

HX-Fin at FinEval 2026: Precision-Anchored Answer Refinement and Multi-Model Evidence Selection for Multilingual Financial Question Answering

Peng Li Yinchun Wang
ailab@chinaamc.com

Abstract

Financial question answering requires both a concise answer grounded in news and a supporting excerpt from heterogeneous financial statements. We present a training-free modular pipeline that separates the two objectives, where the evidence module relies on the frozen output of the answer module. The Answer module constructs a high-precision core using Pairwise Selection, synthesizes a higher-recall extended answer from five candidate views, and selectively transfers supported content through Delta Augmentation and Claim Pruning. The Evidence module applies generate-refine-MBR selection to obtain a primary evidence candidate, independently generates an alternative with QWEN3.6-THINK, and uses Conservative Evidence Selection to choose the final evidence. We evaluate answer module performance on 76 examples scored by a rubric-guided LLM judge. On the 0–4 judging scale, Pairwise Selection raises the average answer score from 2.316 to 2.540 (+0.224), while Delta Augmentation and Claim Pruning provide further gains of +0.079 and +0.066, respectively.

1 Introduction

FinEval2026-Multilingual Financial Q&A, based on MultiFinBen (Peng et al., 2026), requires systems to produce a news-grounded Answer and supporting Financial Statements Evidence from a news passage and indexed financial statements. The task spans English, Chinese, Japanese, Spanish, and Greek, and restricts participating systems to models with fewer than 70 billion parameters in total (FinNLP 2026 Organizers, 2026).

This setting introduces three practical challenges. First, Answer generation and Evidence extraction have different objectives: the former must summarize the requested claim, whereas the latter must locate a precise supporting statement. Second,

the long cross-document input encourages both omission and over-generation. A concise answer may miss a requested comparison, but aggressive synthesis can introduce unsupported entities, periods, or numerical details. Third, outputs generated across diverse input views, models, and reasoning paradigms can convey complementary contents, underscoring the need for robust selection and fusion mechanisms. A single-pass generator provides little control over these competing objectives and sources of variation.

We therefore adopt a training-free, modular workflow that separates Answer Generation from Evidence Generation. The Answer module begins with five complementary candidate views: full-input generation without thinking, news-only generation without thinking, full-input generation with medium reasoning, multilingual MBR fusion, and retrieval-augmented generation. It then constructs a high-precision core answer by applying an order-balanced pairwise router to the full-input and news-only candidates. In parallel, the complete candidate pool is synthesized into a higher-recall extended answer. Delta Augmentation, omission checking, and Claim Pruning selectively transfer supported content from the extended answer into the core, after which question-type-specific refinement and compression produce the final answer. This precision-anchored design expands coverage without allowing a noisy candidate to determine the final claim.

Evidence Generation is conditioned on the frozen answer and follows three stages. First, Qwen3.8-72B produces an initial Evidence candidate and two conservative refinements; MBR selection chooses the candidate with the strongest consensus as the default. Second, QWEN3.6-THINK independently generates an alternative Evidence candidate from the full query and source context. Third, Conservative Evidence Selection retains the MBR candidate unless the alternative is demonstrably more reliable. Keeping Answer construction

and Evidence selection separate makes the responsibility of each component explicit and prevents Evidence processing from changing the scope of the answer.

We evaluate Answer-pipeline transitions with qwen3.7-max as a rubric-guided LLM judge. On 76 test examples scored on a 0–4 scale, Pairwise Selection raises the average score from 2.316 to 2.540, an average gain of 0.224. Delta Augmentation and Claim Pruning provide further average gains of 0.079 and 0.066, respectively, while leaving most outputs unchanged.

Our contributions are threefold: a precision-anchored Answer pipeline with core answer selection and controlled answer refinement; a consensus-guided Evidence pipeline combining generate-refine-MBR selection, an independent QWEN3.6-THINK alternative, and Conservative Evidence Selection; and stage-wise LLM-judge analysis of Pairwise Selection, Delta Augmentation, and Claim Pruning.

2 Method

For each example, the input is a question q , a news passage n , and indexed financial statements S . The system produces an answer a and Evidence e , each of which may be None. The complete workflow is present in Figure 1.

2.1 Answer Generation

The Answer Generation module follows the five-stage, precision-anchored workflow. It first establishes a reliable core answer and then expands it only with content that survives targeted verification. Across all five stages, Qwen3.8-27B is used for LLM-based text generation and assessment: calls denoted as NoThink and MediumThink represents the LLM think parameter disable and medium-effort reasoning respectively.

Candidate answer pool generation. We generate complementary candidates from the raw query, where five methods are applied: Full_input+NoThink, News_input+NoThink, Full_input+MediumThink, Multilingual MBR and RAG(NoThink). Full_input represents all the query content are sent to the LLM. Multilingual MBR do inference for each language independently and then using a lexical minimum Bayes risk (MBR) for best result selection. Specifically, following sampling-based MBR decoding(Freitag et al.,

2022; Eikema and Aziz, 2022), our proposed MBR is defined as follows:

$$c^* = \arg \max_{c_i \in C} \frac{1}{|C| - 1} \sum_{c_j \in C \setminus \{c_i\}} u(c_i, c_j), \quad (1)$$

where C is the candidate set and u is an inverse-document-frequency (IDF) weighted lexical F1 score. For token t , its weight is computed from the candidate pool as

$$\text{idf}(t) = 1 + \log \frac{|C| + 1}{\text{df}(t) + 1}, \quad (2)$$

where $\text{df}(t)$ is the number of candidates containing t . IDF reduces the influence of frequent generic words and gives more weight to discriminative financial terms, entities, and numerical expressions.

Core answer construction. We select the core answer from Full_input+NoThink (c_{full}) and News_input+NoThink (c_{news}) with an order-balanced pairwise log-probability router (**Pairwise Selection**). Inspired by swap-order evaluation (Zheng et al., 2023) and position-bias calibration (Li et al., 2024), Qwen3.8-27B evaluates the pair twice. Prompt P_1 assigns label A to c_{full} and label B to c_{news} , whereas P_2 reverses these assignments. Let $\ell_y(P) = \log p(y | P)$ denote the log-probability of decision token $y \in \{A, B\}$. We normalize over the two decision tokens to obtain the probability assigned to c_{news} under each order:

$$p_{\text{news}}^{(1)} = \frac{\exp(\ell_B(P_1))}{\exp(\ell_A(P_1)) + \exp(\ell_B(P_1))}, \quad (3)$$

$$p_{\text{news}}^{(2)} = \frac{\exp(\ell_A(P_2))}{\exp(\ell_A(P_2)) + \exp(\ell_B(P_2))}. \quad (4)$$

These are normalized A/B decision-token probabilities, not candidate-generation likelihoods. We average their log-odds to obtain an order-balanced preference score:

$$L_{\text{news}} = \frac{1}{2} \sum_{k=1}^2 \log \frac{p_{\text{news}}^{(k)}}{1 - p_{\text{news}}^{(k)}} \quad (5)$$

The router selects c_{news} when $L_{\text{news}} > 0.5$; otherwise, it retains c_{full} .

Extended answer construction. In parallel, the system synthesizes an extended answer from the all the candidate answers. The extended answer favors coverage and may include useful quantitative changes, segment-level details, or strategic context that the conservative core excludes. Because this representation may also contain noise, it is used only as a source of proposed additions and is never published directly.

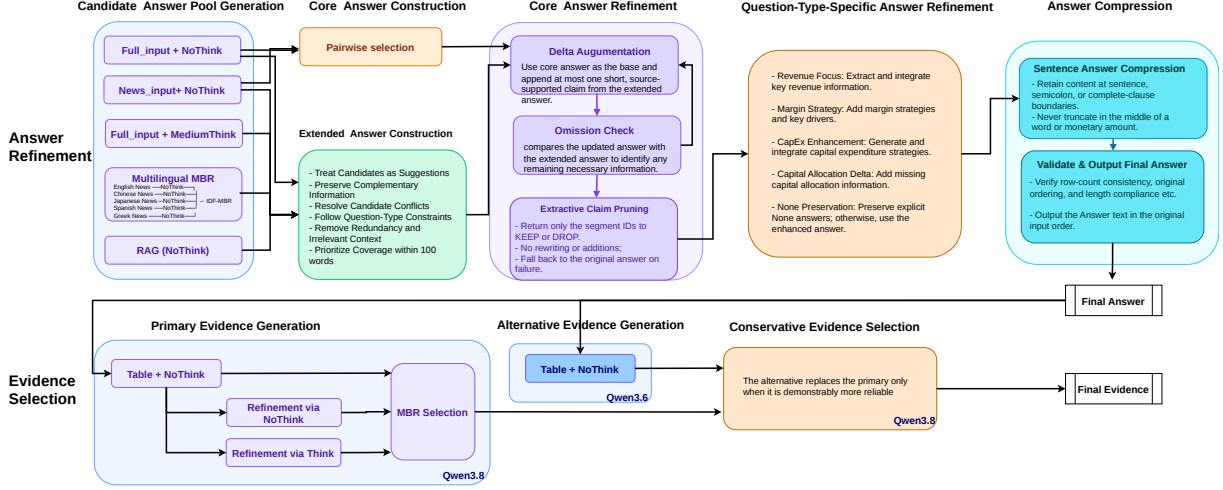


Figure 1: Overview of the workflow. The upper branch and the lower branch show the answer refinement and evidence selection process respectively.

Core answer refinement. As shown in Figure 1, core refinement contains three sequential operations. **Delta Augmentation** uses the core answer as the base and appends at most one short, source-supported claim from the extended answer. Its output is constrained to the JSON format `{"delta": ..., "augmented_answer": ...}`, where `delta` records the added claim and `augmented_answer` contains the resulting answer. If no safe addition is identified, `delta` is `None` and the core answer is returned unchanged. **Omission Check** then compares the updated core with the extended answer to identify any necessary information that remains missing. Finally, **Claim Pruning** assigns answer segments to `KEEP` or `DROP`; it does not rewrite existing claims or introduce new content, and it falls back to the incoming answer if pruning fails. Together, these operations add only targeted information and remove unsupported or redundant material while preserving the core answer’s precision.

Question-type-specific answer refinement. After core refinement, the system applies a question-type refinement. Revenue-focus questions extract and integrate key revenue information; margin-strategy questions add relevant strategies and drivers; capital-expenditure questions add the requested CapEx plans; and capital-allocation questions recover missing allocation details. Explicit `None` answers are preserved.

Answer Compression. Finally, we compress and normalize the refined answer, remove redundant material, enforce the required output format and word limit, and preserve explicit `None` decisions.

The resulting answer a is frozen before Evidence Generation.

2.2 Evidence Extraction

To ensure precise evidence-grounding while mitigating hallucination in null-evidence scenarios, we propose a three-stage evidence selection framework. Instead of relying on single-pass generation, our framework generates, refines, and selects candidates through Conservative Evidence Selection:

Primary Evidence Generation. We construct three complementary Evidence candidates and pick out the best as high-quality default evidence. First, `TableNoThink` prompts Qwen3.8-27B with thinking disabled to extract an initial candidate e_{tab} from the financial statement table. Starting from this same candidate, `Refinement via NoThink` conservatively corrects or reformats the evidence without explicit reasoning, producing e_{nt} , while `Refinement via Think` uses thinking-enabled self-correction to produce e_{t} . Next, we apply Minimum Bayes Risk (MBR) selection over the candidate pool following Eq. (1). Thus, the original extraction and the two refinement paths contribute equally as distinct hypotheses, and the candidate with the strongest consensus becomes the default high-quality Evidence.

Alternative Evidence Generation. To incorporate a reasoning path independent of the Qwen3.8 refinement and MBR stages, we use Qwen3.6-27B with thinking enabled, denoted as QWEN3.6-THINK. Because this candidate is produced by a different model checkpoint and decoding mode, it introduces substantial diversity into the candidate

pool. We treat QWEN3.6-THINK only as an alternative candidate, rather than merging it directly with the MBR output.

Conservative Evidence Selection. Rather than a naïve merge, we adopt a conservative selection mechanism. The QWEN3.6-THINK candidate replaces the primary evidence only if it meets strict confidence and verification criteria. This strategy minimizes false positives while retaining the complementary reasoning supplied by the auxiliary model.

3 Results

3.1 Setup

We evaluate system outputs with qwen3.7-max as an LLM judge, following prior work on rubric-guided LLM evaluation (Liu et al., 2023). For each example, the judge receives the question, the provided source material, and the generated answer, and assigns an integer score from 0 to 4 according to Table A1. We analyze the stage-wise behavior of the Answer pipeline on 76 test examples. For each transition, we report the average judge score before and after the operation together with the numbers of improved, unchanged, and declined examples.

3.2 Stage-wise Analysis

Table 1 reports independently scored changes across three Answer-pipeline transitions.

Stage transition	Before [†]	After [†]	Δ	G/S/B
Full_input+NoThink $\rightarrow$				
Pairwise Selection	2.316	2.540	+0.224	11/62/3
Pairwise Selection $\rightarrow$ Delta Augmentation	2.540	2.618	+0.079	9/61/6
Delta Augmentation $\rightarrow$ Claim Pruning	2.618	2.684	+0.066	9/62/5

Table 1: Stage-wise changes in the qwen3.7-max LLM-judge score. [†]Before and After report the average LLM-judge scores. G/S/B denotes the numbers of examples that improve, remain unchanged, or decline, respectively.

Pairwise Selection produces the largest average gain (+0.224), improving 11 of 76 examples while degrading 3. Delta Augmentation and Claim Pruning add smaller gains (+0.079 and +0.066), improving 9 examples each while leaving 61 and 62 unchanged, respectively.

4 Related Work

Multilingual financial evaluation. MultiFin established multilingual financial NLP resources beyond English (Jørgensen et al., 2023). MultiFin-Ben subsequently introduced multilingual financial reasoning and PolyFiQA as part of a broader multimodal benchmark (Peng et al., 2026). Our method

uses multilingual generation as an inference-time source of complementary Evidence hypotheses rather than as a data augmentation or fine-tuning scheme.

Consensus-based selection. Consensus across sampled reasoning paths can improve reliability over greedy decoding (Wang et al., 2023), and Universal Self-Consistency extends this idea to free-form generation by selecting the most consistent candidate with an LLM (Chen et al., 2023). MBR decoding instead selects an output according to expected utility rather than model probability alone (Freitag et al., 2022; Eikema and Aziz, 2022). We adapt these consensus principles to a small multilingual candidate set using lexical agreement. Selection does not finalize the result: a separate, order-reversed verifier determines whether the alternative is strong enough to replace the primary.

5 Conclusion

We presented a training-free modular pipeline for jointly producing answers and financial-statement Evidence. The Answer module combines five candidate views, constructs a precision-oriented core through Pairwise Selection, and then refines the core by adding supported content from an extended answer via Delta Augmentation and removing unsupported or redundant claims via Claim Pruning. Once frozen, the answer conditions a three-stage Evidence pipeline: generate–refine–MBR selection produces the default candidate, QWEN3.6-THINK supplies an independent alternative, and Conservative Evidence Selection chooses the final output. Stage-wise analysis identifies Pairwise Selection as the largest answer improvement, followed by smaller gains from Delta Augmentation and Claim Pruning. Future work should quantify variability across repeated runs, strengthen factual verification beyond lexical consensus.

Limitations

The proposed pipeline involves multiple operational stages—including generation, routing, refinement, and fallback. Due to the stochastic nature of LLM outputs, maintaining consistency across multiple runs remains a limitation, which we aim to address in future work.

References

Xinyun Chen, Renat Aksitov, Uri Alon, Jie Ren, Kefan Xiao, Pengcheng Yin, Sushant Prakash, Charles Sutton, Xuezhi Wang, and Denny Zhou. 2023. [Universal](#)

self-consistency for large language model generation. *arXiv preprint arXiv:2311.17311*.

Bryan Eikema and Wilker Aziz. 2022. [Sampling-based approximations to minimum Bayes risk decoding for neural machine translation](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10978–10993. Association for Computational Linguistics.

FinNLP 2026 Organizers. 2026. [FinNLP 2026 Shared Task](#). Official shared-task website. Accessed 2026-08-28.

Markus Freitag, David Grangier, Qijun Tan, and Bowen Liang. 2022. [High quality rather than high model probability: Minimum Bayes risk decoding with neural metrics](#). *Transactions of the Association for Computational Linguistics*, 10:811–825.

Rasmus Jørgensen, Oliver Brandt, Mareike Hartmann, Xiang Dai, Christian Igel, and Desmond Elliott. 2023. [MultiFin: A dataset for multilingual financial NLP](#). In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 894–909. Association for Computational Linguistics.

Zongjie Li, Chaozheng Wang, Pingchuan Ma, Daoyuan Wu, Shuai Wang, Cuiyun Gao, and Yang Liu. 2024. [Split and merge: Aligning position biases in LLM-based evaluators](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11084–11108. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. [G-eval: NLG evaluation using GPT-4 with better human alignment](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522. Association for Computational Linguistics.

Xueqing Peng, Lingfei Qian, Yan Wang, Ruoyu Xiang, Yueru He, Yang Ren, Mingyang Jiang, Vincent Jim Zhang, Yuqing Guo, Jeff Zhao, Huan He, Yi Han, Yun Feng, Yuechen Jiang, Yupeng Cao, Haohang Li, Yangyang Yu, Xiaoyu Wang, Penglei Gao, and 28 others. 2026. [MultiFinBen: Benchmarking large language models for multilingual and multimodal financial application](#). In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16934–16963. Association for Computational Linguistics.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. [Self-consistency improves chain of thought reasoning in language models](#). In *The Eleventh International Conference on Learning Representations*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhonghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. [Judging](#)

LLM-as-a-judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, volume 36.

A LLM-Judge Rubric

Score	Label	Criterion
4	Fully correct	Directly, correctly, and completely answers the question. All central claims are supported by the source; the entity, metric, period, number, unit, and direction are correct, with no substantive omission or error.
3	Mostly correct	The core answer is correct and the main requirements are covered. Any omissions, redundancy, or other issues are minor and do not affect the conclusion.
2	Partially correct	Contains useful and correct information, but has clear omissions, local factual errors, or insufficient source support. The answer nevertheless retains some practical value.
1	Seriously deficient	Contains little relevant information. Most of the question is unanswered, or the response has major errors involving the main metric, entity, period, or causal relation.
0	Invalid	Entirely incorrect, irrelevant, or unsupported in its central conclusion; also applies when the response is <code>None</code> although the provided sources contain sufficient information to answer the question.

Table A1: LLM-judge rubric used to score generated answers on a 0–4 scale.