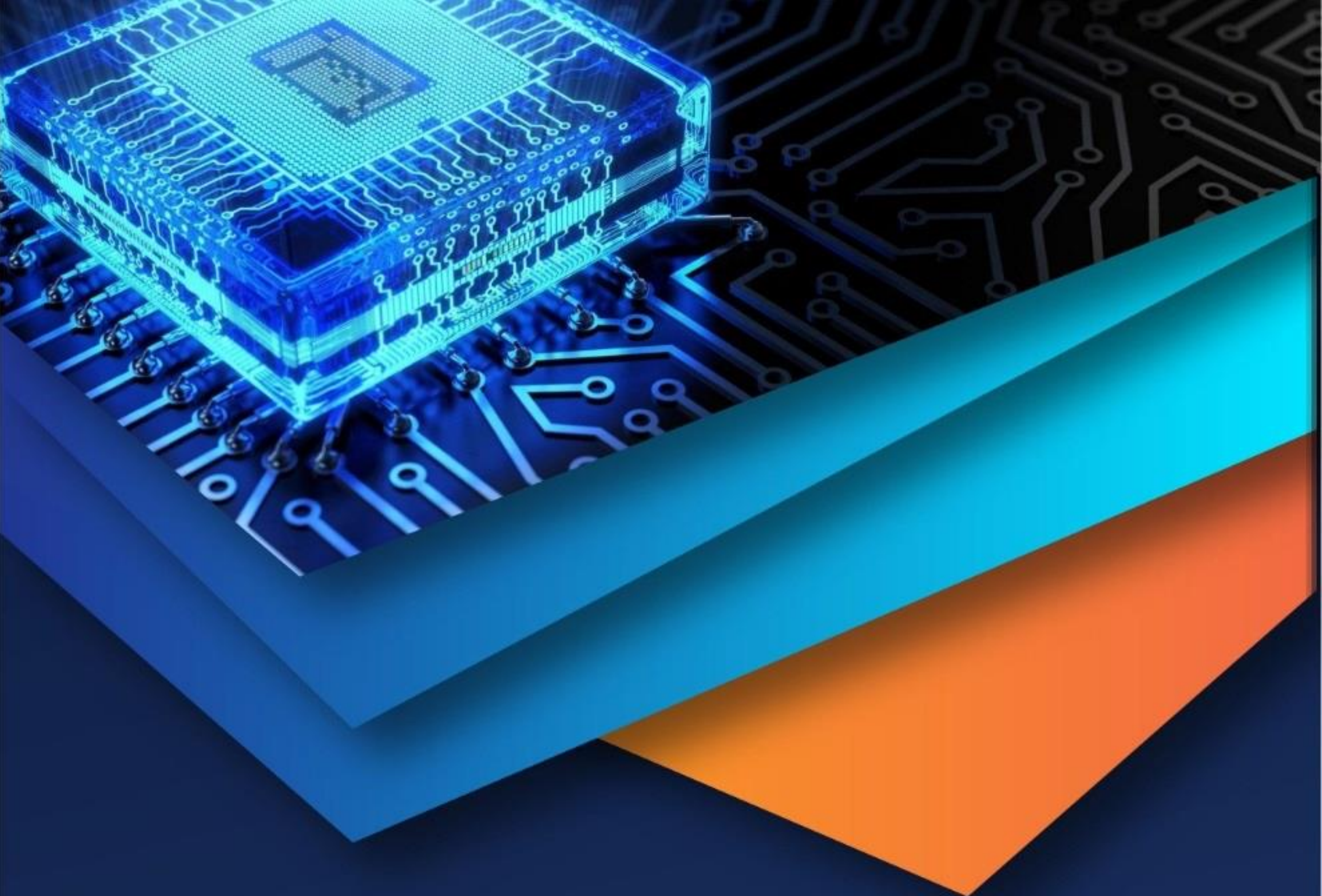
This work demonstrates a scalable, serverless pipeline that transforms raw news headlines into actionable stock sentiment signals with AI-generated interpretations. The modular architecture enables independent scaling of ingestion, processing, and presentation tiers. Future work will incorporate transformer-based models (e.g., FinBERT [11]) for improved domain-specific sentiment accuracy, integrate real-time WebSocket updates to the dashboard, and add historical back-testing against actual stock returns to validate sensitivity factors quantitatively [14].

VII. ACKNOWLEDGMENT

The authors thank their project guide for the invaluable guidance and mentorship throughout this work. The authors also thank the open-source communities behind VADER, Flask, and the AWS SDK for Python (boto3) for providing the foundational tools used in this project.

REFERENCES

[1] T. Loughran and B. McDonald, "When is a Liability not a Liability? Textual Analysis, Dictionaries, and 10-Ks," Journal of Finance, vol. 66, no. 1, pp. 35–65, Feb. 2011.
[2] J. Bollen, H. Mao, and X.-J. Zeng, "Twitter mood predicts the stock market," Journal of Computational Science, vol. 2, no. 1, pp. 1–8, 2011.
[3] P. C. Tetlock, "Giving Content to Investor Sentiment: The Role of Media in the Stock Market," Journal of Finance, vol. 62, no. 3, pp. 1139–1168, 2007.
[4] C. J. Hutto and E. Gilbert, "VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text," in Proc. AAAI Conf. on Web and Social Media (ICWSM), Ann Arbor, MI, 2014.
[5] M. Baldini, M. Bagnato, and D. Bruneo, "Serverless Computing in Cloud-Based IoT Systems," in Proc. IEEE Int. Conf. on Pervasive Computing and Communications Workshops (PerCom), 2017, pp. 577–582.
[6] Google DeepMind, "Gemini: A Family of Highly Capable Multimodal Models," Technical Report, Google, 2023. [Online]. Available: https://deepmind.google/technologies/gemini/
[7] Amazon Web Services, "Amazon SQS Developer Guide." [Online]. Available: https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/
[8] NewsAPI.org, "NewsAPI Documentation." [Online]. Available: https://newsapi.org/docs
[9] S. Mittal and A. Goel, "Stock Prediction Using Twitter Sentiment Analysis," Stanford University CS229 Project Report, 2012. [Online]. Available: https://cs229.stanford.edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis.pdf
[10] R. P. Schumaker and H. Chen, "Textual Analysis of Stock Market Prediction Using Breaking Financial News: The AZFin Text System," ACM Trans. Inf. Syst., vol. 27, no. 2, pp. 12:1–12:19, Feb. 2009.
[11] D. Araci, "FinBERT: Financial Sentiment Analysis with Pre-trained Language Models," arXiv preprint arXiv:1908.10063, 2019.
[12] Amazon Web Services, "AWS Lambda Developer Guide." [Online]. Available: https://docs.aws.amazon.com/lambda/latest/dg/welcome.html
[13] W. Zhang, C. Ding, and M. Zhang, "Earnings Conference Call Summarisation Using Large Language Models," in Proc. ACL Workshop on Natural Language Processing in Finance (FinNLP), 2023, pp. 45–54.
[14] E. F. Fama, "Efficient Capital Markets: A Review of Theory and Empirical Work," Journal of Finance, vol. 25, no. 2, pp. 383–417, May 1970.



10.22214/IJRASET

INDEX COPERNICUS

45.98

IMPACT FACTOR: 7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY