Temporal-Aware Neural Retrieval for Personal Knowledge Bases: A Comparative Study of Time-Sensitive Search Methods

Phantastic AI Research
Technical Report T117-v3
January 2026

***Abstract*—Personal knowledge management systems accumulate temporally-structured documents—daily logs, meeting notes, project updates—where retrieval queries frequently carry implicit or explicit temporal intent. Standard neural retrieval using cosine similarity over dense embeddings ignores this temporal dimension, yielding poor results for time-sensitive queries. We evaluate four approaches to temporal-aware retrieval using UForm 3.1.2 neural embeddings (256-d, CPU-only ONNX inference) over a 700-document corpus spanning 6 months: (1) baseline cosine similarity, (2) exponential time-decay weighting, (3) temporal feature injection into the embedding space, and (4) hybrid intent-detection with date-range pre-filtering. Our experiments with 50 queries (19 temporal, 31 neutral) show the hybrid pre-filter achieves the best overall nDCG@10 of 0.4516 (+46.6% over baseline), with temporal query nDCG@10 of 0.4662 versus 0.0883 for baseline—a 5.3× improvement—while preserving identical neutral query performance.**

I. Introduction

Personal knowledge bases (PKBs) such as daily memory logs, project wikis, and meeting notes are inherently temporal artifacts. When users query "what did I work on last week?" or "recent infrastructure changes," the temporal dimension is as important as semantic relevance. However, standard dense retrieval methods treat all documents equally regardless of their creation date.

We investigate four approaches to incorporating temporal signals into neural retrieval for PKBs, using lightweight CPU-only models suitable for deployment on modest VPS infrastructure without GPU acceleration.

II. Methods

A. Embedding Model

We use UForm 3.1.2 [1] with the `uform3-image-text-english-small` model, producing 256-dimensional L2-normalized embeddings via ONNX runtime on CPU. Embedding the full 700-document corpus takes 28.3 seconds.

B. Corpus

Our corpus comprises 700 documents: 41 real paragraphs extracted from dated memory files (January 22–28, 2026) and 659 synthetic documents spanning topics (infrastructure, AI research, project management, personal notes, Mattermost operations, Denario research) distributed uniformly over 180 days.

C. Query Set

We construct 50 queries with manually-defined graded relevance judgments (scores 0–3):

- **19 temporal queries**: "What happened last week?", "Recent AI experiments", "November 2025 activities", etc.
- **31 neutral queries**: "How to configure nginx reverse proxy", "Embedding models comparison", "Meditation practice", etc.

D. Retrieval Methods

Baseline (Cosine): $s(q, d) = \cos(\mathbf{e}_q, \mathbf{e}_d)$

Time-Decay: $s(q, d) = \cos(\mathbf{e}_q, \mathbf{e}_d) \cdot \exp(-\lambda \cdot \text{age}_d)$ where age_d is document age in days. We sweep $\lambda \in \{0.001, 0.005, 0.01, 0.02, 0.05, 0.1\}$.

Temporal Injection: Append a weighted normalized timestamp to each embedding: $\mathbf{e}'_d = \text{norm}([\mathbf{e}_d; w \cdot t_d])$, $\mathbf{e}'_q = \text{norm}([\mathbf{e}_q; w \cdot 1.0])$. We sweep $w \in \{0.05, 0.1, 0.2, 0.5\}$.

Hybrid Pre-filter: Detect temporal intent via keyword matching ("recently" → 30-day window, "last week" → 14-day window, month names → calendar month filter), restrict candidate set, then rank by cosine within the filtered set. Non-temporal queries fall through to pure cosine.

E. Metrics

nDCG@10, Recall@10, and MRR, computed separately for temporal and neutral query subsets.

III. Results

IV. Analysis

A. Hybrid Pre-filter Dominates

The hybrid pre-filter achieves a 46.6% improvement in overall nDCG@10 over the baseline. Critically, it improves temporal query nDCG by 5.3× (0.0883 → 0.4662) while *preserving identical neutral query performance* (0.4427 in both cases). This is because non-temporal queries fall through to unmodified cosine ranking.

TABLE I
OVERALL RETRIEVAL PERFORMANCE (TOP METHODS)

Method	nDCG@10	Recall@10	MRR
Hybrid pre-filter	**0.4516**	**0.4522**	**0.6503**
Baseline cosine	0.3080	0.3002	0.4952
Temporal inj. w=0.1	0.3055	0.2984	0.4967
Time-decay λ=0.001	0.2653	0.2487	0.5423
Time-decay λ=0.01	0.1951	0.1902	0.3359

TABLE II
PERFORMANCE SPLIT BY QUERY TYPE

	Temporal (n=19)		Neutral (n=31)	
Method	nDCG	Recall	nDCG	Recall
Hybrid pre-filter	**0.4662**	**0.5316**	**0.4427**	0.4036
Baseline cosine	0.0883	0.1316	0.4427	0.4036
Time-decay λ=0.01	0.3707	0.3895	0.0875	0.0681
Temporal inj. w=0.5	0.1749	0.2105	0.3517	0.3033

B. Time-Decay: Temporal Gain at Neutral Cost

Time-decay scoring dramatically improves temporal queries (λ=0.01: 0.0883 $\rightarrow$ 0.3707, +4.2$\times$) but catastrophically degrades neutral queries (0.4427 $\rightarrow$ 0.0875, $-$80.2%). The decay unconditionally penalizes older documents regardless of query intent, making it unsuitable as a general-purpose approach.

C. Temporal Injection: Minimal Effect

Appending timestamp features to embeddings produces negligible improvement. At low weights (w=0.05, 0.1), the temporal signal is too weak to influence ranking. At higher weights (w=0.5), it provides modest temporal improvement (0.1749) while degrading neutral performance. The single-dimension timestamp is overwhelmed by the 256-dimensional semantic embedding.

D. Practical Implications

For PKB retrieval systems, we recommend:

1) **Intent detection first**: Classify whether a query has temporal intent before modifying the retrieval strategy.
2) **Pre-filtering over re-ranking**: Date-range filtering before neural scoring is simpler and more effective than multiplicative score adjustments.
3) **Preserve neutral path**: Non-temporal queries should use unmodified cosine similarity—any temporal bias degrades these queries significantly.

V. CONCLUSION

Temporal-aware retrieval for personal knowledge bases is best served by a hybrid approach: detect temporal intent, apply date-range pre-filtering for temporal queries, and preserve standard cosine ranking for non-temporal queries. This achieves a 5.3$\times$ improvement on temporal queries with zero degradation on neutral queries, using only CPU-based 256-d UForm embeddings. Future work includes learning temporal intent classifiers and exploring adaptive decay rates conditioned on detected temporal granularity.

REFERENCES

[1] UForm: Multi-Modal AI Library, Unum Cloud, 2024. https://github.com/unum-cloud/uform