DeFuzzRAG: Handling Fuzzy Time Expressions for Temporal Robustness in Retrieval-Augmented Generation

Ling-Chun Chen, Hsi-Wen Chen, Ming-Syan Chen

National Taiwan University

{lcchen, hwchen}@arbor.ee.ntu.edu.tw, mschen@ntu.edu.tw

Abstract

Large Language Models (LLMs) have achieved remarkable success across reasoning and knowledge-intensive tasks, yet their static pretraining leaves them unable to handle rapidly evolving or domain-specific knowledge. Retrieval-Augmented Generation (RAG) addresses this by grounding LLM outputs in dynamically retrieved evidence, improving factual accuracy and reducing hallucinations. However, standard RAG pipelines struggle with temporally sensitive queries, especially when documents contain fuzzy or indirect time expressions (e.g., "a few years later"). This leads to Temporal Misalignment, where topically relevant but temporally incorrect results are retrieved. To overcome this, we propose DeFuzzRAG, a lightweight framework that enhances temporal robustness in RAG. DeFuzzRAG employs a small local language model to infer concrete time scopes from vague expressions and applies metadata-based filtering to realign retrieval with the query's temporal intent. Experiments on a benchmark of fuzzified queries demonstrate that DeFuzzRAG substantially improves retrieval accuracy, raising Hit Rate by 15.7% while maintaining efficiency and model-agnostic integration. Our findings highlight the importance of temporal reasoning in RAG and establish DeFuzzRAG as a practical, plug-and-play solution for deploying temporally robust LLM systems in real-world settings.

Introduction

Large Language Models (LLMs) (Achiam et al. 2023) have greatly advanced natural language processing (NLP), achieving strong results in open-domain question answering, reasoning, and summarization (Brown et al. 2020; Zhao et al. 2023). By leveraging large-scale pretraining, they internalize vast textual knowledge, enabling broad applicability. However, LLMs remain limited to the static knowledge captured during pretraining, drawn from fixed web snapshots or curated corpora (Lazaridou et al. 2022). This prevents them from incorporating new or domain-specific information—a critical shortcoming in rapidly evolving fields. Although continual retraining or fine-tuning could alleviate this, such methods are often impractical due to high computational and operational costs (Patterson et al. 2021). To address this, Retrieval-Augmented Generation (RAG) (Lewis et al. 2020) has emerged as a practical solution. By integrating dynamically updated external knowledge sources (e.g., search engines or vector databases), RAG grounds LLM outputs in factual evidence, reducing hallucinations and improving both response accuracy (Gao et al. 2023). and temporal relevance (Shuster et al. 2021).

Retriever	Setting	HR	MRR	Δ HR(%)	Δ MRR(%)
MPNet	Explicit	0.7908	0.5428	-15.1%	-14.7%
	Fuzzy	0.6712	0.4729		
BM25	Explicit	0.5574	0.3793	-17.7%	-28.7%
	Fuzzy	0.4587	0.2947		
BGE	Explicit	0.7900	0.5818	-7.3%	-9.9%
	Fuzzy	0.7326	0.5290		

Table 1: Retrieval performance on TempFuzz-QA is evaluated under both explicit and fuzzy settings. Here, HR measures retrieval coverage, while MRR reflects the final QA accuracy of the RAG system. We report results for explicit queries (with precise temporal anchors) and their fuzzy counterparts. The metrics ΔHR and ΔMRR denote the relative performance drops when moving from explicit to fuzzy queries.

Yet, RAG's effectiveness depends critically on retrieval quality (Karpukhin et al. 2020). Most systems rely on semantic similarity, which performs well in general scenarios but often fails under hard temporal constraints (Karpukhin et al. 2020). For example, the query *"What was the main U.S. foreign policy doctrine just before the Cold War?"* may incorrectly yield the *"Truman Doctrine,"* which arose after the war began, instead of the temporally accurate *"Monroe Doctrine"* or preceding isolationist policies. Such errors are amplified by vague or indirect temporal expressions (James 2003), such as "a few years before the outbreak" or "by the end of the 19th century," which lack explicit date markers.

As shown in Table 1, all RAG retrievers suffer a marked performance drop under fuzzy temporal queries. We define this issue as *Temporal Misalignment*—a mismatch between the query's temporal intent and the retrieved documents. This limitation means that systems relying solely on semantic similarity often return thematically relevant but temporally inaccurate results. Such misalignments are especially problematic in time-sensitive domains like finance (Yu, Chen, and Lu 2023), healthcare (Thirunavukarasu et al. 2023), or marketing (Kasuga and Yonetani 2024), where ac-

curate timing is critical for decision-making.

To better identify and address this issue, we introduce the **Time Attention Ratio (TAR)**, a metric that quantifies how strongly models attend to temporally relevant tokens during response generation. Building on this insight, we propose **DeFuzzRAG**, an efficient framework for time-aware retrieval-augmented generation. Rather than relying on timestamped corpora or retriever modifications, DeFuzzRAG enforces temporal consistency after the initial semantic retrieval. The framework operates in three stages: (1) extracting temporal intent from queries, (2) enriching retrieved documents with explicit temporal metadata, and (3) applying a temporal matching filter to re-rank or discard inconsistent passages. To keep the approach lightweight, temporal scopes are inferred on-the-fly using a small language model (SLM) that converts fuzzy expressions into concrete time ranges. Crucially, DeFuzzRAG avoids retraining either the retriever or the generator, enabling seamless integration into existing RAG pipelines with minimal overhead. Experimental results show that DeFuzzRAG improves Hit Rate by 15.7% and raises TAR by an average of 27.7%, indicating substantially stronger temporal focus.

Related Work

Temporally-Aware RAG. Retrieval-Augmented Generation (RAG) (Lewis et al. 2020) is central to knowledge-intensive LLM applications (Achiam et al. 2023), typically using dense retrievers such as DPR (Karpukhin et al. 2020). While effective in open-domain tasks (Guu et al. 2020; Izacard and Grave 2020; Izacard et al. 2023), most systems treat retrieval as purely semantic matching, overlooking temporal alignment—a known issue in information freshness (Vu et al. 2023). This limitation is critical in time-sensitive domains such as finance (Yu, Chen, and Lu 2023), healthcare (Thirunavukarasu et al. 2023), and marketing (Kasuga and Yonetani 2024). Recent work addresses this by incorporating temporal signals: TempRALM (Gade and Jetcheva 2024) adds temporal proximity scores, and TimeLM (Dhingra et al. 2022) uses timestamp prefixes for time-aware pre-training. Temporal-sensitive QA datasets (Chen et al. 2025) and time-aware retrievers that index temporal metadata (Wu et al. 2024) further increase precision but require corpus-wide metadata or retriever modification, and still rely on embedding similarity, consistent with Theorem 1. In contrast, DeFuzzRAG performs temporal alignment post-retrieval without retraining.

Temporal Data Mining. Temporal Information Retrieval (Campos et al. 2014) has explored temporal relevance in ranking (Feng et al. 2019), query interpretation (Kulkarni et al. 2011), and document dating (Kanhabua and Nørvåg 2009). Recent extensions, including temporal knowledge graphs (Saxena, Chakrabarti, and Talukdar 2021) and adaptive retrieval methods (Qian et al. 2024), capture richer temporal relations but rely on structured annotations, limiting robustness on noisy real-world text. In parallel, Temporal Information Extraction (TIE) focuses on identifying and normalizing vague temporal expressions. TimeML (Pustejovsky et al. 2003) established a standardized annotation schema, HeidelTime (Strötgen et al. 2014) introduced multilingual rule-based normalization, and SUTime (Strötgen and Gertz 2013) provided deterministic patterns for mapping phrases such as "last month" into normalized intervals. More recent work, such as E2E-MHA (Miller et al. 2023), uses transformer models for end-to-end temporal relation extraction, reducing reliance on handcrafted rules. While effective, these systems remain decoupled from retrieval and generation pipelines and do not resolve temporal fuzziness in unstructured data.

Preliminary

RAG systems retrieve semantically relevant content but often fail to capture the temporal intent in user queries, especially when documents express time vaguely, implicitly, or indirectly rather than through explicit timestamps. We argue that these failures reflect a deeper representational weakness, *Temporal Misalignment*, in which topically relevant documents are missed because their temporal expressions do not align with the query's timeframe. To understand how this arises, we analyze common fuzzy temporal expressions in historical, journalistic, and encyclopedic texts. As summarized in Table 2, these expressions fall into four categories: (i) *Generalized Dating* (GD), where time is given broadly or seasonally; (ii) *Ambiguous Temporal Markers* (AMT), involving underspecified or inherently vague cues; (iii) *Event Recasting* (ER), where temporal information appears through paraphrase or contextual reference; and (iv) *Relative Time Framing* (RTF), which uses coarse relational markers instead of precise timestamps.

While the taxonomy captures the breadth of temporal fuzziness, it does not show how these patterns impact retrieval. Standard IR metrics (e.g., recall, rank) (Gao et al. 2023; Guu et al. 2020) conflate topical relevance with temporal alignment, obscuring the source of errors. To isolate temporal factors, we introduce the *Time Attention Ratio (TAR)*, an interpretability metric inspired by attention-based analyses (Hu et al. 2023; Clark et al. 2019). TAR measures the share of model attention flowing from temporal query tokens to temporal document tokens, normalized by attention from all informative query tokens:

$$\text{TAR} = \frac{\sum_{t \in T} \sum_{d \in D_{\text{time}}} \text{Attention}(t, d)}{\sum_{q \in Q'} \sum_{d \in D} \text{Attention}(q, d)},$$

where T represents temporal tokens in the query, Q' the informative query tokens, D_{time} the temporal tokens in the document, and D the informative document tokens. By isolating attention on temporal cues, TAR supplements traditional IR metrics and offers a direct measure of temporal alignment in RAG systems.

We evaluate TAR on 1,000 query–document pairs, each containing an *explicit* version with precise temporal anchors and a *fuzzy* version with vague references. This setup controls topical relevance while isolating temporal fuzziness. Fuzzified documents exhibit a 27.7% average drop in TAR, typically accompanied by lower retrieval ranks, indicating that vague temporal language systematically weakens the model's attention to time-relevant cues. Table 3 illustrates

Type	Description	Explicit Expression	Fuzzy Expression
Generalized Dating	Express in broad or seasonal terms rather than exact dates.	"28 July 1914"	"summer of 1914"
Ambiguous Temporal Markers	Use underspecified or inherently ambiguous temporal terms.	"from 1990 to 1995"	"in the following years"
Event Recasting	Named temporal anchors are paraphrased or described indirectly.	"October Revolution"	"later that year the Bolsheviks seized power"
Relative Time Framing	Exact dates are expressed through coarse or relational markers.	"in 1911"	"in the early 20th century"

Table 2: Taxonomy of fuzzy temporal expression types.

Query	Expression	Document	Rank	TAR
What were some significant events of the Cold War before 1975?	Explicit	"The 1908–1909 Bosnian Crisis began..."	3	0.3374
	Fuzzy	"At the turn of the 20th century..."	18	0.2336
What major military campaigns happened in World War II between 1941 and 1944?	Explicit	"In June 1941, Germany led an invasion..."	2	0.4942
	Fuzzy	"Around the mid-1940s, Germany led an invasion..."	15	0.1767

Table 3: Case studies for analyzing temporal misalignment with TAR.

this effect: for a Cold War query before 1975, the explicit document ("The 1908–1909 Bosnian Crisis began...") ranks 3 with TAR 0.3374, while its fuzzy counterpart ("At the turn of the 20th century...") falls to rank 18 with TAR 0.2336. For a World War II query, the explicit document ("In June 1941...") ranks 2 with TAR 0.4942, whereas the fuzzy version ("Around the mid-1940s...") drops to rank 15 with TAR 0.1767. These cases confirm that temporal misalignment arises systematically from vague temporal expressions despite preserved topical relevance.

Beyond fuzziness, temporal embeddings themselves fail to encode ordering. Common retrievers (Johnson, Douze, and Jégou 2019; Song et al. 2020), typically based on symmetric similarities (inner product, cosine, or L_p norms), treat (t_i, t_j) and (t_j, t_i) identically, making it impossible to decide which time precedes the other.

Theorem 1 (Impossibility of Temporal Comparison via Vector Embeddings)**.** *Let $\mathcal{T}$ be a totally ordered set for timestamp and $\{\phi_m : \mathcal{T} \to \mathbb{R}^{n_m}\}_{m=1}^{k}$ be any finite family of embeddings (dimensions n_m may differ). Let $s : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be a symmetric similarity, i.e., $s(u, v) = s(v, u)$ for all u, v in the appropriate space; for each m, interpret s on $\mathbb{R}^{n_m}$. For any decision rule $H : \mathbb{R}^k \to \{<, =, >\}$ that depends only on the vector of similarities,*

$$D(t_i, t_j) = H\big(s(\phi_1(t_i), \phi_1(t_j)), \ldots, s(\phi_k(t_i), \phi_k(t_j))\big),$$

the induced D is symmetric and therefore cannot correctly decide temporal direction on all unequal pairs; i.e., there exist $t_i \neq t_j$ with $D(t_i, t_j)$ not equal to the true directional label in $\{<, >\}$.

Temporal fuzziness amplifies the structural limitation identified in Theorem 1: vague expressions reduce separability along the time dimension, while symmetric similarity functions cannot restore lost directionality because they score (t_i, t_j) and (t_j, t_i) identically. Empirically, this results in lower TAR and degraded ranks despite preserved topical relevance. A practical remedy is to decouple temporal reasoning from semantic similarity, e.g., via explicit temporal intent extraction, document span inference, and asymmetric filtering before re-ranking.

Methodology

Building on this analysis, and as shown in Figure 1, we introduce **DeFuzzRAG**, a framework that decouples topical relevance from temporal alignment. Instead of relying solely on symmetric similarity scores, DeFuzzRAG performs on-the-fly temporal parsing to generate explicit metadata and enforce temporal consistency after semantic retrieval. The framework has three stages: (1) *Temporal Intent Extraction*, which converts the query's temporal intent into a structured constraint; (2) *Query-Aware Temporal Metadata Generation*, which resolves vague or implicit temporal references in retrieved documents into explicit time ranges; and (3) *Temporal Alignment Filtering*, which compares query constraints with document metadata to remove misaligned results, restoring the antisymmetric directionality missing in embedding-based similarity.

Temporal Intent Extraction

Conventional RAG pipelines rely on dense retrievers (Karpukhin et al. 2020) that rank documents purely by semantic similarity. While effective for topical relevance, they are blind to temporal constraints and often surface documents that are semantically related but temporally misaligned. This is especially problematic for temporally scoped queries (e.g., "before 2000," "in 1975"), where correctness requires both semantic match and precise temporal grounding.

To address this, DeFuzzRAG explicitly extracts the query's temporal intent T_q from the input query Q. Representing this intent before retrieval provides a principled basis for evaluating temporal validity. For example, *"What events occurred in 1975?"* is normalized to $[\texttt{year}, 1975, 1975]$, ensuring that only documents anchored within this interval are considered. Such normalization is essential for de-

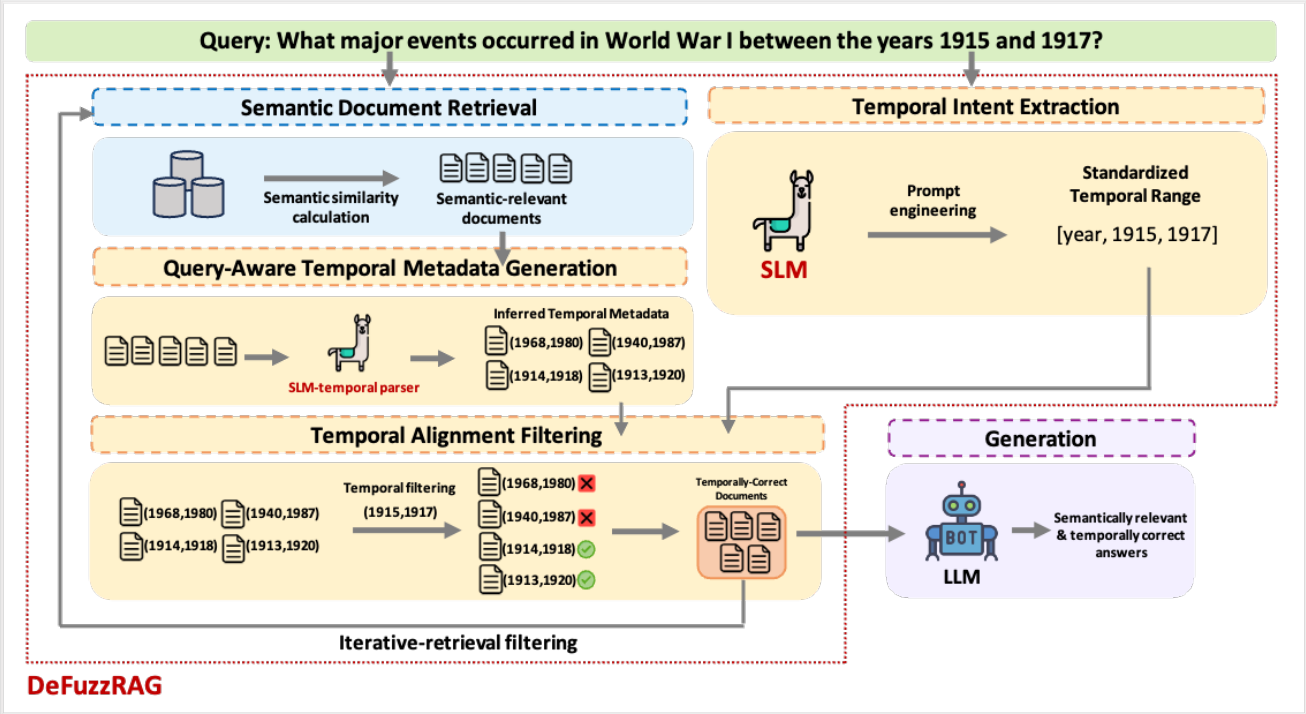

Figure 1: An overview of the DeFuzzRAG.

termining whether documents with vague or underspecified expressions satisfy the query's temporal requirements. We implement temporal intent extraction using a small language model (SLM) (Schick and Schütze 2021) guided by prompt templates. Note that the model adaptively infers temporal granularity (year-, month-, or finer-level) based on query cues, and supports both single years and compound intervals (e.g., "after A but before B") for precise comparison during temporal filtering.

Query-Aware Temporal Metadata Generation

After extracting the query's temporal intent T_q, we next derive explicit temporal ranges for each retrieved document so they can be compared against the query's requirements. Prior time-aware RAG methods (Qian et al. 2024; Gade and Jetcheva 2024; Zhang et al. 2024) attach predefined metadata to documents, but such corpus-wide annotation is inflexible, costly, and struggles with the vague or underspecified temporal expressions common in natural text.

To address this, we introduce a temporal parsing module that dynamically generates structured temporal metadata for the top-K retrieved candidates $d \in \mathcal{D}_{candidate}$. For each document d, explicit expressions (e.g., "in 1975") are extracted directly; when absent or vague (e.g., "a few years after the war"), a small language model (SLM) (Zhang et al. 2024; Wang et al. 2023) performs *query-aware temporal inference*, using cues from the query Q to map the document into a normalized interval T_d (e.g., resolving "a few years after the war" to [1946, 1949] for a WWII query).

By combining explicit extraction with context-sensitive inference, the module produces machine-interpretable metadata that supports reliable temporal filtering. Importantly, T_d does not modify initial retrieval scores; it simply provides the temporal signal necessary to enforce alignment during re-ranking. This design avoids costly corpus-level preprocessing and remains modular, allowing other metadata dimensions (e.g., events, locations, entities) to be incorporated without additional preprocessing.

Temporal Alignment Filtering

The final stage of the DeFuzzRAG framework is *Temporal Alignment Filtering*, which ensures that retrieved documents satisfy both topical relevance and temporal validity. Specifically, the module performs a structured comparison between each candidate document's inferred temporal scope T_d and the query's temporal specification T_q. A document is retained if and only if its temporal interval overlaps with the query constraint, i.e., $T_d \cap T_q \neq \emptyset$. The overlap-based criterion is deliberately chosen to accommodate vague or underspecified temporal expressions, preserving partially relevant evidence while filtering out clearly misaligned documents.

The overlap-based criterion is chosen to accommodate fuzzy temporal language, retaining documents that partially satisfy the constraint while discarding clearly misaligned ones. To avoid under-retrieval, we use an *iterative retrieval–filtering loop*: if fewer than N aligned documents remain, additional candidates are retrieved and rechecked until $|\mathcal{D}_{aligned}| \geq N$ or the pool is exhausted. This post hoc design limits computation to the top-K candidates, remains efficient, and supports modular extensions (e.g., event

or location constraints) without re-indexing. By decoupling semantic relevance from temporal validity, Temporal Alignment Filtering improves factual accuracy and reduces hallucinations. Although distance-based scoring is possible, it requires consistent granularity and may reward close but irrelevant spans; we therefore adopt binary overlap to preserve topical relevance without costly timestamp comparisons.

Illustrative Example

To show the procedure of DeFuzzRAG, we present a concrete example from its first retrieval iteration using the query: *"What were some significant events of the Cold War before 1975?"* This query requires both semantic relevance and adherence to a well-defined temporal constraint.

Step 1: Temporal Intent Extraction. The system first parses the query and extracts the temporal constraint: $T_q = [year, -, 1975]$ This structured output indicates that only documents describing events that occurred before 1975 should be considered temporally relevant.

Step 2: Query-Aware Temporal Metadata Construction. Using a dense retriever (e.g., MPNet), the system retrieves semantically relevant candidates regardless of temporal alignment. For this query, the following top-5 documents are retrieved:

- **D1:** "The late 70s and early 80s marked the renewal of tensions and conflicts..."
  Inferred Time: (1978, 1985)
- **D2:** "Nixon and Brezhnev announced a new era of "peaceful coexistence"... during the 1970s"
  Inferred Time: (1970, 1979)
- **D3:** "Reagan went all out to fight the second cold war..."
  Inferred Time: (1980, 1989)
- **D4:** "President Jimmy Carter tried to place another limit on the arms race... in the late 70s"
  Inferred Time: (1978, 1980)
- **D5:** "Despite the distance in time, the alliances during the Napoleonic era mirror the Cold War..."
  Inferred Time: (1799, 1991)

Each document is then processed by the metadata construction module. Explicit time expressions are extracted when present, while vague phrases (e.g., "late 70s") are interpreted by a small language model into specific ranges. For instance, D5 spans an unusually broad range (1799–1991), as the document draws historical analogies between the Cold War and the Napoleonic era—prompting the parser to capture both periods as temporally relevant.

Step 3: Temporal Alignment Filtering. The system compares each document's inferred time range T_d with the query's constraint $T_q = [-, 1975]$:

- **D1**, **D3**, and **D4** are filtered out as their time spans begin after 1975.
- **D2** is retained as its time range (1970–1979) overlaps with the query target.
- **D5** is retained due to its span (1799–1991) covering the query constraint.

After filtering, the document set is $D_{\text{aligned}} = \{\textbf{D2, D5}\}$, showing that DeFuzzRAG retains only those candidates satisfying the temporal constraint, thereby ensuring results that are both semantically precise and temporally consistent.

Experiment

TempFuzz-QA

To rigorously evaluate DeFuzzRAG under conditions of temporal ambiguity, we construct a new benchmark, TempFuzz-QA, comprising 1,000 document–query pairs sampled from the October 2024 English Wikipedia dump. Each document is a paragraph-length passage describing a real-world event, paired with a query requiring alignment between the query's temporal intent and the document's temporal scope. The dataset was first sampled from 10k Wikipedia passages, reduced to 2k by LLM-as-a-Judge filtering, and finalized to 1,000 high-quality examples through joint agreement among multiple annotators. Each document–query pair in TempFuzz-QA contains two variants: an *explicit* version with precise temporal anchors and a *fuzzified* version with vague or underspecified temporal expressions, forming parallel corpora for controlled evaluation.

To systematically degrade temporal precision while preserving topical identity and event semantics, TempFuzz-QA defines four categories of fuzziness. (i) *Generalized Dating* (GD), where time is expressed in broad or seasonal terms rather than precise dates; (ii) *Ambiguous Temporal Markers* (AMT), which rely on underspecified or inherently vague expressions; (iii) *Event Recasting* (ER), where temporal cues are conveyed indirectly through paraphrase or contextual references; and (iv) *Relative Time Framing* (RTF), where time is indicated through coarse or relational markers instead of exact timestamps. We generate the fuzzified variants automatically using GPT-4o (Achiam et al. 2023), prompting it to rewrite temporal expressions according to these categories. This methodology ensures topical relevance is preserved while isolating the effect of temporal fuzziness on retrieval effectiveness.

Following prior work on temporal question answering (Jia et al. 2018), we define five representative query types to cover diverse temporal reasoning challenges: (1) *Year-Specific* (YS) (e.g., "What significant event in World War I occurred in 1917"), (2) *After-Type Range* (AF) (e.g., "What significant events happened in World War I after 1915"), (3) *Before-Type Range* (BF) (e.g., "What were the key events leading up to World War I before 1914"), (4) *Between-Type Range* (BT) (e.g., "What major events occurred in World War I between the years 1915 and 1917"), and (5) *Event-Anchored Timing* (ET) (e.g., "When did Kaiser Wilhelm II abdicate leading to the end of World War I"). These categories jointly capture explicit spans, relative constraints, and event-based anchors.

Experimental Setup

Baselines. We compare four representative baselines covering both sparse and dense retrieval paradigms, including *(i) BM25* (Robertson, Zaragoza et al. 2009): Sparse lexical

Model	HR_e	HR_f	ΔHR	R_e	R_f	ΔR	MRR_e	MRR_f	ΔMRR	Latency (s / query)
BM25	0.5574	0.4587	17.7%	0.3586	0.2927	18.4%	0.3793	0.2974	21.6%	1.56
MPNet	0.7908	0.6712	15.1%	0.5222	0.4792	8.2%	0.5428	0.4729	12.9%	0.67
Contriever	0.6656	0.6067	8.9%	0.4370	0.4219	3.5%	0.4402	0.4074	7.5%	0.79
BGE	0.7900	0.7426	6.0%	0.5578	0.5228	6.3%	0.5618	0.5290	5.1%	1.06
DeFuzzRAG - LLaMA2-7B	**0.8125**	0.7765	4.4%	0.5589	0.5282	5.5%	0.5674	0.5441	4.1%	2.33
DeFuzzRAG - OpenChat 3.5	0.8000	0.7874	1.6%	**0.5705**	**0.5491**	3.8%	0.5742	0.5359	6.7%	2.56
DeFuzzRAG - Mistral-7B	0.7967	0.7617	4.4%	0.5389	0.5280	2.0%	**0.5762**	0.5402	6.2%	2.84
DeFuzzRAG - TinyLLaMA-1.1B	0.7845	0.6932	11.6%	0.5129	0.4812	6.2%	0.5361	0.4702	12.2%	1.67
DeFuzzRAG - GPT-4	0.8082	**0.7895**	2.3%	0.5549	0.5343	3.7%	0.5714	**0.5483**	4.1%	11.14(API)

Table 4: Qualitative results on TempFuzz-QA, where $k = 5$ for all experiments.

retriever using TF–IDF weighting, serving as a strong traditional baseline. *(ii) MPNet* (Song et al. 2020): Dense bidirectional encoder pretrained with MLM and PLM, providing strong sentence-level embeddings. *(iii) Contriever* (Izacard et al. 2021): Unsupervised dense retriever trained via contrastive learning, effective in zero-shot settings. *(iv) BGE* (Chen et al. 2024): Open-source dense retriever pretrained on multilingual data, effective for open-domain retrieval. For a fair comparison no fine-tuning is applied to the retrievers, allowing us to directly assess their robustness under temporal ambiguity. Unlike temporal retrievers that require metadata indexing or retraining, DeFuzzRAG is applied post-retrieval on unmodified retrievers, yielding gains without fine-tuning. The retrieved documents are subsequently fed into a RAG pipeline powered by GPT-3.5 (Brown et al. 2020) , last updated November 2023, ensuring a clear separation between the retrieval and generation stages.

Evaluation Metrics We assess retrieval performance using three standard metrics widely adopted in information retrieval and open-domain QA (Voorhees, Tice et al. 1999), adapted here for temporal alignment. *(a) Hit Rate (HR)* measures the proportion of queries for which at least one temporally relevant document appears in the top-k results, indicating whether the retriever can return any evidence consistent with the query's temporal intent. *(b) Recall (R)* quantifies the fraction of all temporally relevant documents retrieved among the top-k candidates, averaged across queries, thus reflecting the comprehensiveness of temporal coverage. *(c) Mean Reciprocal Rank (MRR)* computes the average reciprocal rank of the first temporally relevant document, capturing both ranking quality and temporal precision by rewarding systems that surface aligned evidence near the top. Note that we report performance separately on the explicit dataset (subscript e) and the fuzzified dataset (subscript f), together with the relative difference Δ, which quantifies the impact of temporal fuzziness on model performance.

Implementation Details. All experiments are conducted within a locally hosted RAG pipeline implemented in Python, using LangChain (Topsakal and Akinci 2023), with all retrieval operations executed through FAISS (Douze et al. 2024). For temporal intent extraction, DeFuzzRAG employs four open-source small language models (SLMs): LLaMA-2 7B Chat (Touvron et al. 2023), Mistral-7B Instruct (Jiang 2024), OpenChat-3.5 (Wang et al. 2023), and TinyLLaMA-1.1B (Zhang et al. 2024). In addition, we evaluate GPT-4 (Achiam et al. 2023) as a reference to assess the performance gap between SLMs and state-of-the-art LLMs. These models are loaded via HuggingFace Transformers (Wolf et al. 2020) and executed locally on a single NVIDIA GPU with mixed-precision inference. Prompts are designed to normalize vague temporal phrases into explicit time intervals. For query-aware temporal metadata generation, we use MPNet (Song et al. 2020) as the embedding model to retrieve the top-5 candidate documents. Each candidate is enriched with temporal metadata, either extracted from explicit timestamps or inferred by the SLM parser, and retained if it satisfies the overlap criterion within the temporal scope. Finally, temporal alignment filtering is applied to finalize the top-5 documents, consistent with other baselines. To keep DeFuzzRAG lightweight, we use mid-sized open-source SLMs (e.g., LLaMA-2-7B, Mistral-7B) for temporal parsing, and apply lightweight caching to avoid redundancy.

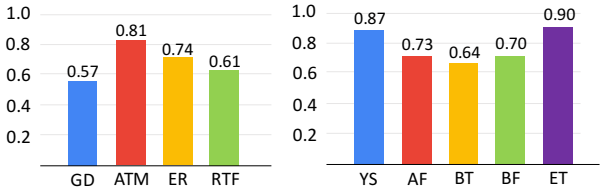

Figure 2: Performance breakdown by (a) fuzziness type and (b) query type in HR_f.

Quantative Results

We first evaluate all models on the explicit (clean) dataset to establish a baseline. As shown in Table 4, DeFuzzRAG variants achieve competitive or superior results across all metrics. For instance, DeFuzzRAG–LLaMA2-7B attains the highest HR_e (0.8125), while DeFuzzRAG–Mistral records the highest MRR_e (0.5762). These findings indicate that DeFuzzRAG generalizes effectively when temporal information is explicit, as it leverages temporal intent extraction and query-aware metadata construction to preserve align-

ment with the query's time scope rather than relying solely on similarity-based retrieval.

On the fuzzified dataset, DeFuzzRAG again delivers superior performance, with different variants excelling across metrics. DeFuzzRAG–OpenChat 3.5 achieves the highest HR_f (0.7895) and the strongest MRR_f (0.5491). By contrast, dense retrievers degrade sharply under fuzziness—for example, MPNet's Hit Rate drops by 15.1% and BM25's by 17.7%. DeFuzzRAG–OpenChat, however, shows only a 1.6% decline in ΔHR and 3.8% in ΔR, less than half of MPNet's losses, underscoring its robustness. Beyond mitigating fuzziness, DeFuzzRAG's temporal filtering acts as progressive search refinement. By removing semantically relevant but temporally misaligned documents after the initial retrieval, the framework concentrates the candidate pool on results that jointly satisfy topical and temporal constraints.

Comparing the variants of DeFuzzRAG, 7B-class models such as LLaMA2-7B and Mistral-7B strike the best balance, with Mistral-7B achieving near-GPT-4 performance in MRR_f (0.5402 vs. 0.5483) at roughly 75% lower latency. OpenChat 3.5 exhibits the greatest robustness to fuzziness, yielding the lowest ΔHR (1.6%) and Recall drop (3.8%), thereby sustaining stable top-5 coverage. By contrast, TinyLLaMA-1.1B, although faster at 1.67 s, suffers a sharp decline in MRR_f (0.4702), indicating insufficient reasoning capacity for temporal disambiguation. GPT-4 sets the performance ceiling with the highest HR_f and MRR_f, but its high latency (11.14 s). Notably, mid-sized open-source SLMs (7B) already deliver strong temporal inference with moderate latency (Table 4), indicating that DeFuzzRAG does not rely on large models. In summary, the mid-sized open-source SLMs are sufficient for accurate temporal intent extraction, offering a viable alternative to large proprietary models for robust temporal reasoning.

As shown in Figure 2a, hit rate varied substantially across fuzziness type. DeFuzzRAG performed best on *Relative Time Framing* (RTF) and *Ambiguous Temporal Markers* (ATM), where contextual cues enable effective temporal disambiguation. In contrast, *Generalized Dating* (GD) yielded the lowest hit rate, reflecting the difficulty of resolving broad or seasonal expressions without explicit anchors. These findings confirm our hypothesis that the degree of temporal structure in fuzziness directly impacts retrieval robustness. Turning to query types (Figure 2b), DeFuzzRAG achieved its highest performance on *Event-Anchored Timing* (ET) and *Year-Specific* (YS) queries (≈ 0.90 and ≈ 0.87). Both categories provide strong temporal anchors, either through explicit years or pivotal events—allowing precise alignment between queries and candidate documents. Other query types showed weaker results, consistent with Theorem 1 that retrieval effectiveness diminishes as temporal constraints become less explicit or span multiple intervals.

Extending to Spatial Fuzziness

Based on TempFuzzQA, we also introduce a benchmark, **SpaceFuzz-QA**, to assess retrieval robustness under *spatial fuzziness*. Similarity, SpaceFuzzQA comprises 1,000 query–document pairs that are spatially fuzzified across three representative categories: (i) *Coarse Regional Substitution*, where a precise location is replaced with a much broader geographic region (e.g., "France" $\rightarrow$ "Western Europe"), forcing the system to reason across larger, loosely defined spatial scopes; (ii) *Hierarchical Abstraction*, where entities are mapped to higher-level administrative or geopolitical units (e.g., "Berlin" $\rightarrow$ "Germany"), requiring sensitivity to geographic containment and administrative hierarchies; (iii) *Indirect Locational Phrasing*, where explicit names are replaced with descriptive or relational references (e.g., "the capital city" $\rightarrow$ "Paris"), demanding both contextual inference and integration of background knowledge.

Model	HR_f	R_f	MRR_f
BM25	0.4842	0.2965	0.3325
Contriever	0.6105	0.4316	0.3593
MPNet	0.7253	0.5133	0.5104
DeFuzzRAG	0.7834	0.5430	0.5367

Table 5: Retrieval performance under SpaceFuzz-QA.

As shown in Table 5, DeFuzzRAG yields at least 8.0% improvement in Hit Rate over all baselines, confirming that the framework's effectiveness extends beyond the temporal domain. The gain, however, is more modest than in our temporal experiments. This stems from the nature of spatial fuzziness: replacing specific locations with broader, semantically related regions (e.g., "Paris" $\rightarrow$ "France") often preserves high embedding similarity, giving baselines incidental robustness despite lacking true spatial reasoning. Their limits show in tasks like latitude/longitude, distance, or direction, where surface similarity fails to ensure accurate retrieval.

Conclusion

In this paper, we identify *Temporal Misalignment* as a key failure mode in RAG systems, where dense retrievers surface topically relevant but temporally inconsistent results due to vague or underspecified time expressions. To address this, we propose **DeFuzzRAG**, a lightweight, on-the-fly framework that (1) extracts temporal intent from queries, (2) generates structured temporal metadata for retrieved documents using a small local language model, and (3) applies Temporal Alignment Filtering to enforce consistency. On the TempFuzz-QA benchmark, DeFuzzRAG yields a 15.7% improvement in Hit Rate over standard RAG, without retriever fine-tuning or corpus preprocessing. Future work will examine additional forms of fuzziness in RAG and incorporate richer world knowledge and retriever adaptation to further improve robustness. Due to space constraints, our spatial evaluation appears only as a preliminary demonstration; a full study of spatial fuzziness is left for future work.

Acknowledgements

This work was supported by the National Science and Technology Council (NSTC), Taiwan, under Grants 114-2223-E-002-009 and 114-2221-E-002-180-MY3, and by the Higher Education Sprout Project, Ministry of Education (MOE) through the "Data Intelligence and Systems Research Center," National Taiwan University.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Campos, R.; Dias, G.; Jorge, A. M.; and Jatowt, A. 2014. Survey of temporal information retrieval and related applications. *ACM Computing Surveys (CSUR)*, 47(2): 1–41.

Chen, J.; Xiao, S.; Zhang, P.; Luo, K.; Lian, D.; and Liu, Z. 2024. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*.

Chen, Z.; Min, E.; Zhao, X.; Li, Y.; Jia, X.; Liao, J.; Li, J.; Wang, S.; Hu, B.; and Yin, D. 2025. A Question Answering Dataset for Temporal-Sensitive Retrieval-Augmented Generation. *arXiv preprint arXiv:2508.12282*.

Clark, K.; Khandelwal, U.; Levy, O.; and Manning, C. D. 2019. What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*.

Dhingra, B.; Cole, J. R.; Eisenschlos, J. M.; Gillick, D.; Eisenstein, J.; and Cohen, W. W. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10: 257–273.

Douze, M.; Guzhva, A.; Deng, C.; Johnson, J.; Szilvasy, G.; Mazaré, P.-E.; Lomeli, M.; Hosseini, L.; and Jégou, H. 2024. The faiss library. *arXiv preprint arXiv:2401.08281*.

Feng, F.; He, X.; Wang, X.; Luo, C.; Liu, Y.; and Chua, T.-S. 2019. Temporal relational ranking for stock prediction. *ACM Transactions on Information Systems (TOIS)*, 37(2): 1–30.

Gade, A.; and Jetcheva, J. 2024. It's About Time: Incorporating Temporality in Retrieval Augmented Language Models. *arXiv preprint arXiv:2401.13222*.

Gao, Y.; Xiong, Y.; Gao, X.; Jia, K.; Pan, J.; Bi, Y.; Dai, Y.; Sun, J.; Wang, H.; and Wang, H. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2(1).

Guu, K.; Lee, K.; Tung, Z.; Pasupat, P.; and Chang, M. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, 3929–3938. PMLR.

Hu, L.; Liu, Y.; Liu, N.; Huai, M.; Sun, L.; and Wang, D. 2023. Seat: stable and explainable attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 12907–12915.

Izacard, G.; Caron, M.; Hosseini, L.; Riedel, S.; Bojanowski, P.; Joulin, A.; and Grave, E. 2021. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Izacard, G.; and Grave, E. 2020. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*.

Izacard, G.; Lewis, P.; Lomeli, M.; Hosseini, L.; Petroni, F.; Schick, T.; Dwivedi-Yu, J.; Joulin, A.; Riedel, S.; and Grave, E. 2023. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251): 1–43.

James, P. 2003. TimeML: Robust specification of event and temporal expressions in text. In *Proceedings of the Fifth International Workshop on Computational Semantics (IWCS-5), 2003*.

Jia, Z.; Abujabal, A.; Saha Roy, R.; Strötgen, J.; and Weikum, G. 2018. Tempquestions: A benchmark for temporal question answering. In *Companion Proceedings of the The Web Conference 2018*, 1057–1062.

Jiang, F. 2024. *Identifying and mitigating vulnerabilities in llm-integrated applications*. Master's thesis, University of Washington.

Johnson, J.; Douze, M.; and Jégou, H. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3): 535–547.

Kanhabua, N.; and Nørvåg, K. 2009. Using temporal language models for document dating. In *Joint European conference on machine learning and knowledge discovery in databases*, 738–741. Springer.

Karpukhin, V.; Oguz, B.; Min, S.; Lewis, P. S.; Wu, L.; Edunov, S.; Chen, D.; and Yih, W.-t. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *EMNLP (1)*, 6769–6781.

Kasuga, A.; and Yonetani, R. 2024. Cxsimulator: A user behavior simulation using llm embeddings for web-marketing campaign assessment. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 3817–3821.

Kulkarni, A.; Teevan, J.; Svore, K. M.; and Dumais, S. T. 2011. Understanding temporal query dynamics. In *Proceedings of the fourth ACM international conference on Web search and data mining*, 167–176.

Lazaridou, A.; Gribovskaya, E.; Stokowiec, W.; and Grigorev, N. 2022. Internet-augmented language models through few-shot prompting for open-domain question answering. *arXiv preprint arXiv:2203.05115*.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474.

Miller, T.; Bethard, S.; Dligach, D.; and Savova, G. 2023. End-to-end clinical temporal information extraction with multi-head attention. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2023, 313.

Patterson, D.; Gonzalez, J.; Le, Q.; Liang, C.; Munguia, L.-M.; Rothchild, D.; So, D.; Texier, M.; and Dean, J. 2021. Carbon emissions and large neural network training. *arXiv preprint arXiv:2104.10350*.

Pustejovsky, J.; Castano, J. M.; Ingria, R.; Sauri, R.; Gaizauskas, R. J.; Setzer, A.; Katz, G.; and Radev, D. R. 2003. TimeML: Robust specification of event and temporal expressions in text. *New directions in question answering*, 3: 28–34.

Qian, X.; Zhang, Y.; Zhao, Y.; Zhou, B.; Sui, X.; Zhang, L.; and Song, K. 2024. TimeR4: Time-aware retrieval-augmented large language models for temporal knowledge graph question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 6942–6952.

Robertson, S.; Zaragoza, H.; et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4): 333–389.

Saxena, A.; Chakrabarti, S.; and Talukdar, P. 2021. Question Answering Over Temporal Knowledge Graphs. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 6663–6676.

Schick, T.; and Schütze, H. 2021. It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2339–2352.

Shuster, K.; Poff, S.; Chen, M.; Kiela, D.; and Weston, J. 2021. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*.

Song, K.; Tan, X.; Qin, T.; Lu, J.; and Liu, T.-Y. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in neural information processing systems*, 33: 16857–16867.

Strötgen, J.; Bögel, T.; Zell, J.; Armiti, A.; Van Canh, T.; and Gertz, M. 2014. Extending HeidelTime for Temporal Expressions Referring to Historic Dates. In *LREC*, 2390–2397.

Strötgen, J.; and Gertz, M. 2013. Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47: 269–298.

Thirunavukarasu, A. J.; Ting, D. S. J.; Elangovan, K.; Gutierrez, L.; Tan, T. F.; and Ting, D. S. W. 2023. Large language models in medicine. *Nature medicine*, 29(8): 1930–1940.

Topsakal, O.; and Akinci, T. C. 2023. Creating large language model applications utilizing langchain: A primer on developing llm apps fast. In *International conference on applied engineering and natural sciences*, 1, 1050–1056.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Voorhees, E. M.; Tice, D. M.; et al. 1999. The TREC-8 Question Answering Track Evaluation. In *TREC*, volume 1999, 82.

Vu, T.; Iyyer, M.; Wang, X.; Constant, N.; Wei, J.; Wei, J.; Tar, C.; Sung, Y.-H.; Zhou, D.; Le, Q.; et al. 2023. Freshllms: Refreshing large language models with search engine augmentation. *arXiv preprint arXiv:2310.03214*.

Wang, G.; Cheng, S.; Zhan, X.; Li, X.; Song, S.; and Liu, Y. 2023. Openchat: Advancing open-source language models with mixed-quality data. *arXiv preprint arXiv:2309.11235*.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, 38–45.

Wu, F.; Liu, L.; He, W.; Liu, Z.; Zhang, Z.; Wang, H.; and Wang, M. 2024. Time-Sensitve Retrieval-Augmented Generation for Question Answering. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, CIKM '24, 2544–2553. New York, NY, USA: Association for Computing Machinery. ISBN 9798400704369.

Yu, X.; Chen, Z.; and Lu, Y. 2023. Harnessing LLMs for Temporal Data-A Study on Explainable Financial Time Series Forecasting. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, 739–753.

Zhang, P.; Zeng, G.; Wang, T.; and Lu, W. 2024. Tinyllama: An open-source small language model. *arXiv preprint arXiv:2401.02385*.

Zhao, W. X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2).