Measuring the Market Impact of Financial News Using Lightweight NLP Models

Baptiste PRAS, Martin LEIVA, Vladimir HERRERA-NATIVI, Javier PEÑA-CASTAÑO

February 12, 2026

Abstract

This project studies how financial news can be transformed into structured market events and linked to subsequent price reactions using lightweight natural language processing models. We construct an event-based dataset by normalizing raw financial disclosures, generating structured impact summaries, extracting issuer entities, and mapping them to tradable tickers. Each news item is converted into a $(\text{date}, \text{ticker})$ event representation, allowing integration with daily market data. A hierarchical summarization architecture is introduced to produce numerically faithful and entity-grounded impact summaries using a compact fine-tuned transformer model. These summaries are evaluated using both ROUGE and an LLM-as-a-judge framework emphasizing factual accuracy and financial relevance. Finally, we retrieve post-announcement market returns to construct an event-based dataset suitable for regression analysis, laying the foundation for quantifying whether structured textual impact signals explain short-term price movements.

1 Introduction

Financial markets react continuously to the arrival of new information. Understanding how textual disclosures translate into measurable price movements remains a central challenge at the intersection of finance and natural language processing (NLP). While recent advances in large language models have demonstrated strong capabilities in text understanding and generation, their computational requirements often limit their practical deployment in structured financial research settings.

This project proposes an end-to-end framework that transforms raw financial disclosures into structured market events suitable for quantitative analysis. Rather than relying on frontier large-scale foundation models requiring extensive computational resources, we design a lightweight architecture capable of producing economically meaningful and numerically reliable representations within a limited training budget.

Our approach begins with rigorous normalization and cleaning of financial news documents to remove structural noise and boilerplate language. We then introduce a hierarchical map–reduce summarization strategy that generates concise impact summaries while preserving numerical precision and correct issuer attribution. In parallel, we implement an entity extraction and ticker resolution pipeline that identifies publicly traded companies mentioned in each article and maps them to valid tradable instruments. This enables the integration of textual disclosures with daily market data and the construction of return-based event targets.

Each article is ultimately converted into a structured representation of the form

$$(\text{date}, \text{ticker}, \text{impact summary}),$$

which serves as the fundamental unit of analysis. This representation bridges unstructured textual information and financial time series data, providing the foundation for an event-driven regression framework aimed at estimating short-horizon market reactions.

The summarization component is explicitly designed to preserve numerical fidelity, maintain entity grounding, and avoid generic filler language. Model performance is evaluated using both lexical similarity metrics (ROUGE) and a finance-oriented LLM-as-a-judge framework that emphasizes factual correctness and professional reliability.

By combining computational efficiency with structured market integration, this work provides a practical and scalable methodology for studying how financial news relates to short-term price dynamics.

2 Dataset Description

The dataset consists of financial news articles with the following fields:

- Announcement date,
- Subject and content text,
- A short textual *Impact* summary made by Mixtral.

Each row corresponds to a single news item.

3 Text Normalization and Preprocessing

Before training any summarization model, raw financial disclosures must be cleaned and standardized. Financial news articles and regulatory filings typically contain boilerplate language, metadata headers, legal disclaimers, multilingual noise, and formatting artifacts that can degrade model performance if left untreated. This stage ensures that the model focuses on economically relevant information.

3.1 Language Filtering

We restrict the dataset to English-language documents. A pre-trained FastText language identification model is used to predict the language of each article. Only texts classified as English with high confidence are retained. This prevents contamination from multilingual disclosures and improves lexical consistency.

3.2 Boilerplate and Structural Cleaning

Financial filings frequently contain recurring structural elements such as:

- "Forward-looking statements"
- "Safe harbor" disclaimers
- "Table of contents"
- Exhibit references
- Legal signatures and formatting blocks

These sections rarely contain new market-relevant information. We therefore remove such patterns using regular expressions targeting common disclosure headers and repetitive legal phrasing. Inline metadata markers are also stripped.

Whitespace normalization is applied to collapse excessive line breaks and spacing into a clean, single-block textual format.

3.3 Minimum Information Constraints

To avoid degenerate training signals, we apply minimum length thresholds:

- Articles shorter than a fixed number of characters are removed.
- Target impact summaries that are too short are discarded.

This ensures that the model learns from sufficiently informative examples.

3.4 Issuer Identification

Each article is associated with a primary entity (issuer). We extract the main subject using heuristic patterns, such as:

- "Company Name (NYSE: TICKER)"
- "Company Name announced..."
- Explicit references to "the Company"

The extracted issuer is stored as a structured variable and later injected into the summarization prompts. This explicit grounding reduces ambiguity in multi-entity announcements and improves attribution consistency.

3.5 Chronological Splitting

To avoid information leakage, the dataset is split chronologically. Earlier documents are used for training and later documents for evaluation. This simulates a realistic forecasting setup where the model is evaluated on unseen future disclosures.

At the end of this stage, we obtain a cleaned and standardized corpus of (date, ticker, text, impact) tuples ready for modeling.

4 Impact Summary Modeling

The objective of this stage is to generate a concise, finance-oriented *impact summary* for each normalized disclosure.

4.1 Motivation

Financial disclosures are often long and exceed the input token limits of standard transformer models. Directly summarizing entire documents can result in truncation, loss of numeric precision, or hallucinated content. To address this, we adopt a hierarchical, map–reduce summarization strategy.

4.2 Chunking Strategy

Each article is divided into overlapping token chunks using a fixed maximum length and stride. This allows the model to process long documents while preserving contextual continuity.

For each chunk, we generate structured bullet-point notes capturing:

- Who performed the action,
- What occurred,

- Key numerical values (amounts, percentages, durations),
- Counterparties and contractual details.

The prompts explicitly instruct the model to preserve technical accounting terms and all numeric information verbatim.

4.3 Map–Reduce Summarization

The summarization proceeds in two stages:

1. **Map stage:** Each chunk is summarized into factual bullet notes.
2. **Reduce stage:** The notes are recursively consolidated into a final 2–4 sentence impact summary.

For very long documents, recursive reduction is applied: notes are grouped, condensed, and then re-summarized to remain within model context limits.

This architecture improves:

- Numerical fidelity,
- Entity grounding,
- Resistance to truncation,
- Information density.

4.4 Training Objective

The model is fine-tuned using supervised learning with reference impact summaries as labels. Each document chunk is associated with the same target summary, allowing the model to learn alignment between local evidence and global impact.

To emphasize the importance of early paragraphs—where key financial information is often disclosed— we apply a higher loss weight to the first chunk of each document.

4.5 Prompt Engineering

The generation prompts enforce strict constraints:

- The first sentence must begin with "The [Issuer] ..."
- No unsupported claims are allowed.
- All numerical values must be preserved.
- Generic filler language is penalized.

Few-shot examples are included to encourage dense, professional analyst-style writing.

4.6 Evaluation Methodology: From ROUGE to LLM-as-a-Judge

Initial Automatic Evaluation with ROUGE. As a first step, we evaluate the model using the ROUGE-L metric, which measures lexical overlap between the generated *impact summary* and the human-written reference summary. In particular, ROUGE-L is based on the Longest Common Subsequence (LCS), capturing how much of the reference summary is reproduced in the correct order.

This metric provides a useful baseline: it allows us to verify that the model is learning to approximate the reference summaries and that performance improves during training. However, ROUGE has important limitations in the financial domain:

- It rewards lexical similarity rather than factual correctness.
- It does not penalize hallucinations if the wording overlaps.
- It does not explicitly evaluate numerical fidelity.
- It cannot detect entity confusion (e.g., misuse of "the company").

In practice, we observed that some earlier models achieved slightly higher ROUGE scores while omitting key financial figures or introducing vague filler language. This highlighted the need for a more semantically grounded evaluation framework.

Final Evaluation with LLM-as-a-Judge. To address these limitations, we introduce a second evaluation layer based on a local instruction-tuned large language model acting as an automatic financial auditor. For each pair (source text, generated summary), the judge evaluates the summary across six dimensions:

- **Accuracy** (factual correctness and absence of hallucination),
- **Issuer Grounding** (correct attribution of actions to the appropriate entity),
- **Numeric Fidelity** (preservation of key figures and dates),
- **Coverage** (capture of the main event and its impact),
- **Conciseness** (information density and absence of repetition),
- **Anti-Filler Professionalism** (absence of clichés, investor advice, and generic language).

Each category is scored on a 0–5 scale. The judge is explicitly instructed to penalize unsupported claims, vague filler phrases, entity confusion, and missing key financial numbers.

Quantitative Results. On the full evaluation set, the average scores obtained by the final model are:

- Accuracy: 3.00
- Issuer Grounding: 3.17
- Numeric Fidelity: 3.01
- Coverage: 3.07
- Conciseness: 2.08
- Anti-Filler Professionalism: 2.38

Interpretation. The model achieves solid mid-range performance (around 3/5) on factual accuracy, grounding, numeric fidelity, and coverage. This suggests that:

- Most summaries correctly capture the core event.
- Key figures are often preserved.
- Major hallucinations are relatively limited.

However, conciseness and anti-filler professionalism remain weaker dimensions. This confirms qualitative observations that some summaries still contain verbosity, mild repetition, or generic investor-oriented language.

While training a frontier LLM such as Mixtral may require several thousands of GPU-hours, our approach reaches competitive results with roughly two GPU-hours of training, highlighting its strong computational efficiency.

Why We Prioritize the LLM Judge. Although ROUGE provides a useful baseline signal during development, the final model selection is guided primarily by the LLM-as-a-judge evaluation. In financial contexts, numerical fidelity, correct entity attribution, and professional tone are more critical than surface-level lexical overlap.

Therefore, a slight decrease in ROUGE (e.g., from 0.29 to 0.27) is acceptable when accompanied by improved factual precision and better preservation of financial details. The evaluation framework thus prioritizes semantic correctness and investor usefulness over textual similarity alone.

5 Entity Extraction and Ticker Resolution

The objective of this stage is to identify the publicly traded companies mentioned in each financial disclosure and map them to valid ticker symbols.

By extracting issuer entities through Named Entity Recognition and resolving them to tradable instruments, we construct structured pairs of the form:

$$(\text{date}, \text{ticker})$$

These pairs constitute the bridge between textual information and market data. They can later be merged with the generated impact summaries and matched with price series to form the input of a regression framework aimed at quantifying short-term market reactions to news.

Our implementation is structured into four components: (1) Named Entity Recognition, (2) Entity-to-ticker linking, (3) Data filtering via ticker coverage, and (4) Chronological train–test splitting.

5.1 Named Entity Recognition

We extract company entities from both the subject line and the content of each article using an existing pre-trained Named Entity Recognition (NER) pipeline based on BERT.

Specifically, we rely on a Hugging Face Transformers pipeline using a BERT-based English NER model fine-tuned on standard entity recognition benchmarks. This allows us to identify organization entities (ORG tags) without training a custom model from scratch.

Before applying NER, we perform additional filtering:

- Removal of special characters and non-alphanumeric artifacts,
- Normalization of whitespace and punctuation,

- Elimination of residual structural markers.

This lightweight preprocessing was important to improve entity boundary detection and reducing noise-induced fragmentation.

Importantly, our approach remains computationally frugal. Rather than fine-tuning a large language model for joint entity extraction and linking, we rely on a compact pre-trained BERT encoder used in inference mode only. This significantly reduces computational cost compared to training large-scale sequence-to-sequence models or end-to-end generative entity-linking systems.

5.2 Entity-to-Ticker Linking

Once company names are extracted, we map them to tradable tickers through a controlled entity linking step.

We use the `yfinance` API to retrieve metadata for publicly listed firms, including official company names and ticker symbols. Extracted organization entities are matched against this reference set using string normalization and similarity-based matching.

This step ensures that each identified "entity" is associated with a valid and tradable ticker symbol. Ambiguous or unmatched entities are discarded to preserve data reliability.

5.3 Data Filtering via Ticker Coverage

Not all extracted entities can be confidently mapped to a listed instrument. To ensure consistency in downstream market analysis, we apply a coverage threshold on ticker frequency.

Only tickers appearing with sufficient frequency are retained. After applying this threshold, approximately 30% of the original dataset is preserved, corresponding to articles with reliable ticker resolution.

Although this filtering reduces dataset size, it significantly improves the structural quality of the resulting event representation and avoids introducing noise from poorly resolved entities.

5.4 Chronological Train–Test Split

To prevent information leakage in subsequent predictive modeling, we perform a chronological split of the dataset.

Events are ordered by announcement date, with earlier observations assigned to the training set and later observations reserved for the test set. This setup mirrors a realistic financial forecasting scenario, where future market reactions must be predicted using only past information.

After this stage, each retained article is transformed into a structured and time-consistent event representation suitable for integration with market price data.

6 Market Impact Modeling and Current Status

The final stage of the project focuses on quantifying the relationship between structured news events and subsequent market reactions.
Using the event representation constructed in previous sections,

$$(\text{date}, \text{ticker}, \text{impact summary}),$$

we retrieve daily market price data surrounding each announcement date in order to construct return-based outcome variables. For each event, we collect closing prices before and after the disclosure to compute short-horizon returns, forming the basis for an event-driven empirical analysis.

This structured integration of textual signals and financial time series enables the construction of a regression framework aimed at estimating whether features derived from impact summaries are associated with contemporaneous or subsequent price movements.

At the current stage of the project, the event normalization, summarization, and ticker-linking components are fully operational, and market data retrieval has been implemented and validated. The predictive modeling layer (intended to measure the explanatory or predictive power of textual impact signals for abnormal returns) remains under development.

Priority was given to ensuring robust textual preprocessing, reliable entity resolution, and numerically faithful summarization before introducing econometric modeling.

7 Conclusion

This project develops an end-to-end framework for converting unstructured financial disclosures into structured event representations suitable for quantitative market analysis. The proposed architecture integrates text normalization, hierarchical impact summarization, entity extraction, ticker resolution, and market data retrieval within a unified and computationally efficient pipeline.

On the textual side, we introduced a lightweight map–reduce summarization model designed to preserve numerical fidelity, maintain correct issuer attribution, and limit generic or speculative language. While traditional lexical metrics such as ROUGE provide a useful baseline, we complement them with a finance-oriented LLM-as-a-judge evaluation framework that emphasizes factual accuracy, numeric consistency, and professional tone. The results indicate solid mid-range performance in core reliability dimensions, while also revealing opportunities for improved conciseness and stylistic discipline.

On the structured data side, we implemented a robust entity extraction and ticker-linking procedure that converts each article into a $(\text{date}, \text{ticker})$ representation. This structured mapping enables the systematic integration of textual disclosures with financial time series data and establishes the empirical foundation for event-driven regression analysis of short-horizon market reactions. Although the full predictive modeling component remains under development, the current system successfully operationalizes the linkage between news content and tradable instruments.

Importantly, the entire framework relies on compact fine-tuned transformer models that can be trained within a modest computational budget. This demonstrates that meaningful financial NLP applications can be developed without reliance on frontier-scale foundation models, making the approach accessible and reproducible under realistic research constraints.

Several limitations remain. Entity resolution may be imperfect in multi-company disclosures. Market reactions are influenced by numerous exogenous factors beyond textual information, limiting purely text-based predictability. Moreover, further improvements in summarization density and stylistic rigor may enhance downstream econometric performance.

Overall, this work provides a structured and computationally efficient foundation for linking financial news to market dynamics. It opens the way for future extensions including refined entity disambiguation, richer textual feature engineering, and formal statistical evaluation of event-driven return patterns.