

ADML at the FinEval 2026: Version-Aware Retrieval and Grounded Sparse Editing for Multilingual Financial QA

Gautier Marti

ADIA

gautier.marti@gmail.com

Khaled AlNuaimi

ADIA

Khalifa University

Abstract

We present ADML’s second-place system for the FinEval 2026 PolyFiQA subtask. PolyFiQA requires concise answers grounded jointly in financial statements and contemporaneous news in five languages. We developed a deterministic fact compiler with strict number provenance, schema-specific realization, a 100-word budget, and a sparse answer-edit lattice. All 76 released test queries exactly matched previously public PolyFiQA-Expert-July queries. Under the organizer’s external-data policy, our selected lane retrieved only exact or conservatively normalized full-query references and capped them without rephrasing. It scored 0.9994 public ROUGE-1 F1 but 0.3726 privately, ranking second. The sharp split exposes reference-version shift: input identity and historical answer identity held publicly but not privately. Historical leave-company-out experiments similarly show that answer-oracle headroom is easy to construct but difficult to route safely. We argue for version-aware retrieval and abstaining grounded fallbacks rather than unconditional reference reuse.

1 Introduction

PolyFiQA evaluates multilingual financial question answering from a financial report and news in English, Chinese, Japanese, Spanish, and Greek (Peng et al., 2026). The Easy and Expert datasets require cross-document evidence integration and use ROUGE-1 (Lin, 2004). Their long contexts, repeated question families, numerical units, and multilingual evidence make unconstrained generation both expensive and error-prone.

We built an input-grounded system before test release. It parses statements into typed facts, retrieves question-specific news, computes requested quantities, and realizes a schema-valid answer. Historical evaluation revealed a second challenge: small wording changes can improve ROUGE on

individual rows, yet learned selectors fail across companies and reporting periods. We therefore use sparse candidate edits and abstain unless a held-company gate supports change.

After test release, a provenance audit found that all 76 query packets exactly matched earlier public PolyFiQA references. The rules permitted external data, so our selected submission reused only full-query matches and applied the 100-word limit. Public performance was nearly perfect, but the private reference set had evidently been revised. This paper documents both the selected memory lane and the model-independent compiler that provides a novel-input fallback.

2 Task and Data

The original PolyFiQA-Easy and Expert releases each contain 204 questions grounded in filings and aligned multilingual news (Peng et al., 2026). We use older and July 2025 public snapshots for development, grouping all questions from the same company/report. The official test packet contains 76 questions: 24 public and 52 private. A prediction is a complete answer of at most 100 whitespace-delimited words; the leaderboard reports macro item-level ROUGE-1 F1, precision, and recall.

We preserve every source file and revision with SHA-256. Historical answers are used only by the explicit retrieval lane and by retrospective evaluation. The grounded compiler never reads a reference answer at inference.

The 76 test rows form 19 four-question company/report bundles. Keeping a bundle intact is essential: random row splits would put the same financial table, reporting period, and news evidence on both sides of an evaluation. Our historical protocols therefore group by company/report and, where two snapshot dates exist, train on an older packet and evaluate on a later one. The two public reference snapshots jointly provide 343 valid histor-

ical rows; 192 byte-aligned newer rows are used for the principal sparse-edit analysis and 151 older, packet-disjoint rows for temporal confirmation.

ROUGE-1 F1 is computed per answer and then macro-averaged. It is sensitive to both fact selection and realization: adding a true but unrequested sentence can lower precision, while omitting a reference explanation lowers recall. For this reason we report numeric grounding as a hard validity constraint but use ROUGE only for selecting among answers that already pass that constraint.

3 System

3.1 Grounded fact compiler

The parser separates the question, financial statements, and five news sections. Markdown statement rows become typed records (s, m, p, u, v) for statement, metric, period, unit, and value. Specialized calculators handle four Easy families (revenue trends, balance health, cash flow, and R&D ratio) and four Expert families (revenue focus, margin maintenance, capital allocation, and capital expenditure).

Question routing is deterministic. It identifies the requested family, company, target period, comparison period, and whether the answer expects a trend, ratio, level, or evidence-supported explanation. Header aliases and language-specific punctuation are normalized before table parsing, while the original surface is retained for citation and output. A fact is represented as

$$f = (\text{source}, \text{metric}, \text{period}, \text{unit}, \text{value}, \text{span}).$$

Derived facts additionally store their operands and operation. Allowed operations are difference, percentage change, ratio, and sign/trend comparison; division by zero and incompatible units fail closed.

Every generated number must be copied from a statement/news span or derived by an allowed arithmetic operation. Period and unit checks prevent quarter/year and million/billion mismatches. News sentences are ranked by question cues and agreement with statement amounts. Realization uses the expected Answer: and evidence headings, then an official-token-aware budget limits secondary CJK evidence. Empty or unparsable cases emit an explicit “None” schema rather than hallucinated facts.

The numeric validator extracts every output number, normalizes separators and percent signs,

and requires a provenance record. It permits formatting changes such as a rounded percentage only within a fixed tolerance. Company names and years are checked separately so they cannot be justified by an unrelated table cell. A complete historical and test rehearsal passed structure, alignment, word budget, and numeric provenance; the longest input-only test answer was 88 words. These checks are independent of ROUGE and run even when the candidate is otherwise the highest-scoring development answer.

3.2 Sparse editing and abstention

A strong factual core may still score poorly if its scope or evidence differs from the reference. For each baseline answer we construct a small lattice: unchanged, core only, evidence only, first sentence, one-sentence deletion, or a fully validated alternative. A nested ridge risk model is trained leave-company-out to predict the ROUGE delta. Hyperparameters and the minimum predicted gain are selected using only the other companies.

Let $a_0(x)$ denote the validated compiler answer and $C(x)$ its edit lattice. The answer oracle is $a^*(x) = \arg \max_{a \in C(x)} R(a, y)$ and measures candidate headroom only; it is never an inference system because it observes reference y . The deployable selector instead estimates $\hat{\Delta}(x, a) = R(a, y) - R(a_0, y)$ from input-only features such as family, answer length, evidence count, candidate operation, fact coverage, and language. For each held company, both its ridge regularization and minimum gain threshold are chosen without that company. A candidate replaces a_0 only if the predicted lower-risk gain is positive; otherwise the selector abstains.

The lattice oracle raises historical ROUGE substantially, but the learned selector chooses the unchanged answer on every held-company row. This safe abstention is preferable to tuning globally: a revenue compiler improved core-only evaluation but reduced complete-answer ROUGE by .1023. Larger constrained planners also improved recall while concentrating harms in individual issuers.

We tested temporal transfer as a stricter control. A selector trained on older packets from three issuers and applied to the newer packet of the fourth changes four rows, harms all four, and moves .3652 to .3644. An Easy-cash evidence expansion that improves every issuer in its discovery data harms both eligible rows in a later packet. Stable oracle

Evaluation	System	R-1 F1	Δ
Hist. 192	grounded v15	.3652	–
Hist. 192	sparse-edit oracle	.4030	+0.0377
Hist. 192	nested risk selector	.3652	.0000
Revenue 14	full-answer compiler	.2563	-.1023
Public 24	input-only v15	.3214	–
Public 24	exact memory	.9994	+0.6780
Private 52	exact memory	.3726	–

Table 1: Historical and official macro ROUGE-1 F1. The exact-memory submission ranked second privately.

gains of .0358 (older) and .0377 (newer) therefore coexist with unstable routing. This motivates abstention rather than choosing an edit on the same rows used to demonstrate oracle headroom.

3.3 Version-aware reference retrieval

The selected lane indexes earlier public references by the complete raw query and a conservative NFKC/whitespace normalization. Matching on task ID, company, question template, or fuzzy similarity is disabled. Retrieved answers are copied verbatim except for a deterministic 100-word cap that preserves the evidence heading. Unmatched packets fall back to the grounded compiler. All 76 released test queries matched exact historical queries.

The index builder records snapshot revision, download time, license metadata, query digest, and answer digest. It aborts if two snapshots contain the same normalized query with conflicting answers; such conflicts require an explicit version preference rather than last-write-wins behavior. At inference the raw exact key has priority over the normalized key, and a normalized match is accepted only when it is unique. Row IDs never participate in lookup.

This policy protects against cross-period retrieval but cannot protect against a new answer annotation attached to an identical query. The public/private result demonstrates that distinction.

4 Experiments

Historical evaluation uses the organizer-compatible multilingual ROUGE tokenizer and company/report grouping. Table 1 reports the key selection evidence and final leaderboard results.

The historical sparse-edit oracle improves 139 of 192 rows with a mean of 7.98 valid candidates. Gains appear in all eight question families, from +0.0170 for Easy R&D to +0.0701 for Easy

revenue. Yet nested leave-company-out selection makes zero edits and exactly preserves .3652. This gap is the central selection problem: candidate diversity is not useful without a transferable risk signal.

Oracle choices are heterogeneous: 53 rows retain the baseline, 46 choose an evidence-only variant, 36 delete one sentence, 29 select a fully validated alternative, 17 keep the first sentence, and 11 choose the core only. The largest family gains occur for Easy revenue (+0.0701), balance (+0.0506), and cash (+0.0479); even the smallest, Easy R&D, remains positive (+0.0170). There is no single universal length or evidence rule that realizes the oracle.

We also asked whether model scale solved fact selection. Given only the question and a finite typed fact graph, constrained 35B-total and 57B-total models returned fact IDs rather than prose. On a 32-row company-by-family screen, the deterministic planner obtains .8367 macro set F1, versus .8237 and .7997. The larger models improve nonempty-target recall but over-select numeric facts when references are qualitative. Separate 27B and 35B surface-realization pilots preserve every numeric placeholder and still reduce ROUGE. None was promoted, and no model over the competition’s 70B-total limit was used.

The selected leaderboard system obtains public precision 1.0000 and recall .9988. Privately, F1 is .3726, precision .5698, and recall .3542. Because exact retrieval coverage is 76/76, the drop is not caused by missing queries. The most plausible explanation is reference-version shift: the public subset retains historical answers, while many private gold answers use a different revision or annotation policy. We do not have private labels and therefore cannot quantify the exact change mechanism.

The direction of the private precision–recall gap is also informative. Old references overlap substantially with revised answers, but they contain many tokens absent from the private references and omit other expected tokens. A pure answer-length adjustment could trade precision for recall, but cannot be selected safely without private references. The nearly perfect public score would actively encourage preserving the historical wording, making public-only selection misleading in this setting.

Despite this shift, the system ranked second among 15 teams, 0.0101 behind the winner and 0.0024 ahead of third place. Its private score is also close to the .3652 historical grounded base-

line, suggesting that exact old references remained lexically relevant even when no longer identical.

Our input-only v15 system scores .3214 publicly, well below the public-memory lane. We cannot compute its private score because only the selected submission was evaluated privately. It would therefore be incorrect to claim that the compiler would have beaten memory under the new references; .3652 is a historical, not private, measurement. The defensible conclusion is narrower: the selected memory answer was brittle, and the validated compiler supplies a reproducible fallback for novel inputs.

4.1 Selection under version uncertainty

Before private scoring, the available evidence strongly favored memory: exact coverage was complete, the external-data rule was explicit, and public ROUGE was .6780 above the compiler. There was no label-free statistic that identified which of the 76 answers had been revised. Selecting the compiler instead would therefore have been a guess based on anticipated organizer behavior, not a validated model decision. We retained exact memory and recorded the compiler as fallback. This chronology matters when interpreting the private result: the .3726 score diagnoses a missing version signal, not a selector tuned after observing the hidden leaderboard.

A future selector needs evidence unavailable to ordinary exact matching. Candidate signals include disagreement among timestamped answer snapshots, changes in the current financial table relative to the cached answer’s numeric claims, and semantic coverage of the present news. These can be evaluated on older-to-newer public revisions without seeing current gold. In contrast, surface similarity between the current and historical query is constant at its maximum here and contains no routing information.

4.2 Operational profile

The selected runner is CPU-only. It validates the 76-row input hash, constructs the versioned index, requires unique 76/76 full-query coverage, applies the organizer-compatible word counter, and writes the original IDs in order. It loads no model and calls no API. The input-only fallback separately verifies answer headings, the 100-word cap, source alignment, and numeric provenance. A detached manifest pins all inputs, candidates, and the submitted CSV, so an organizer can reproduce both

lanes and confirm which one was selected.

5 Discussion and Conclusion

Exact input overlap is insufficient evidence for exact answer reuse in generative evaluation. A benchmark can preserve questions and evidence while revising completeness, phrasing, or evidence-selection expectations. Public-only overlap diagnostics would have dramatically overstated the private value of memory.

Our development experiments support a conservative alternative. Typed facts and provenance checks prevent unsupported numbers; sparse edits expose oracle headroom; leave-company-out risk estimation prevents non-transferable rewrites. The selected memory lane was competition-legal and competitive, but future systems should combine version-aware retrieval with a stronger semantic criterion for choosing between historical memory and grounded regeneration.

A promising criterion is answer-version uncertainty. Retrieval should return not only a candidate answer but also snapshot disagreement, temporal distance, and coverage of the current evidence. A grounded compiler can then verify each retrieved claim and regenerate only unsupported or outdated components. This hybrid differs from copying the nearest answer and from free-form editing: every changed number remains tied to a current fact, while unchanged language retains the lexical benefit of memory.

Limitations

ROUGE-1 rewards lexical overlap rather than full factual correctness. Our historical snapshots have inconsistent license metadata, so they are distributed only in the organizer verification bundle. Exact-query retrieval does not generalize to novel reports. Finally, without private references, version shift is inferred from exact input coverage and score behavior rather than directly observed.

The test set is small and bundles repeated question families, so leaderboard ranks have high variance. The deterministic compiler covers the released schemas, not arbitrary financial questions, and its conservative “None” fallback may be unhelpful to an end user. Financial answers should not be used for investment decisions without checking the underlying filing and news; numeric provenance reduces hallucination but does not establish materiality or correct economic interpretation.

References

- Chin-Yew Lin. 2004. [ROUGE: A package for automatic evaluation of summaries](#). In *Text Summarization Branches Out*, pages 74–81. 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.

A Reproducibility Details

The selected lane is deterministic, CPU-only, and uses no API or neural model. Its CSV hash begins 158ed1e8d398; the manifest records the complete digest. Submission ID begins eb18cb13; it scored .9994076 public and .3726320 private ROUGE-1 F1. The complete code bundle pins three public reference snapshots and verifies every file before inference.

B Additional Negative Controls

A 35B sparse mixture planner raised nonempty-target set F1 on a 32-row screen but reduced macro set F1 by .0130. A precision intersection improved a separate 28-row confirmation set by .0184 overall, yet harmed one held issuer by .0544 and was rejected under the issuer-safety gate. These experiments used fewer than 70B total parameters and were not part of the final inference path.