RESEARCH-ARTICLE

TempRetriever: Fusion-based Temporal Dense Passage Retrieval for Time-Sensitive Questions

ABDELRAHMAN ABDALLAH, University of Innsbruck, Innsbruck, Tyrol, Austria

BHAWNA PIRYANI, University of Innsbruck, Innsbruck, Tyrol, Austria

JONAS WALLAT, L3S Research Center, Hannover, Niedersachsen, Germany

AVISHEK ANAND, Delft University of Technology, Delft, Zuid-Holland, Netherlands

ADAM JATOWT, University of Innsbruck, Innsbruck, Tyrol, Austria

Open Access Support provided by:

L3S Research Center

University of Innsbruck

Delft University of Technology

Published: 21 February 2026

Citation in BibTeX format

WSDM '26:The Nineteenth ACM International Conference on Web Search and Data Mining
February 22 - 26, 2026
ID, Boise, USA

Conference Sponsors:
SIGKDD
SIGWEB
SIGIR
SIGMOD

TempRetriever: Fusion-based Temporal Dense Passage Retrieval for Time-Sensitive Questions

Abdelrahman Abdallah
University of Innsbruck
Innsbruck, Austria
abdelrahman.abdallah@uibk.ac.at

Bhawna Piryani
University of Innsbruck
Innsbruck, Austria
bhawna.piryani@uibk.ac.at

Jonas Wallat
L3S Research Center
Hannover, Germany
jonas.wallat@l3s.de

Avishek Anand
Delft University of Technology
Delft, The Netherlands
avishek.anand@tudelft.nl

Adam Jatowt
University of Innsbruck
Innsbruck, Austria
adam.jatowt@uibk.ac.at

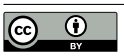
Abstract

Temporal information is crucial for information retrieval, yet most dense retrieval systems focus exclusively on semantic similarity while neglecting temporal alignment between queries and documents. We propose TempRetriever[1], a lightweight framework that explicitly incorporates temporal information into dense passage retrieval through learned fusion techniques. Unlike existing approaches requiring extensive architectural modifications or specialized pre-training, TempRetriever enhances standard dense retrievers by combining semantic embeddings with temporal representations using four fusion strategies: Feature Stacking, Vector Summation, Relative Embeddings, and Element-Wise Interaction. Our approach introduces a learned temporal encoder and time-based negative sampling strategy to address temporal misalignment during training. We evaluate TempRetriever on three temporal question answering datasets (ArchivalQA, ChroniclingAmericaQA, NobelPrize) spanning altogether years from 1800 to 2022. TempRetriever achieves substantial improvements over standard DPR: 6.86% on ArchivalQA (Recall@1) and 4.40% on ChroniclingAmericaQA (Recall@1). Our method also outperforms state-of-the-art temporal retrieval systems, obtaining 9.62% improvement over BiTimeBERT and 5.16% over TS-Retriever. Notably, TempRetriever's fusion techniques can enhance existing temporal methods, improving BiTimeBERT by 5.12% and TS-Retriever by 6.17%, demonstrating modularity and practical value. Zero-shot evaluation confirms strong generalization across domains, and integration with retrieval-augmented generation shows consistent end-to-end improvements.

CCS Concepts

• **Information systems** → **Question answering**; **Information retrieval**.

[1]https://github.com/DataScienceUIBK/TempRetriever



WSDM '26, Boise, ID, USA

ACM ISBN 979-8-4007-2292-9/2026/02
https://doi.org/10.1145/3773966.3777938

Keywords

Temporal Question Answering, Dense Passage Retrieval, Temporal Information Retrieval

ACM Reference Format:
Abdelrahman Abdallah, Bhawna Piryani, Jonas Wallat, Avishek Anand, and Adam Jatowt. 2026. TempRetriever: Fusion-based Temporal Dense Passage Retrieval for Time-Sensitive Questions. In *Proceedings of the Nineteenth ACM International Conference on Web Search and Data Mining (WSDM '26), February 22–26, 2026, Boise, ID, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3773966.3777938

1 Introduction

Temporal information is fundamental to human understanding and plays a critical role in information retrieval, yet most modern dense retrieval systems largely ignore the temporal dimension [5, 11, 12, 17, 29, 36, 51, 41, 38]. When searching for information, users often have implicit or explicit temporal constraints that significantly affect document relevance. Consider the query *"What are the key policies of the current US President?"* vs. *"What were the key policies implemented by the US President in 2022?"* While both queries seek presidential policy information, the first requires documents about the present administration, whereas the second demands documents specifically from 2022. Traditional dense retrieval models like Dense Passage Retrieval (DPR) [19] would treat these queries similarly, potentially returning temporally misaligned results.

The importance of temporal awareness extends across numerous domains where information quality depends heavily on timeliness and temporal context [1, 7, 22, 27, 29, 43, 52]. In news retrieval, legal document search, historical research, and scientific literature review, the *when* of information is often as crucial as the *what*. As Metzger [23] demonstrated, timeliness serves as a fundamental measure of information quality alongside accuracy, relevance, and objectivity. Despite this, most dense retrieval systems focus exclusively on semantic similarity while neglecting temporal alignment between queries and documents.

Current dense passage retrieval methods [19, 31, 39] have achieved remarkable success in open-domain question answering by encoding queries and documents into dense vector representations that capture semantic relationships. However, these approaches typically operate on synchronic document collections like Wikipedia [40, 28], which represent world knowledge at a fixed point in time. In contrast, many real-world applications require retrieval from

diachronic collections—evolving document sets where temporal context significantly impacts relevance. For instance, a query about "recent developments in COVID-19 treatment" should prioritize documents from recent months over older publications, even if the older documents contain semantically similar content. Previous attempts to incorporate temporal understanding into language models have largely focused on architectural modifications [26, 33, 45] or specialized pre-training procedures [32, 47]. While these approaches show promise, they often require significant computational overhead, extensive architectural changes, or domain-specific pre-training that limits their practical applicability. For example, BiTimeBERT [45] necessitates comprehensive pre-training with temporal objectives, while TS-Retriever [47] requires specialized contrastive learning procedures with temporal constraints.

We propose TempRetriever, a novel approach to temporal-aware dense retrieval that explicitly incorporates temporal information through lightweight fusion techniques. Unlike previous methods that require architectural overhauls or extensive pre-training, TempRetriever enhances existing dense retrieval frameworks by embedding both query dates and document timestamps directly into the retrieval process. Our approach enables retrieval of passages that achieve both contextual relevance and temporal alignment without fundamental changes to the underlying model architecture. The core insight behind TempRetriever lies in treating temporal information as a first-class citizen in the retrieval process. We introduce four fusion techniques: Feature Stacking (FS), Vector Summation (VS), Relative Embeddings (RE), and Element-Wise Interaction (EWI) that combine semantic and temporal embeddings in different ways. These techniques allow the model to learn how temporal and semantic signals should be balanced for optimal retrieval performance. Additionally, we propose a novel time-based negative sampling strategy that explicitly addresses temporal misalignment during training by selecting negative examples based on their temporal characteristics relative to positive documents.

Our method consistently achieves higher recall across different k values, demonstrating its effectiveness in ranking temporally relevant documents higher in the result list. This improvement is particularly significant for Recall@1, where TempRetriever achieves a 6.63% improvement over standard DPR, indicating better precision in identifying the most relevant documents.

To demonstrate the generalizability of our approach, we extend TempRetriever to handle implicit temporal questions—queries that lack explicit temporal references but have inherent temporal intent. We develop a query date prediction model that infers temporal context from query content, enabling unified temporal-aware retrieval across both explicit and implicit temporal queries. Furthermore, we integrate TempRetriever into Retrieval-Augmented Generation (RAG) pipelines, showing how improved temporal retrieval translates to better answer generation in downstream applications. Our comprehensive evaluation spans three temporal question answering datasets: ArchivalQA [44], ChroniclingAmericaQA [30], and NobelPrize [47]. These datasets cover different temporal ranges and domains, from historical news (1800-2007) to structured factual content (1902-2022), providing robust evidence of TempRetriever's effectiveness across diverse temporal retrieval scenarios.

Contributions. Our work makes four key contributions to temporal information retrieval:

(1) **Temporal Fusion Framework:** We propose TempRetriever, a novel dense retrieval framework that explicitly incorporates both query dates and document timestamps through four fusion techniques.
(2) **Time-based Negative Sampling:** We introduce a novel negative sampling strategy that leverages temporal characteristics to select training examples.
(3) **Unified Temporal Query Handling:** We demonstrate how TempRetriever can be extended to implicit temporal questions through a query date prediction component, allowing to provide a complete framework for both explicit and implicit temporal retrieval.
(4) **Comprehensive Empirical Analysis:** We provide extensive experimental validation across multiple datasets and metrics, including zero-shot evaluation on the NobelPrize benchmark.

2 Related Work

Dense retrieval methods encode queries and documents into continuous vector spaces that capture semantic relationships beyond lexical overlap [19, 50, 24]. The seminal DPR work [19] demonstrated that BERT-based encoders could outperform traditional sparse methods like BM25. Subsequent improvements focused on better training strategies [31, 39], architectural innovations like ColBERT [20, 35], and enhanced negative sampling [15, 48]. Despite these advances, dense retrieval systems typically treat documents as static entities without considering temporal context. Our work demonstrates how temporal information can be seamlessly integrated into existing dense retrieval frameworks.

Classical TIR emerged as a specialized field incorporating time-related signals into search systems [4, 16]. Early work concentrated on temporal query understanding [23, 27] and document timestamp extraction [6], primarily operating on sparse retrieval models. Recent TIR research explored temporal dynamics in web search [21], news retrieval [18], and social media analysis [53]. However, these approaches have not been extensively adapted to modern dense retrieval architectures [3]. Our work bridges this gap by bringing classical TIR insights into the dense retrieval paradigm.

Recent work has enhanced language models with temporal reasoning capabilities [49, 52, 38, 8, 37, 46, 29]. Cole et al. [9] introduced Temporal Span Masking (TSM) for pre-training temporal reasoning, while Rosin and Radinsky [33] proposed temporal attention mechanisms. Wang et al. [45] developed BiTimeBERT, integrating document timestamps through specialized pre-training tasks including time-aware masked language modeling. While BiTimeBERT achieves strong performance, it requires comprehensive temporal pre-training, making it computationally expensive. Our approach enhances existing pre-trained models without requiring temporal-specific pre-training. Most directly related, Wu et al. [47] proposed TS-Retriever, integrating supervised contrastive learning with temporal constraints by altering time specifiers in queries to generate temporally mismatched negatives. While TS-Retriever shares our temporal-aware retrieval goal, our approach differs significantly: TS-Retriever requires specialized contrastive learning procedures, whereas TempRetriever uses lightweight fusion techniques that can enhance any dense retriever. Additionally, TS-Retriever focuses

on explicit temporal constraints, while we handle implicit temporal queries through date prediction. Our fusion-based approach is more modular—it can enhance existing temporal methods like TS-Retriever itself.

Positioning of Our Work. TempRetriever distinguishes itself from prior work through its emphasis on *lightweight, generalizable temporal enhancement*. Unlike BiTimeBERT, which requires extensive temporal pre-training, or TS-Retriever, which needs specialized contrastive learning, our fusion-based approach can enhance any existing dense retriever with minimal modification. Our temporal fusion techniques are inspired by classical TIR principles but adapted for modern neural architectures. Furthermore, our approach is uniquely modular—we demonstrate that TempRetriever can improve not only standard dense retrievers but also existing temporal methods like BiTimeBERT and TS-Retriever. This modularity, combined with our extension to implicit temporal queries and integration with RAG pipelines, provides a comprehensive framework for temporal-aware information retrieval that addresses limitations across multiple prior approaches. The temporal negative sampling strategy we introduce draws inspiration from hard negative mining in dense retrieval [19, 48] but specifically targets temporal misalignment—a challenge not addressed by previous negative sampling methods. This contribution bridges classical TIR insights about temporal relevance with modern dense retrieval training techniques.

3 Method

We propose TempRetriever, a framework for incorporating temporal information into dense passage retrieval through explicit temporal-semantic fusion. Our approach extends existing dense retrieval models by learning to combine semantic embeddings with temporal representations, enabling retrieval systems to consider both content relevance and temporal alignment.

3.1 Problem Formulation

Let $\mathcal{V}$ represent the vocabulary space encompassing all possible tokens within a corpus. We define $\mathcal{P} = \{p_1, p_2, \ldots, p_M\}$ as a collection of M text passages, where each passage $p \in \mathcal{P}$ is a sequence of tokens drawn from $\mathcal{V}$. Similarly, $\mathcal{Q}$ denotes the set of all possible queries, with each query $q \in \mathcal{Q}$ also being a sequence of tokens from $\mathcal{V}$. We introduce the temporal space $\mathcal{T}$ containing timestamps associated with both queries and passages, where $t_q \in \mathcal{T}$ represents the temporal context of query q, and $t_{p_i} \in \mathcal{T}$ represents the publication date or relevant timestamp of passage p_i.

Traditional dense retrieval optimizes for semantic similarity between queries and passages through learned embeddings. However, temporal relevance introduces an additional dimension where the temporal distance and alignment between t_q and t_{p_i} significantly impacts the true relevance of passage p_i to query q. Our goal is to learn a joint representation that captures both semantic content and temporal context, enabling the retrieval of passages that are semantically relevant and temporally aligned.

Formally, we aim to learn encoding functions that map queries and passages into a shared embedding space $\mathbb{R}^d$ where temporal and semantic similarities are jointly optimized:

$$f_q(q, t_q) : \mathcal{Q} \times \mathcal{T} \to \mathbb{R}^d \tag{1}$$

$$f_p(p_i, t_{p_i}) : \mathcal{P} \times \mathcal{T} \to \mathbb{R}^d \tag{2}$$

The relevance score between a query q and passage p_i is then computed as:

$$\text{sim}(q, p_i) = f_q(q, t_q)^\top f_p(p_i, t_{p_i}) \tag{3}$$

3.2 Temporal Dense Passage Retrieval

Figure 1 illustrates the overall architecture of TempRetriever in comparison to baseline approaches. Our framework consists of three main components: semantic encoders for text content, a temporal encoder for timestamp information, and fusion mechanisms that combine these representations.

Semantic Encoding. We employ a shared BERT-based encoder [10] to generate semantic embeddings for both queries and passages. Specifically, we use the same pre-trained BERT model with parameters θ_{BERT} to encode textual content:

$$\mathbf{s}_q = E_{BERT}(q; \theta_{BERT}) = \text{BERT}(q)_{[\text{CLS}]} \in \mathbb{R}^{d_s} \tag{4}$$

$$\mathbf{s}_p = E_{BERT}(p; \theta_{BERT}) = \text{BERT}(p)_{[\text{CLS}]} \in \mathbb{R}^{d_s} \tag{5}$$

where $d_s = 768$ is the dimension of BERT embeddings, and the [CLS] token serves as the aggregate representation. While we use the notation E_{BERT} for both queries and passages, the same model parameters are shared across both applications.

Temporal Encoding. A key component of TempRetriever is the temporal encoder E_t, which transforms timestamp information into dense vector representations that can be meaningfully combined with semantic embeddings. The temporal encoder architecture is designed to capture temporal relationships and patterns that are relevant for retrieval tasks.

Temporal-Semantic Fusion. We propose four fusion techniques to combine semantic and temporal embeddings, each capturing different aspects of temporal-semantic interaction:

Feature Stacking (FS): Concatenates semantic and temporal embeddings along the feature dimension:

$$\text{Fuse}_{FS}(\mathbf{s}, \mathbf{e}_t) = [\mathbf{s} \oplus \mathbf{e}_t] \in \mathbb{R}^{d_s + d_t} \tag{6}$$

This approach preserves all information from both modalities and allows the model to learn complex interactions through subsequent processing. The resulting embedding dimension is $d = d_s + d_t = 1536$.

Vector Summation (VS): Combines embeddings through element-wise addition:

$$\text{Fuse}_{VS}(\mathbf{s}, \mathbf{e}_t) = \mathbf{s} + \mathbf{e}_t \in \mathbb{R}^{d_s} \tag{7}$$

This technique assumes that semantic and temporal information can be additively combined, maintaining the original embedding dimension while integrating temporal signals.

Relative Embeddings (RE): Captures the relative relationship between semantic and temporal representations:

$$\text{Fuse}_{RE}(\mathbf{s}, \mathbf{e}_t) = \mathbf{s} - \mathbf{e}_t \in \mathbb{R}^{d_s} \tag{8}$$

This approach emphasizes the difference between semantic and temporal signals, potentially useful when temporal deviation from semantic expectations is informative.

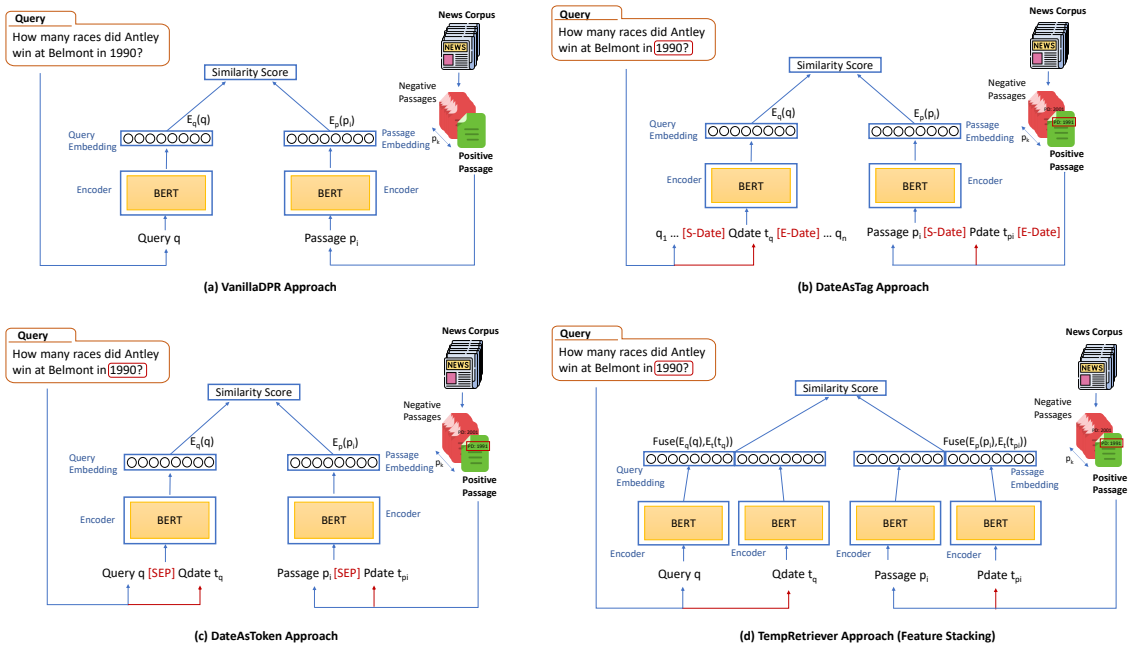


Figure 1: Architecture overview of TempRetriever compared to other approaches. (a) Vanilla DPR processes only textual content. (b-c) DateAsTag and DateAsToken incorporate temporal information as special tokens or plain text, respectively. (d) TempRetriever explicitly fuses semantic and temporal embeddings through learned fusion techniques. The temporal encoder E_t transforms timestamps into dense representations that are combined with BERT embeddings via fusion functions.

Element-Wise Interaction (EWI): Performs element-wise multiplication to capture feature interactions:

$$\text{Fuse}_{EWI}(\mathbf{s}, \mathbf{e}_t) = \mathbf{s} \odot \mathbf{e}_t \in \mathbb{R}^{d_s} \tag{9}$$

This technique models multiplicative interactions between semantic and temporal features, allowing temporal information to modulate semantic representations.

The final query and passage representations are:

$$\mathbf{q}_{final} = \text{Fuse}(E_{BERT}(q), E_t(t_q)) \tag{10}$$

$$\mathbf{p}_{final} = \text{Fuse}(E_{BERT}(p), E_t(t_p)) \tag{11}$$

Training Objective. We train TempRetriever using a contrastive learning objective that encourages temporally and semantically relevant passages to have higher similarity scores than irrelevant ones. For each training instance consisting of a query q_i with timestamp t_{q_i}, a positive passage p_i^+ with timestamp $t_{p_i^+}$, and n negative passages $\{p_{i,j}^-, t_{p_{i,j}^-}\}_{j=1}^n$, the loss is:

$$\mathcal{L} = -\log \frac{\exp(\mathbf{q}_{final}^\top \mathbf{p}_{final}^+ / \tau)}{\exp(\mathbf{q}_{final}^\top \mathbf{p}_{final}^+ / \tau) + \sum_{j=1}^n \exp(\mathbf{q}_{final}^\top \mathbf{p}_{j,final}^- / \tau)} \tag{12}$$

where τ is a temperature parameter. This objective enables the model to learn temporal-semantic representations that distinguish between temporally aligned and misaligned documents.

3.3 Handling Implicit Temporal Questions

While TempRetriever primarily targets explicit temporal questions containing direct date references, many real-world queries have implicit temporal intent that must be inferred. To address this challenge, we extend our framework with a query date prediction component.

Query Date Prediction Model. We develop a BERT-based classification model to predict the most likely temporal context for implicit queries. The model is trained on the Event Sentence dataset [42], which contains 22,399 short event descriptions annotated with corresponding years spanning 1987 to 2007.

The architecture consists of a BERT encoder followed by a classification head:

$$\mathbf{h}_{query} = \text{BERT}(q)_{[\text{CLS}]} \tag{13}$$

$$\mathbf{p}_{year} = \text{softmax}(W_{cls}\mathbf{h}_{query} + b_{cls}) \tag{14}$$

where $W_{cls} \in \mathbb{R}^{21\times768}$ projects to 21 year classes (1987-2007). The predicted year $\hat{t}_q = \arg\max(\mathbf{p}_{year})$ is then used as the temporal context for TempRetriever.

Integration with TempRetriever. For implicit temporal queries, we apply a two-stage process: first, the query date prediction model estimates the temporal context $\hat{t}_q$; second, TempRetriever uses this predicted timestamp along with the query text for temporal-aware retrieval. This enables unified handling of both explicit and implicit temporal queries within the same framework.

3.4 Time-based Negative Sampling

Effective negative sampling is crucial for training high-quality dense retrievers [19, 15]. For temporal retrieval, we propose a time-based negative sampling strategy that explicitly considers temporal characteristics when selecting training examples.

Sampling Strategies. We experiment with three negative sampling approaches:

- *Random Negative:* Randomly select passages that do not contain the correct answer, representing the standard approach used in most dense retrieval systems.
- *Same-Year Negative:* Randomly select negative passages from the same year as the positive passage's timestamp. This strategy helps the model learn to distinguish between documents that are temporally aligned but semantically irrelevant.
- *Different-Year Negative:* Select negative passages from years different from the positive passage's year. This approach helps the model learn temporal discrimination by contrasting temporally misaligned documents.

Table 1: Statistics of ArchivalQA, ChroniclingAmericaQA, and NobelPrize datasets. Note: NobelPrize is used only for zero-shot evaluation (no training/validation splits).

Dataset	Type	Train	Val	Test	Time Range
ArchivalQA	Explicit	62,157	7,841	7,783	1987-2007
ArchivalQA	Implicit	5,000	1,000	1,000	1987-2007
ChroniclingAmericaQA	Explicit	22,918	1,232	1,268	1800-1920
NobelPrize	Explicit	-	-	3,224	1902-2022

4 Experimental Setup

We conduct comprehensive experiments to evaluate TempRetriever across multiple temporal question answering datasets, comparing against both traditional dense retrieval methods and state-of-the-art temporal retrieval systems. This section details our experimental design, datasets, implementation choices, and evaluation methodology.

4.1 Datasets and Preprocessing

We evaluate our approach on three temporal question answering datasets that span different domains, time periods, and temporal reasoning challenges. Table 1 provides a comprehensive overview of dataset statistics and splits.

ArchivalQA [44] is a benchmark dataset designed for open-domain question answering over historical news collections. The dataset comprises question-answer pairs derived from news articles published between 1987 and 2007 from the New York Times Annotated Corpus (NYT corpus) [34].

ChroniclingAmericaQA [30] focuses on question-answering over American historical newspaper collections spanning 1800 to 1920. This dataset provides a longer historical perspective and different linguistic characteristics compared to modern news articles.

NobelPrize [47] serves as our zero-shot evaluation benchmark, comprising 3,224 queries and 989 documents constructed from structured sources like the Nobel Prize dataset from Kaggle. Spanning from 1902 to 2022, this dataset tests the generalisation capabilities of models trained on other temporal datasets without domain-specific fine-tuning.

Corpus Preprocessing. For **ArchivalQA**, we follow the standard preprocessing approach used in DPR [2, 19]. Each NYT article is segmented into disjoint text blocks of 100 words, with each block serving as an independent passage for retrieval. This process yields 19,851,114 passages across the entire corpus. Each passage is prepended with the article's title and separated by a [SEP] token to provide additional context. **ChroniclingAmericaQA** uses a different structure where the corpus is already organized into passages of approximately 250 words each, totaling 151,485 passages. These passages represent coherent segments of historical newspaper content and do not require further segmentation. For **NobelPrize**, we use the passages as provided in the original dataset without additional preprocessing, maintaining consistency with prior work for fair comparison.

4.2 Baseline and State-of-the-Art Methods

We compare TempRetriever against comprehensive baselines representing different approaches to temporal information integration:

Traditional Dense Retrieval Baselines:

- *Vanilla DPR* [19] serves as our primary baseline, using standard BERT-base encoders with in-batch negative sampling for semantic-only retrieval.
- *DateAsTag* (DateAsTag) incorporates temporal information through special tokens, marking dates as `[S-DATE] date [E-DATE]` for both queries and passages.
- *DateAsToken* (DateAsToken) appends timestamps as plain text (`query [SEP] date`), testing whether BERT can capture temporal relationships through natural language understanding.

State-of-the-Art Temporal Retrieval Methods:

- *BiTimeBERT* [45]: A temporal language model that integrates document timestamps and temporal references through specialized pre-training tasks.
- *TS-Retriever* [47]: A time-sensitive retrieval framework that employs supervised contrastive learning with temporal constraints.
- *Contriever* [15]: A state-of-the-art unsupervised dense retrieval model trained with contrastive learning objectives.
- *TAS-B* [14]: An efficient neural retrieval model that balances performance and computational cost.
- *JinaAI Embeddings* [13]: A recent embedding model (jina-embeddings-v2-base-en) designed for semantic search applications.
- *OpenAI Embeddings* [25]: The text-embedding-ada-002 model representing commercial state-of-the-art embedding systems.

4.3 Implementation Details

Model Architecture and Parameters. All methods use BERT-base-uncased as the foundation encoder, ensuring fair comparison across approaches. For TempRetriever, we implement the temporal encoder E_t as described in Section 3.2. The final embedding dimension varies by fusion technique: 1, 536 for Feature Stacking

Table 2: R@{1,5,10,20,50,100} retrieval accuracy for ArchivalQA and ChroniclingAmericaQA datasets, evaluated on both validation and test sets. The results measure the percentage of top retrieved passages that contain the correct answer across different temporal retrieval models, including the baseline VanillaDPR, DateAsTag, DateAsToken, and TempRetriever approaches.

Dataset	Method	Validation Results						Test Results					
		R@1	R@5	R@10	R@20	R@50	R@100	R@1	R@5	R@10	R@20	R@50	R@100
ArchivalQA	VanillaDPR	63.09$_{\pm 0.76}$	82.42$_{\pm 0.29}$	87.47$_{\pm 0.11}$	90.97$_{\pm 0.15}$	94.40$_{\pm 0.21}$	96.14$_{\pm 0.03}$	62.98$_{\pm 0.55}$	82.50$_{\pm 0.33}$	87.17$_{\pm 0.45}$	90.97$_{\pm 0.27}$	94.64$_{\pm 0.23}$	96.15$_{\pm 0.16}$
	DateAsTag	57.94$_{\pm 0.71}$	78.78$_{\pm 0.61}$	84.28$_{\pm 0.41}$	88.48$_{\pm 0.48}$	92.43$_{\pm 0.26}$	94.64$_{\pm 0.13}$	58.37$_{\pm 0.70}$	78.43$_{\pm 0.84}$	83.93$_{\pm 0.45}$	88.21$_{\pm 0.41}$	92.77$_{\pm 0.25}$	94.93$_{\pm 0.10}$
	DateAsToken	63.51$_{\pm 0.32}$	83.41$_{\pm 0.28}$	88.52$_{\pm 0.08}$	91.97$_{\pm 0.08}$	94.92$_{\pm 0.05}$	96.52$_{\pm 0.02}$	63.44$_{\pm 0.29}$	83.23$_{\pm 0.05}$	88.35$_{\pm 0.10}$	91.72$_{\pm 0.03}$	94.94$_{\pm 0.23}$	96.72$_{\pm 0.08}$
	TempRetriever $_{FS}$	**69.72**$_{\pm 0.34}$	**87.82**$_{\pm 0.32}$	**91.38**$_{\pm 0.09}$	**94.21**$_{\pm 0.06}$	**96.44**$_{\pm 0.12}$	**97.77**$_{\pm 0.03}$	**69.84**$_{\pm 0.86}$	**87.78**$_{\pm 0.46}$	**91.65**$_{\pm 0.36}$	**94.28**$_{\pm 0.35}$	**96.78**$_{\pm 0.23}$	**97.75**$_{\pm 0.13}$
ChroniclingAmericaQA	VanillaDPR	44.97$_{\pm 0.24}$	61.14$_{\pm 0.16}$	67.86$_{\pm 0.16}$	73.32$_{\pm 0.21}$	79.12$_{\pm 0.10}$	83.53$_{\pm 0.15}$	45.70$_{\pm 0.46}$	60.47$_{\pm 0.25}$	67.40$_{\pm 0.14}$	72.20$_{\pm 0.18}$	78.69$_{\pm 0.13}$	82.26$_{\pm 0.09}$
	DateAsTag	45.00$_{\pm 0.17}$	61.48$_{\pm 0.31}$	67.62$_{\pm 0.19}$	72.97$_{\pm 0.07}$	79.14$_{\pm 0.18}$	82.99$_{\pm 0.18}$	45.75$_{\pm 0.48}$	60.77$_{\pm 0.49}$	66.77$_{\pm 0.22}$	72.44$_{\pm 0.32}$	78.72$_{\pm 0.18}$	81.89$_{\pm 0.21}$
	DateAsToken	44.81$_{\pm 0.04}$	61.28$_{\pm 0.28}$	67.62$_{\pm 0.16}$	73.60$_{\pm 0.13}$	79.12$_{\pm 0.25}$	83.17$_{\pm 0.07}$	45.81$_{\pm 0.09}$	60.79$_{\pm 0.19}$	66.58$_{\pm 0.20}$	72.09$_{\pm 0.15}$	78.82$_{\pm 0.07}$	82.10$_{\pm 0.22}$
	TempRetriever $_{FS}$	**50.22**$_{\pm 0.19}$	**64.88**$_{\pm 0.23}$	**71.16**$_{\pm 0.30}$	**77.90**$_{\pm 0.21}$	**83.05**$_{\pm 0.11}$	**86.22**$_{\pm 0.29}$	**50.10**$_{\pm 0.10}$	**66.01**$_{\pm 0.17}$	**71.85**$_{\pm 0.39}$	**77.19**$_{\pm 0.11}$	**82.81**$_{\pm 0.25}$	**86.52**$_{\pm 0.10}$

Table 3: nDCG@10 and MAP@10 results on ArchivalQA and ChroniclingAmericaQA datasets across validation and test sets. These ranking quality metrics demonstrate that TempRetriever not only improves recall but also enhances the overall ranking of relevant passages.

Dataset	Method	nDCG@10		MAP@10	
		Val	Test	Val	Test
ArchivalQA	VanillaDPR	61.25	61.25	57.02	57.12
	DateAsTag	54.71	55.47	50.38	51.23
	DateAsToken	60.71	60.76	56.30	56.44
	TempRetriever $_{FS}$	**67.76**	**68.02**	**63.64**	**63.81**
ChroniclingAmericaQA	VanillaDPR	48.35	48.87	45.44	46.29
	DateAsTag	48.24	48.41	45.51	45.98
	DateAsToken	48.35	48.87	45.44	46.29
	TempRetriever $_{FS}$	**53.03**	**52.85**	**50.29**	**50.09**

Table 4: Recall-{1,20,50,100} retrieval accuracy comparing TempRetriever variants against state-of-the-art temporal retrieval models on ArchivalQA and ChroniclingAmericaQA datasets. Results are evaluated on both validation and test sets.

Dataset	Model	Validation				Test			
		R@1	R@20	R@50	R@100	R@1	R@20	R@50	R@100
ChroniclingAmericaQA	BiTimeBert	35.74	66.94	74.90	78.31	38.52	65.59	72.69	76.95
	TS-Retriever	47.85	73.60	79.53	83.43	46.65	74.27	78.77	82.56
	VanillaDPR	44.97	73.32	79.12	83.53	45.70	72.20	78.69	82.26
	TempRetriever $_{VS}$	49.23	75.14	81.80	85.38	49.72	76.72	81.45	84.77
	TempRetriever $_{RE}$	48.90	75.55	81.48	84.40	47.28	75.85	82.24	85.56
	TempRetriever $_{EWI}$	47.44	74.82	80.18	84.48	46.57	75.22	80.66	84.06
	TempRetriever $_{FS}$	**50.22**	**77.90**	**83.05**	**86.22**	**50.10**	**77.19**	**82.81**	**86.52**
ArchivalQA	BiTimeBert	61.71	89.99	93.88	95.52	60.22	90.48	94.09	95.62
	TS-Retriever	63.44	92.53	95.88	97.33	64.68	92.92	95.76	97.29
	VanillaDPR	63.09	90.97	94.40	96.14	62.98	90.97	94.64	96.15
	TempRetriever $_{VS}$	68.42	93.30	95.97	97.30	67.96	93.55	96.24	97.55
	TempRetriever $_{RE}$	66.76	94.02	96.40	97.46	66.11	94.32	96.83	97.88
	TempRetriever $_{EWI}$	**69.97**	93.34	95.71	97.08	69.27	93.05	95.85	97.16
	TempRetriever $_{FS}$	69.72	**94.21**	**96.44**	**97.77**	**69.84**	**94.28**	**96.78**	**97.75**

(concatenation of 768-dim semantic and 768-dim temporal embeddings) and 768 for other fusion techniques (VS, RE, EWI).

It is important to clarify the role of the validation and test sets in our evaluation methodology. The validation set is not used for hyperparameter tuning or model selection in our experiments. Instead, both the validation and test sets serve as independent evaluation partitions to assess model performance. This dual evaluation approach provides a more comprehensive assessment of model robustness and consistency across different data splits. All baseline models (VanillaDPR, DateAsTag, DateAsToken) and comparison methods (BiTimeBERT, TS-Retriever) were trained using identical experimental settings and procedures as TempRetriever. Specifically, all models were fine-tuned on the same training splits of ArchivalQA and ChroniclingAmericaQA datasets using the same hyperparameters.

Hardware and Software Environment. All experiments are conducted on NVIDIA A100 GPUs with 40GB memory. We use PyTorch 1.12 with Transformers library version 4.20 for model implementation. Training times range from 6-8 hours for ChroniclingAmericaQA to 24-30 hours for ArchivalQA, depending on a dataset size.

5 Results

We present here comprehensive results demonstrating the effectiveness of TempRetriever across multiple temporal retrieval scenarios. Our evaluation covers explicit temporal questions, zero-shot generalization, implicit temporal query handling, and integration with retrieval-augmented generation systems.

5.1 Explicit Temporal Questions Results

Table 2 presents the main results comparing TempRetriever against baseline methods on ArchivalQA and ChroniclingAmericaQA datasets. TempRetriever consistently outperforms all baseline approaches across different recall metrics, demonstrating the effectiveness of explicit temporal-semantic fusion.

On ArchivalQA, TempRetriever $_{FS}$ achieves substantial improvements over the Vanilla DPR baseline: 6.86% improvement in Recall@1 (69.84% vs 62.98%) and 1.60% improvement in Recall@100 (97.75% vs 96.15%) on the test set. For ChroniclingAmericaQA, the improvements are even more pronounced: TempRetriever $_{FS}$ achieves 4.40% improvement in Recall@1 (50.10% vs 45.70%) and 4.26% improvement in Recall@100 (86.52% vs 82.26%) compared to Vanilla DPR. Notably, DateAsTag underperforms other methods, particularly on ArchivalQA where it achieves only 58.37% Recall@1 compared to Vanilla DPR's 62.98%. This suggests that simply marking temporal information with special tokens is insufficient and may actually interfere with semantic understanding. DateAsToken performs comparably to Vanilla DPR, indicating that treating timestamps as natural language provides minimal benefit without explicit temporal modeling.

Table 5: Zero-shot evaluation results on NobelPrize dataset. Models trained only on ArchivalQA are evaluated on Nobel Prize queries without domain-specific training. TempRetriever variants demonstrate strong generalization capabilities across different temporal domains.

Model	R@1	R@5	R@10	R@20	R@50	R@100	P@1	P@5	P@10	P@20	P@50	P@100
Contriever	4.72	16.89	28.14	43.78	68.40	85.17	17.69	13.14	11.12	8.90	5.50	3.43
TAS-B	2.62	10.17	18.40	31.02	55.18	75.56	9.49	7.27	6.56	5.56	4.03	2.85
JinaAI	12.77	35.58	50.10	65.97	84.15	94.46	38.47	24.68	18.19	12.38	6.57	3.77
OpenAI	9.50	29.56	40.93	55.33	77.57	92.84	29.93	19.75	15.21	10.60	6.23	3.52
TsContriever	19.92	52.70	69.13	81.52	91.54	94.57	58.63	36.49	25.34	15.72	7.30	3.81
TempRetriever $_{FS}$	21.62	58.29	73.35	84.41	**93.58**	**96.97**	63.26	40.24	27.09	16.27	7.43	**3.89**
TempRetriever $_{RE}$	17.27	47.77	63.62	78.45	91.14	96.08	52.96	33.61	23.36	14.97	7.20	3.84
TempRetriever $_{EWI}$	**22.55**	**60.59**	**76.11**	**86.37**	93.49	95.79	**66.68**	**42.39**	**28.34**	**16.74**	**7.45**	3.85
TempRetriever $_{VS}$	16.96	49.52	65.36	78.61	90.88	95.37	54.13	35.07	24.21	15.16	7.20	3.82

Table 6: Performance comparison on implicit temporal questions from ArchivalQA dataset. Results include nDCG@10 and Recall metrics for validation and test sets, demonstrating TempRetriever's effectiveness even when temporal context must be inferred.

Dataset	Method	nDCG@10		R@k (Validation)		
		Val	Test	R@10	R@50	R@100
ArchivalQA	VanillaDPR	39.14	35.42	73.60	85.50	89.50
	BiTimeBERT	24.66	21.05	59.80	75.50	81.80
	DateAsTag	37.11	36.82	71.90	84.30	88.10
	DateAsToken	38.68	38.56	72.90	85.10	89.00
	TempRetriever $_{FS}$	40.58	40.89	74.10	86.50	90.20
	TempRetriever $_{VS}$	41.40	40.72	74.70	86.40	90.10
	TempRetriever $_{EWI}$	39.10	37.12	72.40	84.80	88.50

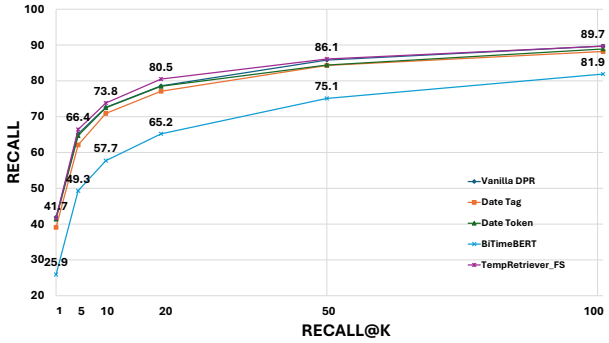


Figure 2: Recall performance comparison for implicit temporal questions on ArchivalQA dataset.

Table 3 presents ranking quality metrics that complement the recall-based evaluation. TempRetriever $_{FS}$ consistently achieves the highest nDCG@10 and MAP@10 scores across both datasets, confirming that temporal-semantic fusion improves not just retrieval coverage but also ranking quality. On ArchivalQA, TempRetriever achieves 68.02% nDCG@10 compared to 61.25% for Vanilla DPR, representing a 6.77% improvement in ranking quality. For ChroniclingAmericaQA, the improvement is 52.85% vs 48.87%, a 3.98% gain.

5.2 Comparison with State-of-the-Art Methods

Table 4 compares TempRetriever against state-of-the-art temporal retrieval methods, including BiTimeBERT and TS-Retriever, demonstrating the superiority of our fusion-based approach. TempRetriever variants consistently outperform both BiTimeBERT and TS-Retriever across all metrics. On ChroniclingAmericaQA, TempRetriever $_{FS}$ achieves 50.10% Recall@1 compared to BiTimeBERT's 38.52% and TS-Retriever's 46.65%, representing improvements of 11.58% and 3.45%, respectively. For ArchivalQA, TempRetriever $_{FS}$ reaches 69.84% Recall@1, outperforming BiTimeBERT (60.22%) by 9.62% and TS-Retriever (64.68%) by 5.16%. The superior performance against BiTimeBERT is particularly noteworthy given that BiTimeBERT requires extensive temporal pre-training, while TempRetriever achieves better results through lightweight fusion techniques applied to standard BERT models.

5.3 Zero-Shot Evaluation on NobelPrize Dataset

Table 5 presents zero-shot evaluation results on the NobelPrize dataset, where models trained exclusively on ArchivalQA are evaluated without any domain-specific fine-tuning. TempRetriever demonstrates remarkable zero-shot generalization capabilities. TempRetriever $_{EWI}$ achieves the highest Recall@1 (22.55%) and Precision@1 (66.68%), significantly outperforming the previous best method TsContriever (19.92% Recall@1, 58.63% Precision@1). TempRetriever $_{FS}$ achieves the highest Recall@100 (96.97%), demonstrating strong performance across different retrieval depths. The strong zero-shot performance is particularly impressive given that the NobelPrize dataset spans a much longer time period (1902-2022) than the training data (1987-2007) and involves structured factual content rather than news articles. This demonstrates that the temporal reasoning capabilities learned by TempRetriever generalize effectively across different domains and temporal ranges.

5.4 Implicit Temporal Question Results

For implicit temporal questions that lack explicit date references, we evaluate the complete pipeline, including query date prediction followed by temporal-aware retrieval. Table 6 and Figure 2 present results on the ArchivalQA implicit question subset.

Even for implicit temporal questions where temporal context must be predicted, TempRetriever variants outperform baseline methods. TempRetriever $_{VS}$ achieves the highest nDCG@10 (41.40%)

Table 7: Comparative case study showing TempRetriever vs Vanilla DPR performance on NobelPrize queries. [C] indicates correct retrieval, [X] indicates incorrect retrieval. Examples demonstrate how temporal alignment helps retrieve appropriate documents while highlighting remaining challenges.

Query	TempRetriever Top-1 Result	Vanilla DPR Top-1 Result
Who was the winner of the Nobel Prize in Medicine 2022 ?	[C] **Correct:** Svante Pääbo. The Nobel Prize in Medicine 2022 . Born: 20, Apr, 1955, Stockholm, Sweden...	[X] **Incorrect:** David Julius. The Nobel Prize in Medicine 2021 . Born: 4, Nov, 1955, New York, USA...
Who was the winner of the Nobel Prize in Chemistry 1993 ?	[C] **Correct:** Kary B. Mullis. The Nobel Prize in Chemistry 1993 . Born: 28, Dec, 1944, Lenoir NC, USA...	[X] **Incorrect:** Linus Pauling. Nobel Prize in Chemistry 1954 . Born: 28, Feb, 1901, Portland, Oregon, USA...

Table 8: Zero-shot Temporal QA results in RAG setting using top-1 retrieved passages. Exact Match (EM), Recall, and Contain metrics evaluate answer generation quality across different language models on ChroniclingAmericaQA and ArchivalQA datasets.

		Validation						Test					
		ChroniclingAmericaQA			ArchivalQA			ChroniclingAmericaQA			ArchivalQA		
	Retriever	EM	Recall	Con	EM	Recall	Con	EM	Recall	Con	EM	Recall	Con
Gemma-2-2B	VanillaDPR	14.13	25.66	16.97	38.10	57.49	44.13	15.78	26.68	18.23	35.78	57.16	44.71
	DateAsTag	13.64	25.38	17.14	35.56	53.58	40.63	15.70	26.73	18.31	33.31	53.52	41.62
	DateAsToken	14.21	25.71	17.38	38.12	57.49	44.20	15.54	26.85	18.15	35.86	56.84	44.31
	TempRetriever	**15.10**	**27.16**	**18.92**	**38.32**	**58.30**	**44.67**	**16.54**	**27.29**	**19.07**	**36.60**	**57.91**	**45.32**
Llama-2-7B	VanillaDPR	14.37	24.65	15.75	38.79	55.48	42.49	15.78	24.85	15.86	36.57	55.40	43.00
	DateAsTag	13.24	23.50	16.57	36.66	52.21	39.71	15.07	24.72	16.65	34.27	51.54	39.68
	DateAsToken	14.05	24.33	16.08	39.07	56.06	42.77	17.04	26.36	17.99	36.77	55.24	42.59
	TempRetriever	**14.45**	**25.98**	**18.03**	**39.94**	**57.38**	**43.71**	**17.73**	**27.48**	**18.91**	**37.51**	**56.43**	**43.56**
Llama-3-8B	VanillaDPR	14.86	26.01	17.30	40.87	57.56	44.52	17.28	27.15	18.31	38.23	57.17	44.59
	DateAsTag	16.08	26.43	18.27	37.25	53.16	40.68	16.17	26.76	18.07	35.28	53.08	41.20
	DateAsToken	16.57	27.33	18.03	40.97	57.57	44.43	17.99	29.19	19.88	38.28	57.00	44.31
	TempRetriever	**16.90**	**28.52**	**19.65**	**41.92**	**58.52**	**45.21**	**18.36**	**30.74**	**20.78**	**39.25**	**58.14**	**45.14**

Table 9: Negative sampling strategies across ArchivalQA and ChroniclingAmericaQA datasets. Results show nDCG@{5,10} and MAP@{5,10} on test sets for different numbers of negative documents and sampling strategies.

Dataset	Strategy	# Neg Docs	nDCG		MAP	
			@5	@10	@5	@10
ArchivalQA	Random	4	60.34	62.20	57.50	58.26
	Same Year	4	63.12	65.15	60.03	60.88
	Diff. Year	4	62.78	64.88	59.61	60.49
	Random	5	61.49	63.53	58.41	59.25
	Same Year	5	**64.79**	**66.71**	**61.75**	**62.55**
	Diff. Year	5	64.44	66.38	61.51	62.31
ChroniclingAmericaQA	Random	4	51.44	52.82	49.50	50.09
	Same Year	4	51.69	52.84	49.48	49.95
	Diff. Year	4	53.52	55.10	51.72	52.37
	Random	5	51.38	53.13	49.66	50.40
	Same Year	5	51.75	53.11	49.99	50.56
	Diff. Year	5	**54.30**	**55.48**	**52.32**	**52.80**

and Recall@10 (74.70%) on the validation set, compared to Vanilla DPR's 39.14% nDCG@10 and 73.60% Recall@10.

The relatively smaller improvements compared to explicit temporal questions are expected, as the query date prediction model can make mistakes. Our query date prediction model achieves a Mean Absolute Error (MAE) of 3.51 years with 20% exact year accuracy, which limits the temporal precision available for retrieval. Despite this challenge, TempRetriever still demonstrates meaningful improvements, suggesting that even approximate temporal context enhances retrieval performance.

5.5 RAG Integration Results

We next integrate TempRetriever into Retrieval-Augmented Generation (RAG) pipelines to evaluate its impact on end-to-end question answering performance. Table 8 presents results using different language models for answer generation. TempRetriever consistently improves answer generation quality across all language models and datasets. With Llama-3-8B, TempRetriever achieves 16.90% Exact Match on ChroniclingAmericaQA validation set compared to 14.86% for Vanilla DPR, representing a 2.04% improvement. On ArchivalQA, the improvement is from 40.87% to 41.92% (1.05% gain).

5.6 Case Study and Error Analysis

Tab. 7 provides a qualitative comparison between TempRetriever and vanilla DPR on Nobel Prize queries. When the query explicitly specifies a year (e.g., the 2022 Nobel Prize in Medicine or the 1993 Nobel Prize in Chemistry), TempRetriever correctly retrieves the temporally aligned laureate (Svante Pääbo and Kary B. Mullis, respectively), whereas vanilla DPR often returns a prominent but temporally mismatched candidate (e.g., David Julius for 2021 or Linus Pauling for 1954). These examples illustrate how temporal alignment helps resolve otherwise plausible but temporally incorrect retrievals by explicitly modeling temporal information.

6 Additional Studies

We conduct additional studies to analyze the contribution of individual components in TempRetriever and validate our design choices.

6.1 Negative Sampling Strategy Analysis

We evaluate three negative sampling strategies to understand how temporal characteristics of training examples affect model performance. Table 9 reveals dataset-specific preferences for negative sampling strategies. For ArchivalQA, the Same-Year strategy performs best, achieving 66.71% nDCG@10 with 5 negative documents compared to 63.53% for Random sampling. This suggests that news retrieval benefits from learning to distinguish between documents from the same time period but with different semantic content. Conversely, ChroniclingAmericaQA achieves optimal performance with the Different-Year strategy (55.48% nDCG@10), indicating that historical newspaper retrieval benefits more from learning temporal discrimination across different time periods. This difference reflects the distinct temporal reasoning requirements of modern news versus historical documents. Figure 3 shows that performance consistently improves as negative examples increase from 4 to 10, with diminishing returns beyond 8 negatives. This suggests that

more temporal contrastive examples help the model learn better temporal discrimination, though computational efficiency limits practical negative sampling to 8-10 examples.

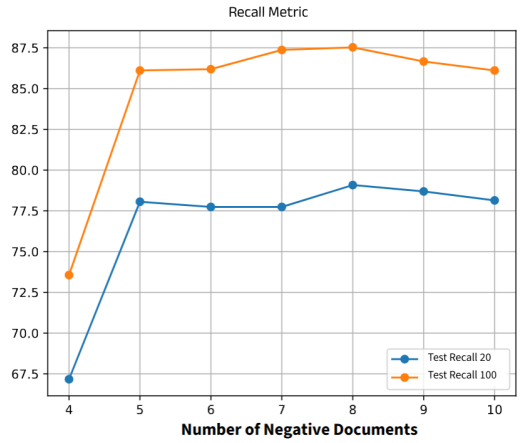


Figure 3: Impact of negative sample quantity on retrieval performance for ChroniclingAmericaQA dataset using Different-Year sampling strategy. Performance generally improves with more negative examples, plateauing around 8-10 negatives.

6.2 Modular Enhancement of Existing Temporal Methods

A key advantage of TempRetriever is its modularity—our fusion techniques can enhance existing temporal retrieval methods. Table 10 demonstrates how TempRetriever fusion improves BiTimeBERT and TS-Retriever performance. TempRetriever fusion improves BiTimeBERT performance across both datasets. On ChroniclingAmericaQA, Feature Stacking with BiTimeBERT achieves 41.75% Recall@1 compared to 38.52% for standalone BiTimeBERT—a 3.23% improvement. The improvements with TS-Retriever are even more substantial. On ChroniclingAmericaQA, TS-Retriever + TempRetriever $_{FS}$ achieves 53.20% Recall@1 versus 46.65% for standalone TS-Retriever—a 6.55% improvement. For ArchivalQA, the combination reaches 70.37% Recall@1 compared to 64.68% for TS-Retriever alone (5.69% improvement).

6.3 Computational Overhead Analysis

We conclude experiments by the computational cost analysis of TempRetriever compared to baseline methods across different fusion techniques as shown in Tab. 11: TempRetriever variants impose minimal computational overhead compared to existing temporal methods. Feature Stacking has the highest overhead (1.34× training time, 2.0× memory) due to doubled embedding dimensions, while Vector Summation and Element-Wise Interaction add only 12-15% training overhead with no memory increase. All TempRetriever variants are significantly more efficient than BiTimeBERT (3.2× training overhead) and TS-Retriever (2.1× training overhead).

Table 10: Enhanced performance of existing temporal methods (BiTimeBERT and TS-Retriever) when combined with TempRetriever fusion techniques. Results show R@{1,20,50,100} on ChroniclingAmericaQA and ArchivalQA test sets.

Dataset	Model	R@1	R@20	R@50	R@100
ChroniclingAmericaQA	BiTimeBERT	38.52	65.59	72.69	76.95
	+ TempRetriever $_{VS}$	38.67	70.80	77.27	**82.79**
	+ TempRetriever $_{EWI}$	40.15	70.51	77.37	82.56
	+ TempRetriever $_{FS}$	**41.75**	70.56	**78.69**	81.77
	TS-Retriever	46.65	74.27	78.77	82.56
	+ TempRetriever $_{VS}$	50.99	**77.43**	**82.08**	84.93
	+ TempRetriever $_{RE}$	52.80	77.17	82.70	**87.00**
	+ TempRetriever $_{FS}$	**53.20**	**77.43**	82.00	86.03
ArchivalQA	BiTimeBERT	60.22	90.48	94.09	95.62
	+ TempRetriever$_{VS}$	63.88	91.96	95.00	96.49
	+ TempRetriever$_{EWI}$	60.52	90.76	94.30	95.90
	+ TempRetriever$_{FS}$	**65.66**	**92.55**	**95.49**	**96.81**
	TS-Retriever	64.68	92.92	95.76	97.29
	+ TempRetriever $_{VS}$	69.88	95.31	97.62	98.50
	+ TempRetriever $_{EWI}$	**70.96**	94.94	97.08	98.18
	+ TempRetriever $_{FS}$	70.37	**95.41**	**97.52**	**98.45**

Table 11: Computational overhead analysis showing training time, inference speed, and memory usage on ArchivalQA test dataset.

Method	Training Time (×)	Inference Speed (×)	Memory Usage (×)	R@1 Performance
Vanilla DPR	1.0×	1.0×	1.0×	62.98
TempRetriever $_{VS}$	1.12×	1.05×	1.0×	67.96
TempRetriever $_{EWI}$	1.15×	1.08×	1.0×	69.27
TempRetriever $_{FS}$	1.34×	1.25×	2.0×	69.84
BiTimeBERT	3.2×	1.8×	1.5×	60.22
TS-Retriever	2.1×	1.4×	1.2×	64.68

7 Conclusion

We introduced TempRetriever, a lightweight framework that explicitly incorporates temporal information into dense passage retrieval through fusion techniques. Unlike existing approaches requiring extensive architectural modifications or specialized pre-training, TempRetriever enhances standard dense retrievers by combining semantic embeddings with temporal representations using four fusion strategies. Our comprehensive evaluation on three temporal QA datasets demonstrates substantial improvements: 6.86% on ArchivalQA and 4.40% on ChroniclingAmericaQA for Recall@1, while outperforming state-of-the-art temporal methods like BiTimeBERT and TS-Retriever. The modularity of our approach enables enhancement of existing temporal retrieval systems, improving the performance of BiTimeBERT by 5.12% and of TS-Retriever by 6.17%. Strong zero-shot generalization on NobelPrize dataset and consistent improvements in RAG pipelines demonstrate practical value for real-world temporal information retrieval applications.

Acknowledgments

The computational results presented here have been achieved (in part) using the LEO HPC infrastructure of the University of Innsbruck and the EuroHPC Joint Undertaking by awarding this project the access to the EuroHPC supercomputer LEONARDO, hosted by CINECA (Italy) and the LEONARDO consortium through EuroHPC.

References

[1] Abdelrahman Abdallah, Bhawna Piryani, and Adam Jatowt. 2023. Exploring the state of the art in legal qa systems. *Journal of Big Data*, 10, 1, 127.

[2] Abdelrahman Abdallah, Bhawna Piryani, Jamshid Mozafari, Mohammed Ali, and Adam Jatowt. 2025. Rankify: a comprehensive python toolkit for retrieval, re-ranking, and retrieval-augmented generation. *arXiv preprint arXiv:2502.02464*.

[3] 2024. *Temporal extraction and retrieval. Information Retrieval: Advanced Topics and Techniques*. (1st ed.). Association for Computing Machinery, New York, NY, USA, 359–387. ISBN: 9798400710506. https://doi.org/10.1145/3674127.3674137.

[4] Ricardo Campos, Gaël Dias, Alípio M Jorge, and Adam Jatowt. 2014. Survey of temporal information retrieval and related applications. *ACM Computing Surveys (CSUR)*, 47, 2, 1–41.

[5] Jiawei Cao, Jie Ouyang, Zhaomeng Zhou, Mingyue Cheng, Yupeng Li, Jiaxian Yan, and Qi Liu. 2025. Re3: learning to balance relevance & recency for temporal information retrieval. *arXiv preprint arXiv:2509.01306*.

[6] Nathanael Chambers, Taylor Cassidy, Bill McDowell, and Steven Bethard. 2014. Dense event ordering with a multi-pass architecture. *Transactions of the Association for Computational Linguistics*, 2, 273–284.

[7] Wenhu Chen, Xinyi Wang, and William Yang Wang. 2021. A dataset for answering time-sensitive questions. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

[8] Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Haotian Wang, Ming Liu, and Bing Qin. 2024. TimeBench: a comprehensive evaluation of temporal reasoning abilities in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Lun-Wei Ku, Andre Martins, and Vivek Srikumar, (Eds.) Association for Computational Linguistics, Bangkok, Thailand, (Aug. 2024), 1204–1228. doi:10.18653/v1/2024.acl-long.66.

[9] Jeremy R. Cole, Aditi Chaudhary, Bhuwan Dhingra, and Partha Talukdar. 2023. Salient span masking for temporal understanding. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*. Andreas Vlachos and Isabelle Augenstein, (Eds.) Association for Computational Linguistics, Dubrovnik, Croatia, (May 2023), 3052–3060. doi:10.18653/v1/2023.eacl-main.222.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*. Jill Burstein, Christy Doran, and Thamar Solorio, (Eds.) Association for Computational Linguistics, Minneapolis, Minnesota, (June 2019), 4171–4186. doi:10.18653/v1/N19-1423.

[11] Bhuwan Dhingra, Jeremy R Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W Cohen. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10, 257–273.

[12] Alexandru Dumitru, Venktesh V, Adam Jatowt, and Avishek Anand. 2025. Evaluating list construction and temporal understanding capabilities of large language models. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)*, 369–379.

[13] Michael Günther et al. 2023. Jina embeddings 2: 8192-token general-purpose text embeddings for long documents. *arXiv preprint arXiv:2310.19923*.

[14] Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently teaching an effective dense retriever with balanced topic aware sampling. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, 113–122.

[15] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

[16] Adam Jatowt, Yukiko Kawai, and Katsumi Tanaka. 2005. Temporal ranking of search engine results. In *Web Information Systems Engineering–WISE 2005: 6th International Conference on Web Information Systems Engineering, New York, NY, USA, November 20-22, 2005. Proceedings 6*. Springer, 43–52.

[17] Zhen Jia, Soumajit Pramanik, Rishiraj Saha Roy, and Gerhard Weikum. 2021. Complex temporal question answering on knowledge graphs. In *Proceedings of the 30th ACM international conference on information & knowledge management*, 792–802.

[18] Nattiya Kanhabua. 2009. Exploiting temporal information in retrieval of archived documents. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, 848–848.

[19] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, (Eds.) Association for Computational Linguistics, Online, (Nov. 2020), 6769–6781. doi:10.18653/v1/2020.emnlp-main.550.

[20] Omar Khattab and Matei Zaharia. 2020. Colbert: efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, 39–48.

[21] Krishna Kulkarni and Jan-Eike Michels. 2012. Temporal features in sql: 2011. *ACM Sigmod Record*, 41, 3, 34–43.

[22] Costas Mavromatis, Prasanna Lakkur Subramanyam, Vassilis N. Ioannidis, Adesoji Adeshina, Phillip R Howard, Tetiana Grinberg, Nagib Hakim, and George Karypis. 2022. Tempoqr: temporal question reasoning over knowledge graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36, 5, (June 2022), 5825–5833. doi:10.1609/aaai.v36i5.20526.

[23] Miriam J Metzger. 2007. Making sense of credibility on the web: models for evaluating online information and recommendations for future research. *Journal of the American society for information science and technology*, 58, 13, 2078–2091.

[24] Sewon Min, Danqi Chen, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2019. A discrete hard EM approach for weakly supervised question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, (Eds.) Association for Computational Linguistics, Hong Kong, China, (Nov. 2019), 2851–2864. doi:10.18653/v1/D19-1284.

[25] Arvind Neelakantan et al. 2022. Text and code embeddings by contrastive pre-training. *arXiv preprint arXiv:2201.10005*.

[26] Yein Park, Chanwoong Yoon, Jungwoo Park, Minbyul Jeong, and Jaewoo Kang. 2025. Does time have its place? temporal heads: where language models recall time-specific information. *arXiv preprint arXiv:2502.14258*.

[27] Marius Pasca. 2008. Towards temporal web search. In *Proceedings of the 2008 ACM symposium on Applied computing*, 1117–1121.

[28] Bhawna Piryani, Abdelrahman Abdallah, Jamshid Mozafari, Avishek Anand, and Adam Jatowt. 2025. It's high time: a survey of temporal information retrieval and question answering. *arXiv preprint arXiv:2505.20243*.

[29] Bhawna Piryani, Abdelrahman Abdallah, Jamshid Mozafari, Avishek Anand, and Adam Jatowt. 2025. It's high time: a survey of temporal question answering. (2025). https://arxiv.org/abs/2505.20243 arXiv: 2505.20243 [cs.CL].

[30] Bhawna Piryani, Jamshid Mozafari, and Adam Jatowt. 2024. Chroniclingamericaqa: a large-scale question answering dataset based on historical american newspaper pages. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (SIGIR '24). Association for Computing Machinery, Washington DC, USA, 2038–2048. ISBN: 9798400704314. doi:10.1145/3626772.3657891.

[31] Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, QiaoQiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021. RocketQAv2: a joint training method for dense passage retrieval and passage re-ranking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, (Eds.) Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, (Nov. 2021), 2825–2835. doi:10.18653/v1/2021.emnlp-main.224.

[32] Guy D. Rosin, Ido Guy, and Kira Radinsky. 2022. Time masking for temporal language models. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining* (WSDM '22). Association for Computing Machinery, Virtual Event, AZ, USA, 833–841. ISBN: 9781450391320. doi:10.1145/3488560.3498529.

[33] Guy D. Rosin and Kira Radinsky. 2022. Temporal attention for language models. In *Findings of the Association for Computational Linguistics: NAACL 2022*. Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, (Eds.) Association for Computational Linguistics, Seattle, United States, (July 2022), 1498–1508. doi:10.18653/v1/2022.findings-naacl.112.

[34] Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6, 12, e26752.

[35] Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. ColBERTv2: effective and efficient retrieval via lightweight late interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, (Eds.) Association for Computational Linguistics, Seattle, United States, (July 2022), 3715–3734. doi:10.18653/v1/2022.naacl-main.272.

[36] Apoorv Saxena, Soumen Chakrabarti, and Partha Talukdar. 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)*. Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, (Eds.) Association for Computational Linguistics, Online, (Aug. 2021), 6663–6676. doi:10.18653/v1/2021.acl-long.520.

[37] Zhaochen Su, Jun Zhang, Tong Zhu, Xiaoye Qu, Juntao Li, Min Zhang, and Yu Cheng. 2024. Timo: towards better temporal reasoning for language models. *arXiv preprint arXiv:2406.14192*.

[38] Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2023. Towards benchmarking and improving the temporal reasoning capability of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Anna Rogers, Jordan Boyd-Graber, and

Naoaki Okazaki, (Eds.) Association for Computational Linguistics, Toronto, Canada, (July 2023), 14820–14835. doi:10.18653/v1/2023.acl-long.828.

[39] Zhengyang Tang, Benyou Wang, and Ting Yao. 2022. DPTDR: deep prompt tuning for dense passage retrieval. In *Proceedings of the 29th International Conference on Computational Linguistics*. Nicoletta Calzolari et al., (Eds.) International Committee on Computational Linguistics, Gyeongju, Republic of Korea, (Oct. 2022), 1193–1202. https://aclanthology.org/2022.coling-1.103/.

[40] Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57, 10, (Sept. 2014), 78–85. doi:10.1145/2629489.

[41] Jonas Wallat, Abdelrahman Abdallah, Adam Jatowt, and Avishek Anand. 2025. A study into investigating temporal robustness of llms. *arXiv preprint arXiv:2503.17073*.

[42] Jonas Wallat, Adam Jatowt, and Avishek Anand. 2024. Temporal blind spots in large language models. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, 683–692.

[43] Jiexin Wang, Adam Jatowt, Michael Färber, and Masatoshi Yoshikawa. 2020. Answering event-related questions over long-term news article archives. In *Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part I 42*. Springer, 774–789.

[44] Jiexin Wang, Adam Jatowt, and Masatoshi Yoshikawa. 2022. Archivalqa: a large-scale benchmark dataset for open-domain question answering over historical news collections. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (SIGIR '22). Association for Computing Machinery, Madrid, Spain, 3025–3035. ISBN: 9781450387323. doi:10.1145/3477495.3531734.

[45] Jiexin Wang, Adam Jatowt, Masatoshi Yoshikawa, and Yi Cai. 2023. Bitimebert: extending pre-trained language representations with bi-temporal information. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (SIGIR '23). Association for Computing Machinery, Taipei, Taiwan, 812–821. ISBN: 9781450394086. doi:10.1145/3539618.3591686.

[46] Yifan Wei, Yisong Su, Huanhuan Ma, Xiaoyan Yu, Fangyu Lei, Yuanzhe Zhang, Jun Zhao, and Kang Liu. 2023. MenatQA: a new dataset for testing the temporal comprehension and reasoning abilities of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*. Houda Bouamor, Juan Pino, and Kalika Bali, (Eds.) Association for Computational Linguistics, Singapore, (Dec. 2023), 1434–1447. doi:10.18653/v1/2023.findings-emnlp.100.

[47] Feifan Wu, Lingyuan Liu, Wentao He, Ziqi Liu, Zhiqiang Zhang, Haofen Wang, and Meng Wang. 2024. Time-sensitve retrieval-augmented generation for question answering. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management* (CIKM '24). Association for Computing Machinery, Boise, ID, USA, 2544–2553. ISBN: 9798400704369. doi:10.1145/3627673.3679800.

[48] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv preprint arXiv:2007.00808*.

[49] Siheng Xiong, Ali Payani, Ramana Kompella, and Faramarz Fekri. 2024. Large language models can learn temporal reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Lun-Wei Ku, Andre Martins, and Vivek Srikumar, (Eds.) Association for Computational Linguistics, Bangkok, Thailand, (Aug. 2024), 10452–10470. doi:10.18653/v1/2024.acl-long.563.

[50] Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-end open-domain question answering with BERTserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*. Waleed Ammar, Annie Louis, and Nasrin Mostafazadeh, (Eds.) Association for Computational Linguistics, Minneapolis, Minnesota, (June 2019), 72–77. doi:10.18653/v1/N19-4013.

[51] Chenhan Yuan, Qianqian Xie, and Sophia Ananiadou. 2023. Zero-shot temporal relation extraction with ChatGPT. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*. Dina Demner-fushman, Sophia Ananiadou, and Kevin Cohen, (Eds.) Association for Computational Linguistics, Toronto, Canada, (July 2023), 92–102. doi:10.18653/v1/2023.bionlp-1.7.

[52] Chenhan Yuan, Qianqian Xie, Jimin Huang, and Sophia Ananiadou. 2024. Back to the future: towards explainable temporal reasoning with large language models. In *Proceedings of the ACM Web Conference 2024* (WWW '24). Association for Computing Machinery, Singapore, Singapore, 1963–1974. ISBN: 9798400701719. doi:10.1145/3589334.3645376.

[53] Fuwei Zhang, Zhao Zhang, Xiang Ao, Fuzhen Zhuang, Yongjun Xu, and Qing He. 2022. Along the time: timeline-traced embedding for temporal knowledge graph completion. In *Proceedings of the 31st ACM international conference on information & knowledge management*, 2529–2538.