

Utilizing Metadata for Better Retrieval-Augmented Generation

Raquib Bin Yousuf¹[0009-0001-1245-0450], Shengzhe Xu¹[0009-0008-8589-6526], Mandar Sharma¹[0000-0002-7012-9323], Andrew Neeser¹, Chris Latimer², and Naren Ramakrishnan¹[0000-0002-1821-9743]

¹ Virginia Tech, Virginia, USA
 {raquib, shengzx, mandarsharma, aneeser24, naren}@vt.edu
² Vectorize.io, Colorado, USA
 chris.latimer@vectorize.io

Abstract. Retrieval-Augmented Generation systems depend on retrieving semantically relevant document chunks to support accurate, grounded outputs from large language models. In structured and repetitive corpora such as regulatory filings, chunk similarity alone often fails to distinguish between documents with overlapping language. Practitioners often flatten metadata into input text as a heuristic, but the impact and trade-offs of this practice remain poorly understood. We present a systematic study of metadata-aware retrieval strategies, comparing plain-text baselines with approaches that embed metadata directly. Our evaluation spans metadata-as-text (prefix and suffix), a dual-encoder unified embedding that fuses metadata and content in a single index, dual-encoder late-fusion retrieval, and metadata-aware query reformulation. Across multiple retrieval metrics and question types, we find that prefixing and unified embeddings consistently outperform plain-text baselines, with the unified at times exceeding prefixing while being easier to maintain. Beyond empirical comparisons, we analyze embedding space, showing that metadata integration improves effectiveness by increasing intra-document cohesion, reducing inter-document confusion, and widening the separation between relevant and irrelevant chunks. Field-level ablations show that structural cues provide strong disambiguating signals. Our code, evaluation framework, and the RAGMATE-10K dataset are publicly hosted³.

Keywords: Retrieval-Augmented Generation (RAG) · Metadata-aware Retrieval · Dense Retrieval · Query Reformulation · Benchmark Datasets

1 Introduction

Large Language Models (LLMs) have become central to information access, enabling systems that answer questions, summarize documents, and reason over

³ <https://github.com/raquibvt/RAGMate>

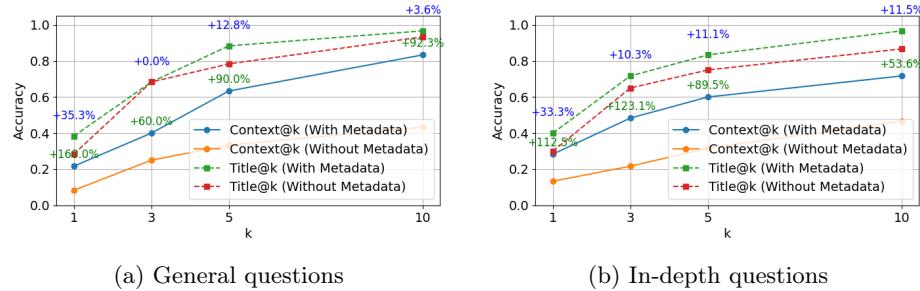


Fig. 1: Context@K and Title@K metric improvement in retrieval when using and not using metadata across query types (Dual Encoder Unified Embeddings)

unstructured corpora [29, 4]. Retrieval-Augmented Generation (RAG) architectures extend these models by first retrieving relevant context and then conditioning generation on it [22]. This hybrid setup has become the dominant paradigm for open-domain question answering, long-document understanding, and domain-specific inference [15, 11].

However, RAG pipelines are only as reliable as the context they retrieve. When document corpora are repetitive or semantically ambiguous, such as regulatory filings, legal records, or scientific papers, retrievers often return superficially relevant but substantively unhelpful chunks [15, 28]. In these settings, small differences in document structure or entity context (for example company name, fiscal year, or section title) become essential for retrieving the correct information.

A particularly illustrative case is the U.S. Securities and Exchange Commission (SEC) Form 10-K filings. These documents follow rigid templates, reuse language across companies and years, and contain subtle, structure-dependent variations. Such properties challenge traditional vector-based retrieval, which relies on chunk-level semantic embeddings [12, 15] that often lack sufficient discriminative power. Many of the disambiguating signals already exist in structured metadata such as filing year, form type, section heading, and industry classification, yet these fields are usually applied only for post-retrieval filtering [35].

For example, consider the query “What risks does the company identify related to supply chain disruptions?” Without metadata, many chunks across different companies, years, and sections contain nearly identical language, causing semantic-only retrievers to return plausible but incorrect context. Metadata such as company, fiscal year, and section title disambiguate the query intent, anchoring retrieval to the correct filing and section.

SEC filings have therefore become a common testbed for financial QA with LLMs [19, 20, 7], alongside broader benchmarks in finance [31, 10]. Together, these studies highlight both the opportunities and the limitations of RAG pipelines on repetitive, domain-specific corpora. Motivated by recent work in contextual fine-tuning and long-context prompting [1, 5], we ask:

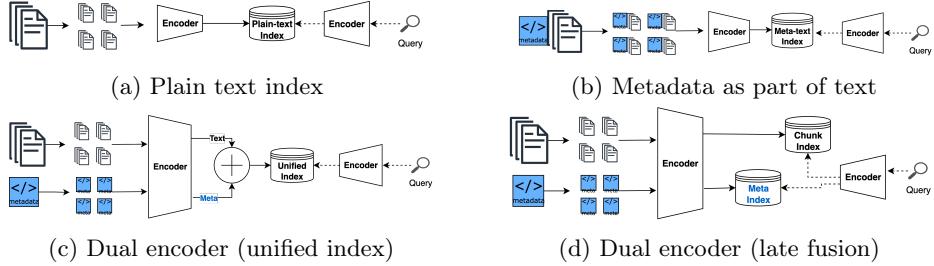


Fig. 2: Conceptual overview of document and metadata embedding strategies for retrieval. Dotted arrows indicate query propagation.

Can metadata be utilized as a first-class input in RAG pipelines, not just as a filter, but as embedded content that improves retrieval quality and downstream answer correctness?

We investigate whether metadata-aware retrieval benefits from more modular designs such as dual encoders that embed content and metadata separately, or query reformulation that explicitly surfaces metadata cues. As a simple baseline, we also evaluate adding metadata directly to the chunk as text. In particular, prefixing metadata before the chunk yields strong retrieval accuracy, though it is computationally expensive since any metadata update requires re-embedding the full index. To make maintenance lighter, we propose a dual-encoder framework in which metadata and content are embedded independently and then combined. Metadata embeddings only need to be computed once per field and can be fused with precomputed text embeddings, avoiding costly re-embedding of the full corpus. Among these variants, unified embeddings, where metadata and content vectors are summed into a single index, achieve accuracy comparable to, and sometimes surpassing, prefixing while simplifying serving. In contrast, late-fusion scoring is less competitive.

Beyond empirical comparisons, we also analyze why metadata integration improves retrieval. Through statistical analysis of embedding spaces, we show that metadata increases intra-document cohesion, decreases inter-document confusion, and expands the variance of similarity scores, creating a more discriminative geometry for retrieval. This analysis confirms that metadata does not simply add tokens but reshapes the embedding landscape in a way that improves separability between relevant and irrelevant chunks.

We summarize our contributions as follows:

- To our knowledge, the first systematic study showing that metadata signals significantly improve retrieval on RAG systems.
 - A dual-encoder framework that embeds metadata and content separately, improving index maintenance while achieving accuracy on par with, and sometimes better than, prefixing.

- A statistical embedding-space analysis that explains these gains, demonstrating increased intra-document cohesion, reduced inter-document confusion, and clearer separation between relevant and irrelevant chunks.
- Field-level ablations show that company and year fields provide strong disambiguating signals, while section titles offer more modest gains.
- Release of RAGMATE-10K, a metadata-rich benchmark with chunk grounded QA for reproducible evaluation.

We review related work (§2), introduce metadata-as-text and dual-encoder designs (§3.2), describe the RAGMATE-10K dataset and setup (§4), present results (§5), analyze embedding geometry (§5.1), report ablations on section titles (§6), and conclude with limitations and future work.

2 Related Work

Information access has long motivated technologies from search engines [27, 8] to diverse downstream applications [17, 24]. The advent of attention-based models [30, 12] and modern chat-based LLMs [29, 4] has made retrieval-augmented generation (RAG) [22] a central paradigm. While standalone LLMs often hallucinate, which is problematic in high-stakes domains such as medicine [33], law [28], intelligence analysis [32], and software engineering [21], RAG mitigates this by grounding generation in external knowledge. Approaches span both fine-tuned retrievers [25, 5] and retraining-free pipelines that dominate current practice [15, 11].

Within retrieval itself, methods differ in how they represent knowledge: some embed raw document chunks [15], others use (question, answer) pairs [35], and some generate synthetic queries or summaries to improve matching [14, 6, 26]. These span dense embedding approaches and sparse, keyword-based retrieval. Recent work such as EnrichIndex [9] further explores offline LLM-based enrichment by generating summaries, purposes, and synthetic QA pairs to strengthen first-stage retrieval without online LLM costs.

Most prior work leverages metadata only indirectly, either for filtering or through late-fusion signals. Financial QA benchmarks such as FinanceBench [19], SEC-QA [20], and SecQue [7], along with broader datasets in finance [31, 10], highlight the challenges of repetitive, domain-specific corpora but stop short of systematically evaluating metadata integration. While some recent approaches enrich documents with additional LLM-generated semantic context, they do not isolate the role of structured metadata as a retrieval signal. Our work differs by treating metadata as a first-class retrieval signal, comparing simple metadata-as-text strategies with modular dual-encoder designs.

3 Metadata as a First-Class Signal

We begin with the simplest approach to metadata integration: concatenating structured metadata fields directly to the chunk text before embedding. This

technique treats metadata as part of the semantic input (*metadata-as-text*, MaT) and encodes it into the same vector space as content.

3.1 Metadata as Text: A Minimal Baseline

Let the corpus be a set of N document chunks with associated structured metadata:

$$D = \{(m_i, c_i)\}_{i=1}^N,$$

where c_i is the text of the i -th chunk, and m_i is a key-value map of structured fields (e.g., company, form_type, section, year).

We define a serialization function that produces a compact, human-readable header from the metadata:

$$s(m_i) = \text{"company : {...}; form : {...}; section : {...}; year : {...}"}$$

Using this header, we construct a metadata-prefixed chunk string via a concatenation operator:

$$\tilde{c}_i = \text{concat}(s(m_i), c_i).$$

We also evaluate a suffix variant that appends the metadata header after the chunk: $\tilde{c}_i^{\text{sf}} = \text{concat}(c_i, s(m_i))$. We report results for both prefix ($\tilde{c}_i^{\text{pre}} = \text{concat}(s(m_i), c_i)$) and suffix in Table 2, while figures use the prefix variant unless explicitly noted.

An off-the-shelf text encoder f_θ (frozen) maps strings to d -dimensional vectors:

$$\tilde{\mathbf{e}}_i = f_\theta(\tilde{c}_i) \in \mathbb{R}^d.$$

Given a user query q , we compute its embedding $\mathbf{e}_q = f_\theta(q)$ and retrieve by cosine similarity over the single MaT index:

$$\text{Score}_{\text{MaT}}(q, i) = \cos(\mathbf{e}_q, \tilde{\mathbf{e}}_i), \quad \text{rank by Score}_{\text{MaT}}.$$

3.2 Dual Encoders: Modular Integration

Flattening metadata into the chunk text (Section 3.1) improves retrieval but is computationally expensive, since any metadata update requires re-embedding the full chunk index. To address this, we design dual-encoder approaches that embed content and metadata separately, making updates lighter and more modular. Within this framework, we first present a unified single-index that merges both signals directly in embedding space, retaining the simplicity of serving while avoiding costly re-indexing. We then contrast it with a late-fusion dual encoder that combines scores at query time, and finally describe query-side strategies that surface metadata cues in the query.

Unified Single-Index via Weighted-Sum Fusion Let the corpus be $D = \{(m_i, c_i)\}_{i=1}^N$, where c_i is chunk text and m_i is a key–value metadata map (e.g., company, form, year, section). We encode content and metadata into the same d -dimensional space:

$$\mathbf{e}_i^{\text{text}} = f_{\theta}^{\text{text}}(c_i), \quad \mathbf{e}_i^{\text{meta}} = f_{\theta}^{\text{meta}}(m_i).$$

We L2-normalize both vectors and form a convex combination to build a single fused index:

$$\begin{aligned}\hat{\mathbf{e}}_i^{\text{text}} &= \frac{\mathbf{e}_i^{\text{text}}}{\|\mathbf{e}_i^{\text{text}}\|_2}, & \hat{\mathbf{e}}_i^{\text{meta}} &= \frac{\mathbf{e}_i^{\text{meta}}}{\|\mathbf{e}_i^{\text{meta}}\|_2}, \\ \mathbf{e}_i^{\text{sum}}(\alpha) &= \frac{\alpha \hat{\mathbf{e}}_i^{\text{text}} + (1 - \alpha) \hat{\mathbf{e}}_i^{\text{meta}}}{\|\alpha \hat{\mathbf{e}}_i^{\text{text}} + (1 - \alpha) \hat{\mathbf{e}}_i^{\text{meta}}\|_2}, & \alpha \in [0, 1].\end{aligned}\tag{1}$$

At query time, we embed the query once with the text encoder and retrieve by cosine similarity against the fused index. Since document embeddings are already L2-normalized, leaving the query unnormalized does not affect ranking under cosine similarity:

$$\text{Score}_{\text{sum}}(q, i; \alpha) = \cos(\mathbf{e}_q^{\text{text}}, \mathbf{e}_i^{\text{sum}}(\alpha)).\tag{2}$$

For inner-product distance, both $\mathbf{e}_q^{\text{text}}$ and $\mathbf{e}_i^{\text{sum}}(\alpha)$ should be L2-normalized to emulate cosine.

Eqs. (1)–(2) yield a single index of dimension d , without doubling as in concatenation. They keep metadata and text separable until fusion and avoid runtime score fusion.

Dual Encoder with Late-Fusion Scoring Alternatively, we maintain two indices (content and metadata) and combine scores at query time. Late fusion exposes α at query time, which is useful for diagnostics, but requires two lookups. Given a query text embedding $\mathbf{e}_q^{\text{text}}$ and document embeddings $(\mathbf{e}_i^{\text{text}}, \mathbf{e}_i^{\text{meta}})$, we compute

$$\text{Score}_{\text{late}}(q, i; \alpha) = (1 - \alpha) \cos(\mathbf{e}_q^{\text{text}}, \mathbf{e}_i^{\text{text}}) + \alpha \cos(\mathbf{e}_q^{\text{text}}, \mathbf{e}_i^{\text{meta}}), \quad \alpha \in [0, 1].\tag{3}$$

3.3 Query-Side Strategies: Metadata-aware Reformulation

We apply an LLM-based reformulation operator to incorporate schema cues (e.g., company, form, year, section) into the query, which is then embedded with the text encoder:

$$\phi_{\text{text}}(q) = \text{Reformulate}(q).$$

This reformulated query can be used with both metadata-as-text retrieval and dual-encoder variants but it adds extra overhead during query time. Query reformulation is implemented as a single LLM call conditioned on the metadata schema (Table 1) and a small set of example values per field. Given a query, the LLM extracts explicit metadata constraints and rewrites the query to incorporate them. Retrieval then uses the rewritten query embedded with the same frozen text encoder as all other methods, leaving the retrieval pipeline unchanged.

3.4 Embedding-Space Theory of Metadata Integration

We consider a metadata-informed embedding $\tilde{\mathbf{e}}_i^*$ that augments a chunk embedding $\mathbf{e}_i = f_\theta(c_i)$ with structured metadata, either through token-level prefixing (MaT) or vector-level fusion (Unified, see Sec. 3.2). Let $d \in \mathcal{D}$ denote a document (e.g., a company–year SEC filing) and $i \in d$ a chunk belonging to d . The following propositions describe how such embeddings reshape the similarity landscape.

Proposition 1 (Intra-document cohesion increases).

$$\mathbb{E}_{i,j \in d}[\cos(\tilde{\mathbf{e}}_i, \tilde{\mathbf{e}}_j)] > \mathbb{E}_{i,j \in d}[\cos(\mathbf{e}_i, \mathbf{e}_j)].$$

Metadata anchors chunks to their document identity, pulling them closer in embedding space.

Proposition 2 (Inter-document confusion decreases).

$$\mathbb{E}_{i \in d_1, j \in d_2, d_1 \neq d_2}[\cos(\tilde{\mathbf{e}}_i, \tilde{\mathbf{e}}_j)] < \mathbb{E}_{i \in d_1, j \in d_2, d_1 \neq d_2}[\cos(\mathbf{e}_i, \mathbf{e}_j)].$$

Metadata provides discriminative cues (company, year, section) that reduce spurious similarity across different documents.

Proposition 3 (Score variance increases).

$$\text{Var}[\cos(\mathbf{e}_q, \tilde{\mathbf{e}}_i)] > \text{Var}[\cos(\mathbf{e}_q, \mathbf{e}_i)], \quad q \sim \text{typical queries}.$$

Unified embeddings interpolate between content-only and metadata-only signals via a convex weight α , and therefore inherit the above properties whenever $\alpha < 1$. MaT achieves a similar effect through token-level injection, while Unified does so through vector-level fusion with tunable weighting. In effect, this creates a more structured space with clearer separation between relevant and irrelevant candidates.

4 Methodology

4.1 Dataset: RAGMATE-10K

We introduce RAGMATE-10K, a dataset of SEC 10-K filings designed to evaluate metadata-aware retrieval. It consists of 25 filings from five U.S. technology companies (Apple, Alphabet, Adobe, Oracle, Nvidia), each segmented into non-overlapping 350-token chunks with 50-token overlap. This yields $N = 4,490$ retrieval units, each represented as a tuple (m_i, c_i) where c_i is the text content and m_i its structured metadata (company, year, section, form type). RAGMATE-10K is publicly available.³

We create 30 human-authored templates that instantiate into company- and year-specific questions, covering both general (e.g., business overview) and in-depth (e.g., risk factors) information needs. Excluding Apple filings from evaluation avoids contamination, leaving 120 test queries.

Ground-truth answers are generated by constraining a language model to use only chunks from the target filing. The model must cite the supporting chunks, providing supervision for both retrieval accuracy and answer grounding.

Table 1: Flat metadata schema used in all experiments.

Field	Description	Example
<code>company_name</code>	Filing entity name	Alphabet Inc.
<code>form_type</code>	SEC form type	10-K
<code>section</code>	Document section heading	Item 1 - BUSINESS
<code>fiscal_year_end</code>	Fiscal year end date	12-31
<code>period_of_report</code>	Reporting period close date	2023-12-31
<code>filed_date</code>	SEC filing submission date	2024-01-31
<code>exchange_listings</code>	Public exchange(s) listed on [NYSE]	
<code>SIC_code</code>	Industry classification	COMPUTER

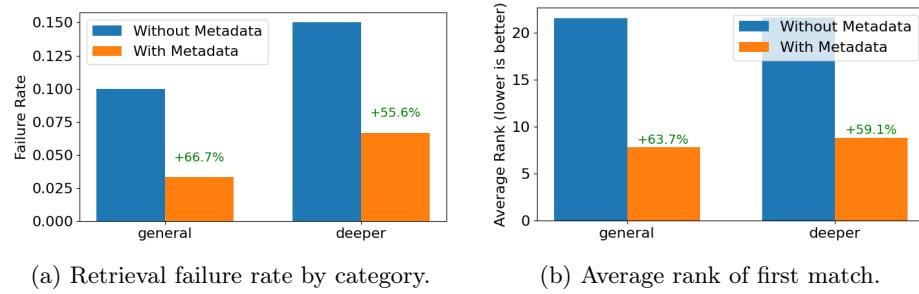


Fig. 3: Comparative retrieval performance vs. plain baseline across query types using the Dual Encoder Unified Embedding approach.

4.2 Implementation Details

We isolate metadata design effects by using a frozen text encoder f_θ and a fixed retrieval pipeline. Each retrieval unit is a pair (m_i, c_i) , where c_i is a chunk of document text and m_i is a flat key–value metadata dictionary.

We evaluate two representative embedding models: OpenAI’s `text-embedding-3-small` (dimension 1536) [2] and BAAI’s `bge-m3` (dimension 1024) [3], a strong open-source retriever optimized for multilingual and cross-domain retrieval. Both encoders are used in frozen form without fine-tuning.

Metadata m_i is represented as a flat key–value dictionary, independent of the chunk text. The fields are serialized into a fixed-order header for text-based variants, and passed verbatim to the metadata encoder for dual-encoder variants.

We evaluate retrieval quality using cosine similarity with top- K search, varying $K \in \{1, \dots, 10\}$. Performance is measured against ground-truth supporting chunks using four metrics:

- **Context@K**: whether at least one retrieved chunk within the top K supports the ground-truth answer.
- **Title@K**: whether at least one top- K chunk comes from the correct document (company and year).
- **Average Matched Rank**: the mean rank position of the highest-ranked supporting chunk (lower is better).

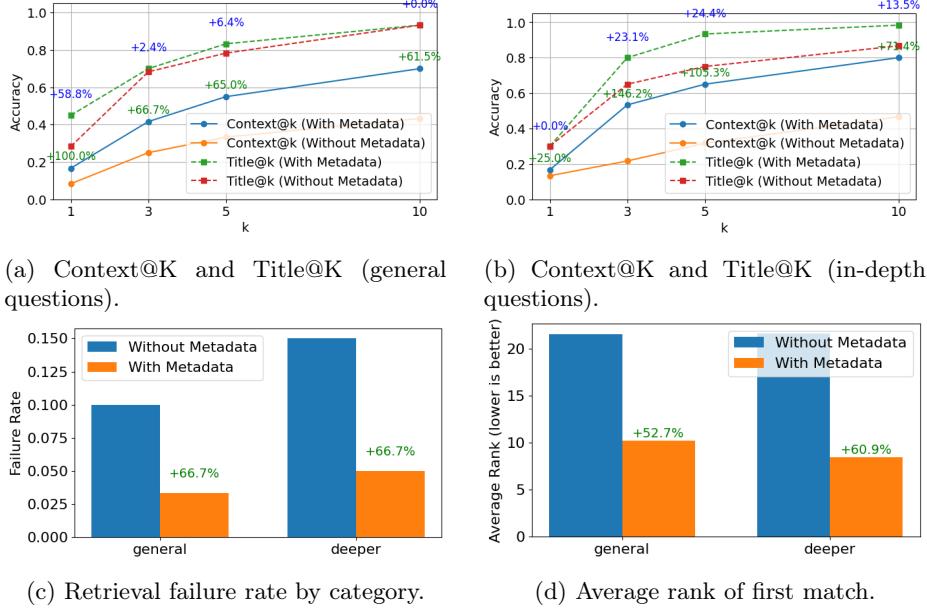


Fig. 4: Comparative retrieval performance vs. plain baseline across query types using Metadata as Text (Prefix)

- **Retrieval Failure Rate:** the proportion of queries for which no relevant chunk is retrieved (lower is better).

5 Findings

Our experiments show that metadata-aware retrieval consistently improves over a plain content-only baseline. Among the strategies, two stand out: prefix-based metadata-as-text (MaT) and dual-encoder unified embeddings.

Unified embeddings emerge as the most effective and practical approach. By fusing metadata and content vectors into a single index, they achieve accuracy that matches or surpasses prefixing while offering clear advantages in index maintenance and serving. This makes unified embeddings a strong candidate for deployment in real-world RAG systems where metadata evolves over time.

At the same time, metadata-as-text remains a simple and high-performing baseline. Direct concatenation reliably boosts retrieval accuracy and requires no architectural changes or additional infrastructure, making it appealing as a training-free baseline.

Late-fusion dual encoders and metadata-aware query reformulation trail these methods. Sweeping the fusion weight $\alpha \in \{0.0, 0.1, \dots, 1.0\}$ shows moderate values ($\alpha \approx 0.3\text{--}0.6$) work best, confirming that metadata should complement, not dominate, content signals. In practice, late fusion offers little advantage: unified embeddings provide similar balance within a single index, avoiding extra

Table 2: Retrieval performance summary at cutoff $k=5$ across two embedding models. Dual encoders are reported with $\alpha = 0.5$.

(a) OpenAI `text-embedding-3-small`

Method	General					Deeper				
	Context@5 ↑	Title@5 ↑	Avg	Rank ↓	Failure ↓	Context@5 ↑	Title@5 ↑	Avg	Rank ↓	Failure ↓
No Metadata	33.33	78.33	21.61	10.00		31.67	75.00	21.63	15.00	
Meta-Suffix	48.33	88.33	11.49	1.67		53.33	86.67	10.52	3.33	
Dual(late fusion)	36.67	78.33	10.80	15.00		38.33	78.33	15.43	15.00	
Dual(Reformulated)	41.38	82.76	11.04	22.41		38.60	78.95	13.89	19.30	
Meta-Prefix	55.00	83.33	10.22	3.33		65.00	93.33	8.46	5.00	
Dual(Unified)	63.33	88.33	7.84	3.33		60.00	83.33	8.84	6.67	

(b) BAAI `bge-m3`

Method	General					Deeper				
	Context@5 ↑	Title@5 ↑	Avg	Rank ↓	Failure ↓	Context@5 ↑	Title@5 ↑	Avg	Rank ↓	Failure ↓
No Metadata	26.67	93.33	16.45	30.00		33.33	95.00	15.95	31.67	
Meta-Suffix	33.33	95.00	17.93	6.67		45.00	95.00	11.00	11.67	
Dual(late fusion)	30.00	88.33	13.67	35.00		31.67	91.67	19.37	31.67	
Dual(reformulated)	25.86	86.21	15.74	34.48		26.32	94.74	16.94	40.35	
Meta-Prefix	61.67	96.67	6.97	3.33		58.33	95.00	10.98	5.00	
Dual(Unified)	55.00	93.33	10.73	8.33		51.67	95.00	10.09	10.00	

lookups and latency. Still, the α -sweep is useful diagnostically to gauge metadata’s marginal value (see Figure 5).

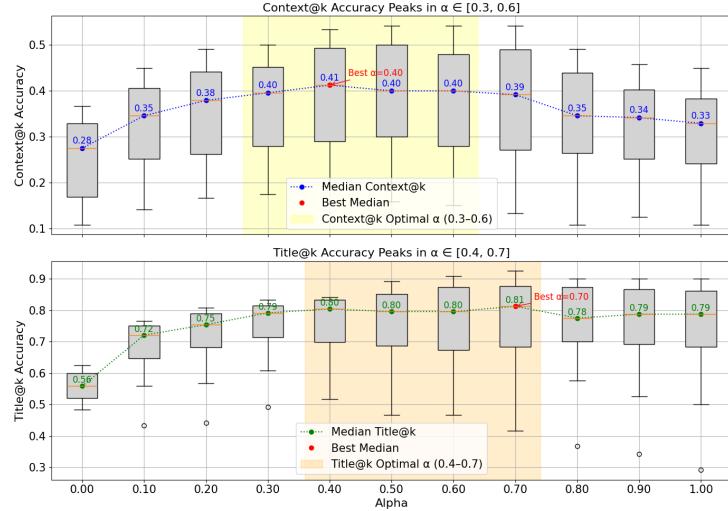


Fig. 5: Retrieval performance across metadata weight α for Dual encoder with late-fusion scoring. Metadata improves results when moderately weighted; full reliance on either content or metadata reduces performance.

Across both embedding models, OpenAI’s `text-embedding-3-small` (1536-dim) and BAAI’s `bge-m3` (1024-dim), the metadata-enriched strategies consis-

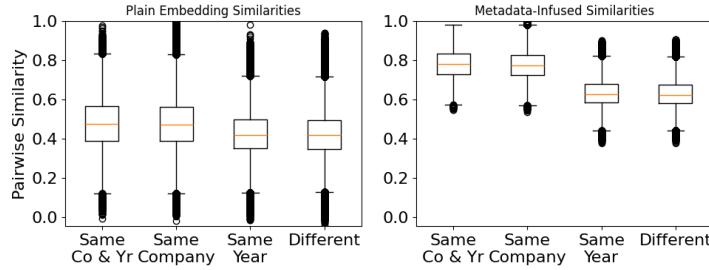


Fig. 6: Embedding space analysis with unified embeddings (grouped intra-/inter-document similarities)

tently outperform the plain baseline. The gains are most pronounced for complex queries that require company- or section-level disambiguation. Figures 1,3 and 4 visualize these comparative performance trends across query types.

Table 2 reports representative metrics at $k=5$, providing a compact numerical view of the performance differences across methods. These headline results set the stage for deeper analyses of trade-offs and ablations in the following sections.

5.1 Analysis of Embedding Space

We test Propositions 1–3 by computing pairwise cosine similarities under both metadata-as-text (MaT) and unified embeddings (Eq. 1). For each pair of chunks we form two similarities, plain $\cos(\mathbf{e}_i, \mathbf{e}_j)$ and metadata-enriched $\cos(\tilde{\mathbf{e}}_i^{\text{MaT}}, \tilde{\mathbf{e}}_j^{\text{MaT}})$ or $\cos(\tilde{\mathbf{e}}_i^{\text{Unif}}, \tilde{\mathbf{e}}_j^{\text{Unif}})$. We stratify pairs into Same Company & Year (positive) and Different (negative), and quantify separation.

Below we report results for the dual-encoder unified embeddings; the MaT variant shows similar trends. Metadata increases similarity across all strata but especially for positive pairs, widening the gap. For example, the mean margin between positives and negatives triples ($0.054 \rightarrow 0.152$), Cohen’s d grows from a small effect (0.45) to a very large one (2.25), and AUC rises from 0.63 to 0.94. As

Table 3: Separation between Same Company & Year (pos) and Different (neg) pairs, computed from pairwise cosine similarities. We report plain baseline values and relative improvements (Δ) for unified and prefix embeddings. Arrows indicate whether higher (\uparrow) or lower (\downarrow) values are better.

Metric	Plain	Δ	Dual(Unified)	Δ	Meta-Prefix
Mean margin (\uparrow)	0.054		+0.098		+0.101
Cohen’s d (\uparrow)	0.450		+1.800		+1.440
Fisher score F (\uparrow)	0.102		+2.425		+1.692
AUC (\uparrow)	0.625		+0.315		+0.283
KS distance (\uparrow)	0.190		+0.520		+0.461
Tail mass $\Pr[\cos \geq 0.8]$, pos (\uparrow)	0.007		+0.407		+0.108
Tail mass $\Pr[\cos \geq 0.8]$, neg (\downarrow)	0.000		+0.006		+0.000

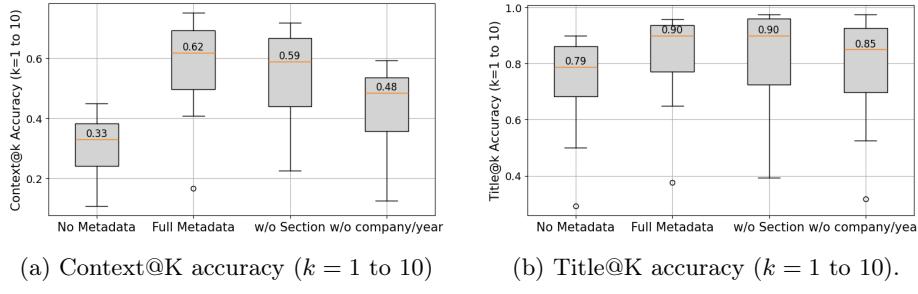


Fig. 7: Impact of metadata ablations on retrieval performance

Table 3 shows, both metadata strategies yield large gains over plain embeddings, with unified embeddings outperforming prefixing on majority of all metrics while being easier to maintain.

6 Impact of Metadata Fields: Ablation Study

Not all metadata fields contribute equally to retrieval performance. Here we focus on two types of signals: global identifiers such as company and year, and local context from section titles.

Chunks extracted from long, repetitive documents like SEC filings often lose their contextual anchor. For example, the phrase “we believe our strategy is working” has very different implications in a “Risk Factors” section versus in “Management’s Discussion and Analysis.” We distinguish two types of context: i) global, provided by fields such as company and year that identify the document as a whole, and ii) local, provided by fields such as section titles that situate a chunk within its broader structure. To test their contributions, we compare four conditions: i) No Metadata (Baseline): plain chunk text without metadata; ii) Full Metadata: all fields, including section titles; iii) w/o section: Full metadata except section titles. iv) w/o company and year: Full metadata except company and year. All embeddings use OpenAI’s `text-embedding-3-small` model with the MaT formulation, and retrieval is evaluated with the metrics from Section 4, focusing on Context@K and Title@K.

Figures 7a and 7b report the results. Company and year provide the strongest disambiguating signal: removing them reduces both Title@K and Context@K. In contrast, removing section titles yields only a modest drop in Context@K with no effect on Title@K, suggesting that global identifiers drive document-level accuracy while section cues primarily aid chunk-level localization.

7 Conclusion, Limitations, and Future Work

We revisited the role of metadata as a first-class retrieval signal in RAG, using SEC filings as a testbed. Across two embedding models, we found that embedding metadata alongside content consistently improves retrieval. Simple prefixing of

metadata is strong, but a dual-encoder with unified embeddings matches or exceeds its accuracy while being easier to maintain. Field-level ablations show that company and year act as strong disambiguators, while section titles are only modestly useful, and embedding-space analysis reveals that metadata improves geometry by tightening intra-document similarity and reducing cross-document confusion.

We also acknowledge limitations. Our study focuses on SEC Form 10-K filings, a deliberately challenging stress-test corpus characterized by rigid templates, heavy lexical repetition, and subtle document-level distinctions that frequently confound semantic-only retrieval. While absolute gains may vary across domains, these worst-case conditions allow us to isolate the effect of metadata integration, and we expect the relative benefits of treating metadata as a first-class retrieval signal, particularly via unified embeddings, to generalize to other structured corpora such as scientific articles, legal records, and technical manuals. Ground-truth answers were generated in a semi-supervised manner, a limitation that nevertheless reflects the common practice of using LLMs as judges or oracles [34, 13, 23, 16, 18]. In addition, we evaluate frozen encoders without exploring fine-tuning or end-to-end training.

Overall, our results suggest that effective metadata integration does not require complex architectures: concatenation or unified fusion already offer a strong balance of accuracy and practicality. Future directions include adaptive weighting, richer metadata modalities (tables, figures), and human evaluation of downstream generation quality.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Introducing Contextual Retrieval, <https://www.anthropic.com/news/contextual-retrieval>
2. Vector embeddings - OpenAI API, <https://platform.openai.com>
3. BAAI/bge-m3 · Hugging Face (Sep 2025), <https://huggingface.co/BAAI/bge-m3>
4. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
5. Asai, A., Wu, Z., Wang, Y., Sil, A., Hajishirzi, H.: Self-rag: Learning to retrieve, generate, and critique through self-reflection (2024)
6. Baek, J., Jeong, S., Kang, M., Park, J.C., Hwang, S.: Knowledge-augmented language model verification. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. pp. 1720–1736 (2023)
7. BenYoash, N., Brief, M., Ovadia, O., Shenderovitz, G., Mishaeli, M., Lemberg, R., Sheetrit, E.: Secque: A benchmark for evaluating real-world financial analysis capabilities. In: Proceedings of the Fourth Workshop on Generation, Evaluation and Metrics (GEM²). pp. 212–230 (2025)
8. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. Computer networks and ISDN systems **30**(1-7), 107–117 (1998)

9. Chen, P.B., Wolfson, T., Cafarella, M., Roth, D.: Enrichindex: Using llms to enrich retrieval indices offline. arXiv preprint arXiv:2504.03598 (2025)
10. Chen, Z., Chen, W., Smiley, C., Shah, S., Borova, I., Langdon, D., Moussa, R., Beane, M., Huang, T.H., Routledge, B.R., et al.: Finqa: A dataset of numerical reasoning over financial data. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. pp. 3697–3711 (2021)
11. Cuconasu, F., Trappolini, G., Siciliano, F., Filice, S., Campagnano, C., Maarek, Y., Tonello, N., Silvestri, F.: The power of noise: Redefining retrieval for rag systems. In: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 719–729 (2024)
12. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers). pp. 4171–4186 (2019)
13. Fu, J., Ng, S.K., Jiang, Z., Liu, P.: Gptscore: Evaluate as you desire. In: Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers). pp. 6556–6576 (2024)
14. Gao, L., Ma, X., Lin, J., Callan, J.: Precise zero-shot dense retrieval without relevance labels. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1762–1777 (2023)
15. Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, H., Wang, H.: Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997 **2**(1) (2023)
16. Hackl, V., Müller, A.E., Granitzer, M., Sailer, M.: Is gpt-4 a reliable rater? evaluating consistency in gpt-4’s text ratings. In: Frontiers in Education. vol. 8, p. 1272229. Frontiers Media SA (2023)
17. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: 2008 Eighth IEEE international conference on data mining. pp. 263–272. Ieee (2008)
18. Huang, H., Bu, X., Zhou, H., Qu, Y., Liu, J., Yang, M., Xu, B., Zhao, T.: An empirical study of llm-as-a-judge for llm evaluation: Fine-tuned judge model is not a general substitute for gpt-4. In: Findings of the Association for Computational Linguistics: ACL 2025. pp. 5880–5895 (2025)
19. Islam, P., Kannappan, A., Kiela, D., Qian, R., Scherrer, N., Vidgen, B.: Financebench: A new benchmark for financial question answering. arXiv preprint arXiv:2311.11944 (2023)
20. Lai, V., Krumdick, M., Lovering, C., Reddy, V., Schmidt, C., Tanner, C.: Secqa: A systematic evaluation corpus for financial qa. In: Proceedings of The 10th Workshop on Financial Technology and Natural Language Processing. pp. 221–236 (2025)
21. Lee, C.T., Neeser, A., Xu, S., Katyan, J., Cross, P., Pathakota, S., Norman, M., Simeone, J.C., Chandrasekaran, J., Ramakrishnan, N.: Can an llm find its way around a spreadsheet? In: Proceedings of the 2025 IEEE/ACM 47th International Conference on Software Engineering (ICSE). ICSE ’25, IEEE Computer Society, Ottawa, ON, Canada (2025), <https://www.computer.org/csdl/proceedings/icse/2025/056900a638/251mGdNO8uY>
22. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T., et al.: Retrieval-augmented generation for

- knowledge-intensive nlp tasks. *Advances in neural information processing systems* **33**, 9459–9474 (2020)
23. Li, J., Sun, S., Yuan, W., Fan, R.Z., Zhao, H., Liu, P.: Generative judge for evaluating alignment. arXiv preprint arXiv:2310.05470 (2023)
 24. Li, L., Chu, W., Langford, J., Schapire, R.E.: A contextual-bandit approach to personalized news article recommendation. In: Proceedings of the 19th international conference on World wide web. pp. 661–670 (2010)
 25. Lin, X.V., Chen, X., Chen, M., Shi, W., Lomeli, M., James, R., Rodriguez, P., Kahn, J., Szilvassy, G., Lewis, M., et al.: Ra-dit: Retrieval-augmented dual instruction tuning. In: The Twelfth International Conference on Learning Representations (2023)
 26. Luo, L., Li, Y.F., Haffari, G., Pan, S.: Reasoning on graphs: Faithful and interpretable large language model reasoning. arXiv preprint arXiv:2310.01061 (2023)
 27. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Tech. rep., Stanford infolab (1999)
 28. Ryu, C., Lee, S., Pang, S., Choi, C., Choi, H., Min, M., Sohn, J.Y.: Retrieval-based evaluation for llms: a case study in korean legal qa. In: Proceedings of the Natural Legal Language Processing Workshop 2023. pp. 132–137 (2023)
 29. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023)
 30. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. *Advances in neural information processing systems* **30** (2017)
 31. Xie, Q., Han, W., Chen, Z., Xiang, R., Zhang, X., He, Y., Xiao, M., Li, D., Dai, Y., Feng, D., et al.: Finben: A holistic financial benchmark for large language models. *Advances in Neural Information Processing Systems* **37**, 95716–95743 (2024)
 32. Yousuf, R.B., Defelice, N., Sharma, M., Xu, S., Ramakrishnan, N.: Llm augmentations to support analytical reasoning over multiple documents. In: 2024 IEEE International Conference on Big Data (BigData). pp. 1892–1901. IEEE (2024)
 33. Zakka, C., Shad, R., Chaurasia, A., Dalal, A.R., Kim, J.L., Moor, M., Fong, R., Phillips, C., Alexander, K., Ashley, E., et al.: Almanac—retrieval-augmented language models for clinical medicine. *Nejm ai* **1**(2), A1oa2300068 (2024)
 34. Zheng, L., Chiang, W.L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., et al.: Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems* **36**, 46595–46623 (2023)
 35. Zhou, D., Schärli, N., Hou, L., Wei, J., Scales, N., Wang, X., Schuurmans, D., Cui, C., Bousquet, O., Le, Q., et al.: Least-to-most prompting enables complex reasoning in large language models. arXiv preprint arXiv:2205.10625 (2022)