FinReflectKG - EvalBench: Benchmarking Financial KG with Multi-Dimensional Evaluation

Fabrizio Dimino
Domyn
New York, US
fabrizio.dimino@domyn.com

Abhinav Arun
Domyn
New York, US
abhinav.arun@domyn.com

Bhaskarjit Sarmah
Domyn
Gurugram, India
bhaskarjit.sarmah@domyn.com

Stefano Pasquali
Domyn
New York, US
stefano.pasquali@domyn.com

Abstract

Large language models (LLMs) are increasingly being used to extract structured knowledge from unstructured financial text. Although prior studies have explored various extraction methods, there is no universal benchmark or unified evaluation framework for the construction of financial knowledge graphs (KG). We introduce **FinReflectKG - EvalBench**, a benchmark and evaluation framework for KG extraction from SEC 10-K filings. Building on the agentic and holistic evaluation principles of FinReflectKG - a financial KG linking audited triples to source chunks from S&P 100 filings and supporting single-pass, multi-pass, and reflection-agent-based extraction modes - EvalBench implements a deterministic *commit-then-justify* judging protocol with explicit bias controls, mitigating position effects, leniency, verbosity and world-knowledge reliance. Each candidate triple is evaluated with binary judgments of faithfulness, precision, and relevance, while comprehensiveness is assessed on a three-level ordinal scale (good, partial, bad) at the chunk level. Our findings suggest that, when equipped with explicit bias controls, LLM-as-Judge protocols provide a reliable and cost-efficient alternative to human annotation, while also enabling structured error analysis. Reflection-based extraction emerges as the superior approach, achieving best performance in comprehensiveness, precision, and relevance, while single-pass extraction maintains the highest faithfulness. By aggregating these complementary dimensions, FinReflectKG - EvalBench enables fine-grained benchmarking and bias-aware evaluation, advancing transparency and governance in financial AI applications.

Keywords: Knowledge Graphs, LLM-as-a-Judge, Evaluation Benchmarks, SEC Filings, Large Language Models, Natural Language Processing, Financial AI

1 Background and Motivation

Large language models (LLMs) are increasingly used in finance across diverse tasks, including the extraction of information from long and unstructured documents. A common target representation for such extracted knowledge is the KnowledgeGraph (KG), which organizes information into structured triples and supports downstream applications such as compliance monitoring, risk management, and large-scale financial analytics [1, 2]. Although recent advances in LLM have improved triple extraction through multi-turn prompting and reflection [3], the evaluation of extracted triples remains a critical challenge [4]. Since financial KGs are designed to support high-stakes applica-

tions such as investment research, credit risk assessment, and portfolio decisions, the lack of reliable validation for LLM-generated triples poses significant risks.

Previous works have explored LLM-as-Judge as a scalable alternative to costly human annotation [5, 6]. However, LLM judges are susceptible to prompt sensitivity and order-based position effects [7], often favoring answers presented earlier, regardless of quality. They also tend to be overly lenient [8] and exhibit stylistic biases that reward persuasive phrasing over factual correctness [9], which undermines the robustness and interpretability of LLM-based evaluations [10, 11]. Recent studies on prompt design show that instruction-only prompting reduces reasoning drift and improves judgment consistency [12–14], whereas encouraging detailed chain-of-thought often produces the opposite effect, leading to overthinking and topic drift [15]. Moreover, evidence suggests that few-shot prompting further stabilizes LLM-as-Judge behavior and increases agreement with human annotations [16]. . Although several open-source toolkits (e.g., `DeepEval`, `Ragas`, `TruLens`, `LangSmith`) incorporate LLM-as-Judge components for tasks such as QA and summarization [17–20], rigorous, bias-aware evaluation protocols tailored to triple-level KG extraction, particularly in finance, remain underspecified.

To address these challenges, we introduce **FinReflectKG - EvalBench**, a benchmark for financial KG extraction built on SEC 10-K filings. FinReflectKG - EvalBench extends the agentic principles of *FinReflectKG* [21] by combining schema-aware extraction with a conservative evaluation protocol. To the best of our knowledge, this is the first systematic benchmark for financial KG extraction that integrates various extraction strategies (single pass, multi pass and reflection) and a rigorous evaluation framework.

This paper makes the following key contributions:

1. We present **FinReflectKG - EvalBench**, the first benchmark for financial KG extraction from SEC filings with a reproducible and bias-aware evaluation framework.
2. We provide a comparative evaluation of extraction modes (single-pass, multi-pass, and reflection) across four complementary dimensions: *faithfulness*, *precision*, *relevance*, and *comprehensiveness*.

2 Methodology

We construct **FinReflectKG - EvalBench** on the corpus of U.S. SEC Form 10-K filings from S&P 100 companies for fiscal year 2024. Formally, let $\mathcal{D}$ denote this corpus where each document $d \in \mathcal{D}$ is segmented into text spans $\mathcal{X}_d = \{x_{d,1}, \ldots, x_{d,K_d}\}$ using a deterministic, structure-aware chunking scheme tailored to financial filings. An information extractor E maps each span to a set of candidate triples,

$$T_x = E(x) \subseteq \mathcal{T}, \qquad t = (s, r, o) \in \mathcal{T},$$

where s and o are subject and object entity mentions, and $r \in \mathcal{R}$ is a relation from a predefined financial vocabulary.

We evaluate extraction outputs using an LLM-as-Judge J, instantiated with the Qwen3-32B model and configured for deterministic decoding (temperature $= 0.0$). As shown in Figure 1, and following the *commit-then-justify* paradigm [22], the judge first produces a structured verdict and then a concise justification (up to 15 words). To further support error analysis, we introduce a warning signal that highlights extraction errors and provides actionable correction paths. These signals can also be leveraged within a feedback loop for iterative self-improvement.

To ensure reliable and reproducible judgments, we enforce strict bias controls. First, we adopt a principle of conservatism: whenever the evidence is ambiguous, the judge defaults to a negative decision (0), thereby mitigating leniency bias. Second, we enforce locality, strictly prohibiting the use of world knowledge or inferences beyond the provided text. Third, we guarantee position independence by instructing the judge not to let the order or placement of sentences influence its verdicts. Finally, we ensure verbosity independence, so that the length or surface form of a candidate triple does not bias the evaluation outcome. In addition, we improve consistency and calibration by including few-shot examples for each evaluation criterion in the judge's prompt. Concrete examples are provided in the Appendix.

```
role: "Knowledge Graph Evaluator"
task: "Determine if the triplet is [evaluation criterion]
       with respect to the source text context"
instructions:
  Decision rule:
    - Return 1 if [criterion satisfied],
    - Return 0 if [criterion not satisfied].
  Bias controls:
    - Be conservative: when uncertain, return 0 (leniency bias).
    - Do NOT infer or add information beyond the text (world knowledge bias).
    - Do NOT let sentence position in the source text affect the decision (
    position bias).
    - Do NOT let the length of the triplets affect the decision (verbosity bias
    ).
  Reasoning vs Warning:
    - Reasoning: Brief explanation of the verdict (up to 15 words).
    - Warning: actionable tag(s) for error type; Do NOT duplicate reasoning;
    use empty string if no actionable issue.
  Output policy:
    - Valid JSON array only, single line.
    - Each item: {"verdict":0|1,"reasoning":"...","warning":"..."}

    Examples:
    ...
```

Figure 1: System prompt design

The three extraction modes define different approaches to constructing knowledge graph triples. Single-pass uses a single LLM for both extraction and normalization. Multi-pass splits the task between two LLMs: one extracts triples, while the other normalizes them according to rubric parameters. Reflection employs an agentic, iterative workflow, where extraction and feedback loops refine triples until inconsistencies are resolved or a maximum iteration limit is reached. Across the three extraction modes, the task is to evaluate the quality of the candidate triples produced for each span x with candidate set T_x. Evaluation is conducted along four complementary dimensions. First, **faithfulness** F measures whether the content of a triple is factually grounded in the source text, without relying on world knowledge or bridging inferences. Second, **precision** P assesses the clarity and specificity of triples, penalizing generic placeholders (e.g., "Company") and imprecise expressions of quantities or dates. Third, **relevance** R checks whether the triple contributes directly to the main theme of the source span rather than introducing tangential information. Finally, **comprehensiveness** C is measured on a three-level ordinal scale (good, partial, bad) and it evaluates coverage at the chunk level, scoring how well the set of triples represents all atomic core facts.

To aggregate results across the corpus, let $\mathcal{X} = \bigcup_{d \in \mathcal{D}} \mathcal{X}_d$ and $\mathcal{T}_{\text{all}} = \bigcup_{x \in \mathcal{X}} T_x$. Local binary metrics (faithfulness, precision, relevance) are micro-averaged:

$$\bar{F} = \frac{1}{|\mathcal{T}_{\text{all}}|} \sum_{(x,t)} F(x,t), \qquad \bar{P},\ \bar{R} \text{ analogously.}$$

Comprehensiveness is macro-averaged across spans:

$$\bar{C} = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} C(x).$$

3 Results

Evaluation of triple generation often treats faithfulness, relevance, precision, and comprehensiveness as separate dimensions. While such a decomposition is informative, we argue that these criteria must ultimately be interpreted jointly in order to capture the full spectrum of trade-offs involved in knowledge graph construction.

Table 1: Evaluation results across extraction modes

	Single Pass	**Multi Pass**	**Reflection**
Comprehensiveness	62.60	62.41	**72.01**
Faithfulness	**87.25**	78.73	83.40
Precision	56.06	58.01	**59.49**
Relevance	91.46	82.64	**92.52**

As reported in Table 1, performance varies markedly across extraction modes. The reflection mode achieves superior results in comprehensiveness, precision, and relevance, whereas the single-pass mode yields the highest score in faithfulness. This pattern is consistent with the intuition that reflection, by design, generates a larger set of triples per chunk, thereby capturing a broader range of atomic core facts. From the perspective of knowledge graph construction, this ability to recover a wider coverage of facts is highly desirable, as it directly impacts the downstream utility of the graph.

At the same time, the relative decline in faithfulness observed for reflection suggests an inherent trade-off: expanding coverage increases the risk of generating triples that extend beyond the strict boundaries of the source text. In contrast, the single-pass approach, though less comprehensive, remains more conservative and better aligned with the original text.

Turning to precision, we observe that absolute scores remain relatively modest across all modes, indicating the need for further improvements. Nonetheless, reflection achieves the best performance in this dimension, suggesting that iterative reasoning can modestly improve structural accuracy. By contrast, relevance yields consistently higher values across modes, with reflection leading, which indicates that most generated triples-despite occasional issues of faithfulness or precision-remain topically aligned with the source text.

Together, these results highlight the complementary strengths and weaknesses of different extraction strategies. Reflection provides the most balanced coverage, excelling in breadth, topical alignment, and structural accuracy, though it requires further calibration to match the strict factual reliability of single-pass generation. These findings underscore the importance of multi-dimensional evaluation, as no single metric alone captures the full spectrum of trade-offs inherent to financial KG construction.

4 Conclusions

In this work, we introduced **FinReflectKG - EvalBench**, the first benchmark for financial knowledge graph extraction from SEC 10-K filings with a reproducible, bias-aware evaluation framework. By integrating schema-aware extraction with a conservative LLM-as-Judge protocol, we provide a principled way to assess triple quality across multiple extraction modes. Our analysis highlights several key insights. Reflection emerges as the most balanced modes, achieving superior comprehensiveness, precision, and relevance, while single-pass remains the most faithful to the source text. This trade-off underscores the necessity of multi-dimensional evaluation, as no single metric alone can fully capture extraction quality.

Beyond reporting scores, FinReflectKG - EvalBench contributes a methodological advancement through explicit bias controls, a commit-then-justify judgment protocol, and the introduction of warning signals that enable actionable error analysis. These design elements not only improve reliability but also open the door to self-improving extraction pipelines, where diagnostic signals guide iterative refinement.

Looking forward, our benchmark provides a foundation for advancing financial KG research in several directions. Future work may extend coverage beyond the S&P 100 universe and incorporate diverse categories of financial documents. By establishing transparent, reproducible, and bias-aware evaluation standards, FinReflectKG - EvalBench aims to accelerate progress toward trustworthy financial KGs that support downstream tasks in compliance, risk management, and large-scale financial analytics.

References

[1] Yifan Li and Kevin M Passino. Findkg: Dynamic financial knowledge graph construction from news with large language models. *arXiv preprint arXiv:2402.02413*, 2024. URL https://arxiv.org/abs/2402.02413.

[2] Shilong Li, Yancheng He, Hangyu Guo, Xingyuan Bu, Ge Bai, Jie Liu, Jiaheng Liu, Xingwei Qu, Yangguang Li, Wanli Ouyang, et al. Graphreader: Building graph-based agent to enhance long-context abilities of large language models. *arXiv preprint arXiv:2406.14550*, 2024.

[3] Bowen Zhang and Harold Soh. Extract, define, canonicalize: An llm-based framework for knowledge graph construction. *arXiv preprint arXiv:2404.03868*, 2024. URL https://arxiv.org/pdf/2404.03868.

[4] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 15(3):1–45, 2024.

[5] Haitao Li, Qian Dong, Junjie Chen, Huixue Su, Yujia Zhou, Qingyao Ai, Ziyi Ye, and Yiqun Liu. Llms-as-judges: A comprehensive survey on llm-based evaluation methods. 2024.

[6] Evidently AI. Llm-as-a-judge: a complete guide to using llms for evaluations, 2025. Online guide.

[7] Lin Shi, Chiyu Ma, Wenhua Liang, Weicheng Ma, and Soroush Vosoughi. Judging the judges: A systematic study of position bias in llm-as-a-judge. In *arXiv*, 2024.

[8] Jiayi Ye, Yanbo Wang, Yue Huang, Dongping Chen, Qihui Zhang, Nuno Moniz, Tian Gao, Werner Geyer, Chao Huang, Pin-Yu Chen, Nitesh V Chawla, and Xiangliang Zhang. Justice or prejudice? quantifying biases in llm-as-a-judge. In *arXiv*, 2024.

[9] Yerin Hwang, Dongryeol Lee, Taegwan Kang, Yongil Kim, and Kyomin Jung. Can you trick the grader? adversarial persuasion of llm judges. In *arXiv*, 2025.

[10] Norah Alzahrani, Hisham Alyahya, Yazeed Alnumay, Sultan Alrashed, Shaykhah Alsubaie, Yousef Almushayqih, Faisal Mirza, Nouf Alotaibi, Nora Al-Twairesh, Areeb Alowisheq, et al. When benchmarks are targets: Revealing the sensitivity of large language model leaderboards. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13787–13805, 2024.

[11] Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not robust multiple choice selectors. *arXiv preprint arXiv:2309.03882*, 2023.

[12] Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, et al. Do not think that much for 2+ 3=? on the overthinking of o1-like llms. *arXiv preprint arXiv:2412.21187*, 2024.

[13] Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Hanjie Chen, et al. Stop overthinking: A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*, 2025.

[14] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022. doi: 10.48550/arXiv.2201.11903. URL https://arxiv.org/abs/2201.11903.

[15] Dimitris Vamvourellis and Dhagash Mehta. Reasoning or overthinking: Evaluating large language models on financial sentiment analysis. *arXiv preprint arXiv:2506.04574*, 2025.

[16] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. URL https://arxiv.org/abs/2306.05685.

[17] Confident AI. Deepeval: An open-source evaluation framework for llm applications. https://github.com/confident-ai/deepeval, 2024. Accessed: 2025-08-20.

[18] Exploding Gradients. Ragas: Evaluation framework for retrieval-augmented generation. https://github.com/explodinggradients/ragas, 2023. Accessed: 2025-08-20.

[19] Truera. Trulens: Evaluation and monitoring for llm applications. https://github.com/truera/trulens, 2023. Accessed: 2025-08-20.

[20] LangChain. Langsmith: Evaluation, monitoring, and tracing for llm applications. https://smith.langchain.com/, 2023. Accessed: 2025-08-20.

[21] Abhinav Arun, Fabrizio Dimino, Tejas Prakash Agarwal, Bhaskarjit Sarmah, and Stefano Pasquali. Finreflectkg: Agentic construction and evaluation of financial knowledge graphs, 2025. URL https://arxiv.org/abs/2508.17906.

[22] Alejandro Lopez-Lira and Yuehua Tang. Can chatgpt forecast stock price movements? return predictability and large language models. *arXiv preprint arXiv:2304.07619*, 2023.

5 Appendix

Faithfulness Examples

Source Text: "OpenAI signed a $1B deal with Microsoft in 2024 in Texas"

Supported (1):
Triplet: ["OpenAI","Signed","$1B deal"]
Verdict: 1 (Supported), *Reasoning:* Triplet grounded in source text, *Warning:* None

Not Supported (0):
Triplet: ["OpenAI","Buy","Microsoft"]
Verdict: 0 (Not Supported), *Reasoning:* Triplet not grounded in source text, *Warning:* Possible hallucination

Precision Examples

Source Text: "OpenAI signed a $1Billion deal with Microsoft"

Precise (1):
Triplet: ["OpenAI","Partners_With","Microsoft"]
Verdict: 1 (Precise), *Reasoning:* Specific entities and relation, *Warning:* None

Triplet: ["OpenAI","Signed","$1Billion deal"]
Verdict: 1 (Precise), *Reasoning:* Specific entities and amount, *Warning:* None

Not Precise (0):
Triplet: ["Company","Related_To","Something"]
Verdict: 0 (Not Precise), *Reasoning:* Generic entity and broad relation, *Warning:* Generic entity

Triplet: ["OpenAI","Signed","$2Billion deal"]
Verdict: 0 (Not Precise), *Reasoning:* Amount mismatch with text, *Warning:* Amount mismatch

Relevance Examples

Source Text: ”OpenAI signed a $1Billion deal with Microsoft in Texas”

Relevant (1):
Triplet:[”OpenAI”,”Partners_With”,”Microsoft”]
Verdict: 1 (Relevant), *Reasoning:* Triplet relevant for the source text, *Warning:* None
Not Relevant (0):
Triplet: [”OpenAI”,”Signed”,”in Texas”]
Verdict: 0 (Not Relevant), *Reasoning:* Location not relevant to the main topic, *Warning:* Off-topic

Comprehensiveness Examples

Source Text: ”In 2024, OpenAI signed a $1Billion deal with Microsoft for AI partnership in Texas.”

Score 3 (Good):
Set of Triplets: [”OpenAI”,”Partners_With”,”Microsoft”], [”OpenAI”,”Signed”,”$1Billion deal”], [”OpenAI”,”Signed”,”in Texas”], [”OpenAI”,”Signed”,”for AI partnership”], [”OpenAI”,”Signed”,”in 2024”], [”Microsoft”,”Partners_With”,”OpenAI”], [”Microsoft”,”Signed”,”$1Billion deal”], [”Microsoft”,”Signed”,”in Texas”], [”Microsoft”,”Signed”,”for AI partnership”], [”Microsoft”,”Signed”,”in 2024”]
Reasoning: The set of triplets covers all core facts from the source text.
Warning: None

Score 2 (Partial):
Set of Triplets: [”OpenAI”,”Partners_With”,”Microsoft”], [”OpenAI”,”Signed”,”$1Billion deal”], [”OpenAI”,”Signed”,”in Texas”], [”OpenAI”,”Signed”,”for AI partnership”], [”Microsoft”,”Partners_With”,”OpenAI”], [”Microsoft”,”Signed”,”$1Billion deal”]
Reasoning: It misses the date and does not cover all for Microsoft.
Warning: Possible positional bias and missing information.

Score 1 (Poor):
Set of Triplets: [”OpenAI”,”Partners_With”,”Microsoft”]
Reasoning: It misses core facts.
Warning: Incomplete set of triplets generation.