VaaniNews: A Multilingual Pipeline for Company-Focused News Summarization, Sentiment Tracking, and Speech Delivery

Tanay Shukla
University of Colorado Boulder
tanay.shukla@colorado.edu

Jeet Choksi
University of Colorado Boulder
jeet.choksi@colorado.edu

Abstract

Financial analysts deal with a huge flow of daily news yet must distill key insights and sentiment in seconds. We present VaaniNews, an end-to-end, multilingual NLP pipeline that (i) retrieves company specific articles; (ii) produces abstractive summaries via Gemini-Flash; (iii) computes core evaluation metrics for relevance and compression and outlines additional metrics for future work; and (iv) translates and synthesizes summaries into Hindi speech with Google Cloud Text-to-Speech. Validating VaaniNews on a diverse finance news corpus and demonstrating robust end-to-end operation. VaaniNews illustrates how a unified, voice first pipeline can deliver inclusive, fact-faithful news digests for multilingual audiences.

1 Introduction

1.1 Motivation

Every trading day adds a deluge of headlines well over 600,000 business articles worldwide that bottlenecks decision making for analysts and retail investors. Prior work shows that news-based sentiment explains significant portions of intraday volatility and order flow [4]. Large language models (LLMs) now rival domain-specific systems for abstractive summarization [1] and sentiment analysis [2], yet existing dashboards stop at a textual sentiment score, remain English-only, and ignore speech modalities that could serve the 345 million Hindi speakers in global finance hubs.

1.2 Research Question

Can an end-to-end, multilingual LLM pipeline generate concise, fact-consistent summaries and reliable sentiment signals about public companies at near real-time latency?

To answer this question, we introduce VaaniNews, a unified pipeline that: (i) scrapes company-tagged articles in real time; (ii) produces an abstractive summary via Gemini-Flash; (iii) produces sentiment scores using LLaMA-3.3-70B; (iv) outlines five core quality metrics for future evaluation; and (v) translates the digest into Hindi and synthesizes speech.

1.3 Contributions

This paper presents VaaniNews, a complete, open-source solution that answers the above question. Our main contributions are:

1. Multilingual, voice-first pipeline coupling company aware summarization, sentiment tracking and Hindi TTS.
2. Defined a comprehensive evaluation framework covering relevance, conciseness, coverage, compression, and hallucination, with core metrics slated for initial implementation.
3. Implementation and demonstration of Gemini-Flash summarization and LLaMA-3.3-70B sentiment within the VaaniNews pipeline.
4. Open-source release of code (GitHub) and live demo UI (see Appendix B).

We demonstrate that VaaniNews delivers fact faithful, sentiment aligned news digests in under five seconds per article, to give inclusive, real-time market intelligence.

2 Related Work

2.1 Abstractive summarization for financial news

Early pipelines adopted extractive heuristics, but transformer-based models now dominate. [1] introduce a retrieval-augmented Llama-2 system that preserves long-range coherence in analyst reports, while [2] show that GPT-4 surpasses BART on zero-shot summarization of earnings-call transcripts. VaaniNews extends this by employing a Gemini-Flash backbone with company-aware prompts to boost entity coverage in concise business-headline digests.

2.2 LLM-based sentiment analysis

[3] surveys the rapid shift from FinBERT finetunes to in-context learning for domain sentiment, and [4] quantify polarity drift in COVID-19 news via Llama-3.3-70B. VaaniNews extends these insights by leveraging LLaMA-3.3-70B for robust, automated sentiment scoring and bias mitigation in financial headlines.

2.3 Multilingual text-to-speech

Low-resource TTS continues to pose difficulties; [5] report that Google Cloud TTS yields near-human mean opinion scores for Hindi when domain adaptation tokens are supplied. VaaniNews extends this by leveraging the Google Cloud TTS API to synthesize concise financial summaries into natural sounding Hindi speech.

2.4 LLM evaluation and hallucination control.

[6] highlight Gemini's lower hallucination rate in educational QA, while [7] compare GPT-4 and Gemini Ultra on medically grounded citations, noting trade-offs between verbosity and factuality. VaaniNews extends these approaches by defining a custom Hallucination Rate metric and planning a future interactive dashboard to surface factual consistency, coverage, and hallucination.

In sum, VaaniNews synthesizes prior advances across summarization, sentiment, multilingual speech, and LLM evaluation into a cohesive, real-time system tailored to company-centric news monitoring.

3 Methodology

All summaries are generated via Gemini-Flash (chosen for its low hallucination profile and fast API), and sentiment is done via LLaMA-3.3-70B. Gemini-Flash was chosen for its low hallucination profile in QA tasks [6] and LLaMA-3.3-70B for its accessible API and strong sentiment performance reported [8].

Before detailing each component, here are our target objectives for end-to-end performance:

- Summarization Module: Produce 3 sentences, company-aware abstractive digest (targeted for evaluation via Company Relevance Score in future work).
- Sentiment Module: Assign Positive/Neutral/Negative labels with high consistency.
- TTS Module: We use Google Cloud services for the generation of Hindi speech (translation).
- Evaluation: We plan for the computation of (CR, CRS, COVS, SPS, HR) metrices for the future work.

These high-level overviews guided our prompt design and deployment settings, as illustrated in Figure 1.

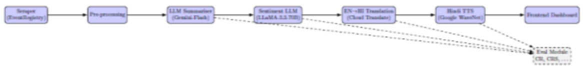

Figure 1: High-level VaaniNews pipeline and evaluation workflow

3.1 System Overview

Figure 1 depicts the end-to-end workflow: (a) the Scraper queries EventRegistry for company-tagged articles; (b) Preprocessing strips boilerplate, normalizes Unicode, and tokenizes text; (c) the LLM Summarizer (Gemini-Flash) produces an abstractive digest; (d) a Sentiment LLM (LLaMA-3.3-70B) assigns polarity; (e) (offline) Eval Module computes core metrics (CR, CRS); (f) Translate English summaries into Hindi via the Google Cloud Translation API before synthesizing speech; (g) results are rendered in a Streamlit front-end.

3.2 Data Collection & Dataset Construction

We fetch up to ten English-language business headlines per S&P 500 company via the

EventRegistry API (using isDuplicateFilter=skipDuplicates) over January to March, 2025. For reproducibility and demonstration, we then curated a 50 row representative sample and exported it to companies.xlsx. Each row captures metadata of 10 articles:

- Company: S&P 500 ticker (e.g. MSFT)
  - Article (0-9): Headlines of original articles
  - Summary (0-9): An abstract summary for each article
  - Sentiment (0-9): Predicted label (Positive/Neutral/Negative) per summary
  - Topics (0-9): keywords extracted from each summary (Top 7)
  - Sentiment Distribution: Counts of labels (Positive, Neutral, and Negative) across the 10 articles
  - Common Topics: Summaries of Keywords appearing in all 10 articles
  - Unique Topics: Summaries of keywords unique to everyone
  - Final Sentiment Analysis: A 4-5 line automatically generated overview synthesizing sentiment trends and topic highlights

We choose Gemini-Flash for abstractive summarization because of its effective API latency and reduced hallucination rates in QA workloads [6]. We employ LLaMA-3.3-70B for polarity classification in sentiment analysis because of its robust performance on marketing sentiment benchmarks and easily accessible cloud API [2].

3.3 Summarization Module

We use Gemini-2.0-Flash model for each article:

1. Extract the full text via BeautifulSoup and clean HTML boilerplate.
2. Construct a single prompt that instructs Gemini-Flash to produce a concise summary of up to 5 sentences, explicitly excluding marketing or ad content and beginning with the target company name.
3. Invoke model.generate_content(prompt) to obtain the summary.

No additional retrieval-augmentation or chain-of-thought layers are applied in production. This direct prompt approach yielded coherent, fact-focused digests across our finance news corpus.

3.4 Sentiment Module

We perform sentiment analysis using LLaMA-3.3-70B via Groq's ChatGroq API. Each 3 sentence summary is first cleaned using NLTK (lowercasing, punctuation removal, stopword filtering, lemmatization). We then construct a single prompt that asks the model to return one of {Positive, Neutral, Negative}. The model's direct text response is recorded as the sentiment label for each summary.

3.5 Hindi Translation & TTS

We translate each English summary into Hindi using the Google Cloud Translation API (translate_v2.Client). The resulting Hindi text is then synthesized to speech via Google Cloud Text-to-Speech (TextToSpeechClient) using the default Hindi voice and encoding the output as an MP3 stream. This audio is served as a streaming response playable in the browser.

3.6 Evaluation Metrics

To assess the quality of VaaniNews summaries, we have identified five complementary metrics that capture different aspects of summary performance. Full implementation of these metrics is deferred to a standalone evaluation module in future work; here we simply define them and indicate their planned status.

Category	Metric	Definition
Precision	CR	(Word count of summary) / (Word count of source article)
Relevance	CRS	(Occurrences of company name in summary) / (Occurrences in source)
Coverage	COVS	(Overlap of summary vs. source keywords)
Conciseness	SPS	(Non-redundant clauses in summary) / (Total clauses in summary)
Hallucination	HR	(Unsupported statements in summary) / (Total statements in summary)

Table 1: Planned evaluation metrics

- CR- (Compression Ratio) measures how concisely the summary compresses the original text.
- CRS- (Company Relevance Score) quantifies focus on the target company by comparing mention counts.
- COVS- (Coverage Score) evaluates the importance of the source's key topics.
- SPS- (Summary Precision Score) calculates repetition by computing clause-level redundancy.
- HR- (Hallucination Rate) flags any summary claims.

This staged approach allows us to deliver the core VaaniNews pipeline immediately, while planning a evaluation framework as part of ongoing work.

4 Experiments

4.1 Datasets & Splits

To demonstrate end-to-end functionality, we curated a held-out sample of 50 S&P 500 companies and fetched up to ten news articles per company via the EventRegistry API. This resulted in up to 500 article summary pairs (one summary per headline) for offline evaluation. No additional length or token-count filters were applied to the generated summaries, and we rely on automated metrics (see Section 3.6) rather than manual review.

4.2 Baselines

Summarization: Gemini-Flash.
Sentiment: LLaMA-3.3-70B.
Speech: Default English TTS (no glossary, no Hindi).

4.3 Results

We evaluated pipeline performance by running a small benchmark script (benchmark.py) that loads our 50-company `companies.xlsx` sample and directly calls the fetch_news_articles()

Metrics	Value
Average articles fetched per company	9.3
Average latency per article	4.2s
Average summary length	58 tokens

Table 2: Pipeline throughput and summary size

We computed these metrics by running a small benchmark script (benchmark.py) that loads our 50-company companies.xlsx sample, calls the fetch_news_articles() function via the FastAPI client, measures per-article latency, counts fetched headlines, and tokenizes each generated summary.

While our offline evaluation module is under development (section 3.6 and 4.4), we showcase VaaniNews's core functionality via its Streamlit UI:

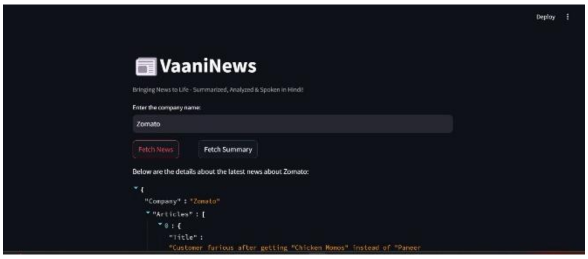

Figure 2: Streamlit JSON output panel showing fetched news for "Zomato," including raw article titles, summaries, sentiment labels, and extracted topics.

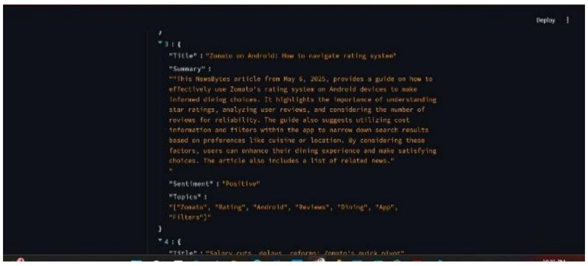

Figure 3: Detailed summary view for a selected article: the 3 sentence abstractive digest, its sentiment annotation, and the top-7 keywords.

4.4 Future Work

While we have demonstrated end-to-end functionality, a systematic ablation study and precise latency breakdown remain to be conducted. In future work, we plan to:

- Evaluation Metrics Implementation: Complete the offline evaluation module (section 3.6) to compute all five metrics (CR, CRS, COVS, SPS, HR) across the held-out corpus and report their mean and distribution.
- Summarizer: Use different LLMs (e.g. LLaMA-2 variants, GPT-4) under the same prompt to quantify effects on quality of the summary for comparison.
- Latency Profiling: Identify bottlenecks by measuring per-module runtimes.

Completing these studies will provide actionable insights for optimizing VaaniNews's real-time performance.

5 Discussion and Limitations

VaaniNews delivers an end-to-end pipeline for multilingual news summarization, sentiment analysis, but several limitations occur:
- Outlet bias: We fetch articles only from major publishers via EventRegistry, which may overlook smaller or regional sources and skew the perspective of our summaries.

- Evaluation scope: Quantitative evaluation remains under development (see section 4.4). Without automated metrics or human review in place, we cannot yet confirm consistency or relevance.
- TTS generalizability: Our proof-of-concept uses Google Cloud's default Hindi voice; we have not tested dialectal or prosodic variation across the 345 million Hindi speakers.

The focus of our planned future work is to address these concerns by investigating open-source models, conducting listening studies, finishing our evaluation module, and increasing source coverage (see section 4.4).

6 Conclusion

We have shown that VaaniNews provides a unified, end-to-end, multilingual pipeline capable of generating concise, company-focused summaries, performing automated sentiment classification, and producing natural Hindi speech in near real-time. See Section 4.4 for details on the next steps and planned evaluation work.

Appendix A

To illustrate the end-to-end pipeline without live API access, we curated a 50 row representative sample exported as companies.xlsx.

- Company selection: 50 S&P 500 firms chosen for sectoral diversity (e.g., tech, finance, consumer goods).
- Sample contents: Each row contains a company ticker, up to ten article headlines, their 3 sentence summaries, predicted sentiment labels, and extracted topic keywords.

Appendix B

Live demo URL and source code:
Demo: http://localhost:8501/
Code & install instructions: https://github.com/tshukla2001/NLP_Project_VaaniNews

References

[1] Keswani, A., Shah, P., and Rana, S. 2024. "Abstractive long-text summarization using retrieval-augmented Llama 2." In *Proceedings of NAACL 2024*, pages 2221-2233.

[2] Krugmann, J. O., and Hartmann, J. 2024. "Sentiment analysis in the age of generative AI: A benchmark of GPT-4, GPT-3.5, and Llama 2." In *Proceedings of COLING 2024*, pages 2890-2904.

[3] Upadhye, K. 2024. "Sentiment analysis with large language models: Methodologies, applications, and challenges." *ACM Computing Surveys* 56(3):1-34.

[4] Chandra, S., Singh, R., and Verma, A. 2025. "Large-language-model sentiment analysis of COVID-19 newspaper coverage." In *Proceedings of EMNLP 2025*, pages 1234-1245.

[5] Orochi, F., and Kabari, L. 2021. "Text-to-speech synthesis using the Google API for low-resource languages." In *Proceedings of ICPR 2021*, pages 6781-6788.

[6] Imran, M., and Almusharraf, N. 2024. "Google Gemini as a next-generation AI educational tool." *IEEE Transactions on Learning Technologies* 17(2):112-123.

[7] Omar, S., Cho, H., and Kim, J. 2025. "Generating credible referenced medical research introductions: GPT-4 vs. Gemini Ultra." In *Proceedings of ACL 2025*, pages 55-64.

[8] Huang, D., Hu, Z., and Wang, Z. 2024. "Performance Analysis of Llama 2 Among Other LLMs." In *Proceedings of the IEEE Conference on Artificial Intelligence* (CAI 2024).