LLM and Graphs for Financial Texts

Aamir Samdani, Brandon Zhang
EIB Advisors: Giuseppe Bonavolontá, Oleg Reichmann
Columbia Faculty Advisors: Dr. Ali Hirsa, Miao Wang

COLUMBIA | ENGINEERING
The Fu Foundation School of Engineering and Applied Science

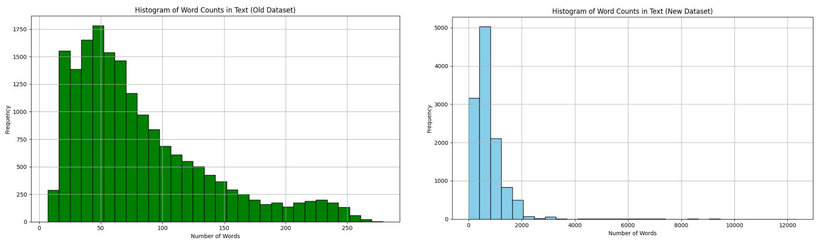
European Investment Bank

Abstract

Global Financial Markets are profoundly shaped by the continuous flow of news-spanning from M&A deals and technological breakthroughs to geopolitical events and macroeconomic developments. Yet, extracting meaningful and actionable insights from this stream of information remains a central challenge in financial analytics. Traditional natural language processing (NLP) methods–such as sentiment scoring and named entity recognition (NER)--often fall short as they fail to capture the nuanced relational context embedded in complex financial discourse.

To address this challenge, our project leverages large language models (LLMs) to extract structured triplets of the form *(Entity 1, Relation 1, Relationship Type, Relation 2, Entity 2)* from various forms of financial texts. These triplets form the foundation of dynamic knowledge graph databases (DKDGs), which evolve over time to reflect shifts in entities and relationships across the financial domain. Building on this framework, we iteratively engineered prompts for triplet generation, enriched our input data, developed custom evaluation metrics for the triplets – both handcrafted and LLM based – which were then integrated into A.I. agent workflows for autonomously improving triplet quality.

Dataset Expansion



- Old dataset: each sample consisted of only the Yahoo Finance article headline concatenated with a short description of the article (~19k samples).
- New dataset: collected ~11k samples with **full** article content from Yahoo Finance, collecting more contextual and information-dense content.
- Average length of new dataset ~743 words per sample while the average length of the old dataset ~82 words, representing on average over **9x increase per sample** .
- Histogram shows the distribution of the number of words in each sample for the old dataset vs. the new dataset. Note that the x-axis for the left (old dataset) is from 0-250, while the right (new dataset) is from 0-12000.

Triplet Generation Prompt Engineering

Aspect	Spring 2025 Promp t	Summer 2025 Prompt Engineering
Output Format	5-Tuple Pythonic syntax with the following structure: `('entity1', 'type1', 'relation', 'entity2', 'type2')`	6-Tuple Pythonic syntax with the following structure: `('entity1', 'type1', 'relation', 'relation category type', 'entity2', 'type2')`
Entity Definitions	-11 Entity types -Entity definition(s) in incorrect and/or smaller dimension set (e.g.inflation + recession listed as concept as opposed to economic indicator)	-14 Entity Types -Definitions amended, refined, and expanded -New entity types include: -DERIVATIVE -EQUITIES -MACRO_INDICATOR
Few-Shot Examples	-5 Examples -No relationship categorization -Certain examples had erroneous entity types and/or classifications (e.g. "Person" labeled as an entity type, but not in definition taxonomy).	-8 Examples -Errors amended -Implementation of Relationship Categories: -GFMK -SSI -GMM

Handcrafted Metrics

To assess the quality of our triplets, we developed unsupervised, lightweight metrics across a variety of dimensions:

Coverage Quality: Measures how well triplets represent the article's key content, using MiniLM-L6-v2 embeddings and weighted cosine similarity scoring (inspired by Semantic Textual Similarity):

$\{s_1, s_2, \ldots, s_n\}$: set of embedded article sentences
$\{t_1, t_2, \ldots, t_m\}$: set of embedded triplets
w_i : importance weight associated with sentence s_i
$\cos(s_i, t_j)$: cosine similarity between sentence s_i and triplet t_j

$$\sum_{i=1}^{n} w_i \cdot \max_{1 \le j \le m} \left(\frac{\cos(s_i, t_j) + 1}{2} \right)$$

Semantic Uniqueness: Measures whether triplets capture distinct information, grouping together cosine similar triplets (if a given triplet is similar enough to a triplet in an existing group, add it to the group. *Score = # groups / # triplets*).

Semantic Validity: Assesses logical soundness by mapping triplets to an ontology with domain/range restrictions. *Score = # valid triplets / # triplets*

Entropy Ratio: Compares the information density of triplets against the source article, checking whether outputs are more focused and predictable. Ideally want ER < 1.

$$H_{article} = \sum_{i=1}^{n} p(w_i) \log_2 p(w_i)$$

$$ER = \frac{H_{triplet}}{H_{article}}$$

$w_i = $ *unique tokens in the article*
$p(w_i) =$ Probability Distribution (Frequency) of all tokens over article text

Judge LLM Workflow

- We introduced independent **Judge LLMs** that assess triplets across similar dimensions as our handcrafted metrics: coverage, uniqueness, semantic validity and consistency.
- One Judge LLM per metric – prompt ingests generated triplets, article, and is asked to assign a score from 0.0-1.0 for the metric in question, as well as an explanation of its score for **enhanced interpretability** .
- Example criteria for Coverage LLM prompt: - `"Do the triplets capture the article's most important, financially actionable content?"`

Metric Evaluation

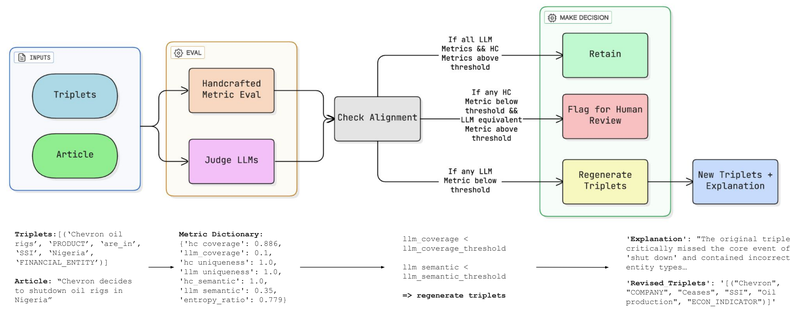
Model	Gemini 2.5 Flash	Amazon Nova Pro
Handcrafted Coverage Score	0.823	0.799
LLM Coverage Score	0.893	0.735
Handcrafted Uniqueness Score	0.584	0.768
LLM Uniqueness Score	0.941	0.974
Handcrafted Semantic Score	0.890	0.775
LLM Semantic Score	0.985	0.954
Entropy Ratio	0.735	0.679

Table 1: Evaluation of all metrics, averaged across 10 samples for each model. Note that the LLM semantic score is the average of its semantic validity score and consistency score.

We compared triplets generated by Gemini 2.5 Flash and Amazon Nova Pro. Gemini 2.5 Flash scored higher on coverage than Amazon Nova Pro, especially in the LLM-generated coverage scores. This matched what we observed qualitatively, especially for longer articles. Both models were similar in Uniqueness and Semantic Validity, though Gemini's handcrafted uniqueness score was notably lower relative to its LLM counterpart. This discrepancy may be due to overaggressive triplet grouping in our handcrafted uniqueness score.

Triplet Regeneration Pipeline

With our metrics as the evaluation criteria, we developed a workflow that would regenerate triplets if any LLM metric was below a defined threshold. Triplets were regenerated by a third LLM (Gemini-2.5-Flash), passing in the original inputs and Judge and Handcrafted metric evaluation. Judge score explanations were included for **targeted and individualized** prompting. If any handcrafted metric was below the threshold but the LLM equivalent was above, then the triplets were flagged for human review.



The above demonstrates the pipeline in action: we provide a sentence as the article as well as a suboptimal triplet that misses the core meaning, and demonstrate how we can autonomously generate a triplet that better captures the essence of the sentence (not perfect, e.g. misses Nigeria, but better)

Final Thoughts

We developed a pipeline that combines LLM-based triplet generation with evaluation and regeneration, supported by both Judge LLMs and lightweight handcrafted metrics. Our results show the feasibility of integrating generation, evaluation, and self-correction into a cohesive workflow, with future work aimed at further prompt improvements, strengthening metrics and advancing dynamic knowledge graph construction.