S^2-MEMORY: A ZERO-TRAINING SPARSE CONCEPT KEY-VALUE MEMORY

Anonymous authors
Paper under double-blind review

ABSTRACT

All AI and cognitive systems rest on a shared and rarely questioned assumption: memory requires storage and training. From Transformer KV caches to the classic "memory trace" theory in cognitive science, this assumption has shaped our understanding of memory.

We have identified a simple quantity that governs the retrieval behavior of sparse concept representations. We call it the **truncation loss** Z: the difference in retrieval quality between a concept activation that is fully retained and one truncated to its top-k entries. For the system we study, Z is large and decisive—retaining the full activation raises MRR from 0.47 to 0.91 ($Z = 0.44$)—which identifies truncation, not the choice of representation, as the dominant failure mode.

To examine this, we built **S^2-Memory**—a 126MB, zero-training, plug-and-play system that represents each sentence as a sparse key over a frozen semantic center space. It retrieves over **156,660 sentences from 54 novels of widely varying genre and style** using **half the storage of an fp32 dense index** (120.3 MB vs. 240.6 MB). Integrated with the 1.5B Qwen model (Qwen Team, 2024), it answers cross-book questions without knowing which book contains the answer. Training cost: \$0. Hardware cost: <\$1,000.

Examining the system revealed three further results. First, the self-referential state that motivated its design—a decayed accumulation of concept activations used directly as the retrieval query—is not merely unhelpful but harmful, and should be removed (MRR 0.0014 against a random baseline of 0.0021). Second, the concept centers are highly collinear (effective rank 292.7 of 2000; mean pairwise cosine 0.199); ZCA whitening repairs this and raises R@5 from 0.830 to 0.920. Third, the abstract centers can be replaced by **content-word centers** that preserve retrieval quality (R@5 0.925 vs. 0.930) while making every center nameable: 99.6% of activated centers are semantically related to the sentence, and masking the content-word centers collapses retrieval quality (MRR 0.791 $\rightarrow$ 0.234) whereas masking function-word centers costs almost nothing (MRR 0.791 $\rightarrow$ 0.788).

On our corpus, BM25 is strongest under lexically overlapping queries (MRR 0.9942) but degrades sharply under non-overlapping queries (MRR 0.2650, median rank $1 \rightarrow 21$), where dense retrieval leads (MRR 0.7240). Our sparse key does not exceed dense retrieval in either condition. Because the intermediate representation is a set of natural-language concept labels rather than a dense vector, it can enter a language model's context directly, without a trained probe or projection layer, and it is not bound to any particular encoder; we do not evaluate the downstream benefit of this interface.

1 INTRODUCTION

All AI and cognitive systems rest on a shared and rarely questioned assumption: memory requires storage and training. From Transformer KV caches to the classic "memory trace" theory in cognitive science, this assumption has shaped our understanding of memory.

In this paper we build a system that seeks to weaken part of this assumption—not by abolishing storage, but by making the stored representation small, training-free, and fully readable—and we report honestly what it achieves and what it does not.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

S²-Memory represents each sentence as a *sparse concept key*: a weighted combination of a small number of interpretable concept centers, rather than a raw dense embedding. The system is 126MB, requires no training, and runs on consumer hardware. This paper is an empirical characterization of that representation: we ask when it matches dense retrieval, when it fails, and why.

Our contributions are three, in order of decreasing confidence.

(1) A negative result: the self-referential state is harmful. The original design maintained a state vector $s_t = \gamma s_{t-1} + \eta w_t$ and used it directly as the retrieval query. We show that this state performs *worse than random* (MRR 0.0014 against a random baseline of 0.0021) and explain the mechanism: with $\gamma = 0.999$ over long sequences, the contribution of any early sentence decays to zero, and per-step normalization collapses the remaining history into a generic topic bias rather than specific content. We remove it.

(2) A mechanism finding: truncation, not representation, governs retrieval quality. We show that the dominant failure mode is not the choice of sparse versus dense representation, but the truncation of the concept activation. Query and target share only 1.25 of their top-8 active centers, so a top-8 truncation discards precisely the centers that carry the match. Retaining the full activation raises R@5 from 0.50 to 0.86 and MRR from 0.47 to 0.92, while top-1024 reaches MRR 0.83. We further identify a second, independent cause—the concept centers are highly collinear (effective rank 292.7 of 2000; mean pairwise cosine 0.199)—and show that ZCA whitening repairs it (R@5 0.830 $\rightarrow$ 0.920).

(3) An engineering result with an interpretability trade-off. Once truncation is removed, sparse and dense retrieval become nearly indistinguishable. The abstract centers can be replaced by **content-word centers** at a cost of only 0.5 points of R@5, making every center nameable: masking the content-word centers collapses retrieval (MRR $0.791 \rightarrow 0.234$) whereas masking function-word centers costs almost nothing.

What we do not claim. We do not claim that sparse concept retrieval outperforms dense retrieval; on an equal storage budget, quantized dense vectors remain stronger. We do not claim to outperform BM25, which is stronger than every configuration we tested on our corpus. And we make no claim about replacing long-context attention or about cognitive models of memory.

2 Related Work

Sparse and dense retrieval. Dense passage retrieval matches queries and passages in a learned embedding space (Karpukhin et al., 2020); lexical methods such as BM25 remain competitive on many benchmarks (Robertson & Zaragoza, 2009). Our work does not propose a new retrieval objective. It studies what happens when the intermediate representation is made *readable*—a set of natural-language labels rather than a dense vector—and reports the retrieval cost of doing so.

Interpretable retrieval. A parallel line seeks to make dense retrieval interpretable. IMRNNs (Saxena et al., 2026) augments a frozen dense retriever with runtime embedding modulation, exposing interpretable semantic dependencies between query and document while improving retrieval (+6.35% nDCG across seven benchmarks). This interpretability is *post-hoc*: it is produced by trained adapters applied at inference time. Our approach differs in kind—the intermediate representation is itself a set of natural-language labels, so readability is a property of the representation rather than an attribution computed over it, and no adapter or training is required. The trade-off is explicit in our results: we do not improve over dense retrieval.

Query–document style gaps. HyPE (Vake et al., 2025) moves hypothetical-content generation from query time to indexing time, transforming retrieval into question–question matching and reporting up to 42-point gains in context precision. Like HyPE we operate at indexing time; unlike it, we invoke no LLM during or after indexing, and we claim no retrieval gain.

Evaluation sensitivity. Our results also bear on how retrieval methods are compared. Under lexically overlapping queries BM25 far outperforms dense retrieval on our corpus; under non-overlapping queries the ordering reverses (Section 4.5). We therefore report both conditions and caution against drawing method rankings from a single query construction.

3 SYSTEM

3.1 OVERVIEW

S^2-Memory is a retrieval memory module with the following properties:

- **Size**: 126MB, static.
- **Training**: none. The encoder and concept centers are frozen.
- **Deployment**: plug-and-play; runs on a consumer laptop (16GB RAM, RTX 5060).
- **Cost**: $0 training; hardware under $1,000.

The evaluation corpus consists of five public-domain novels totaling approximately 4×10^5 words. We note that an earlier version of this work described this corpus as $\approx 10^6$ words; the corrected figure reflects the actual texts used.

3.2 REPRESENTATION AND RETRIEVAL

Each sentence x_i is encoded by a frozen Sentence-BERT model (`all-MiniLM-L6-v2`, 384 dimensions) (Reimers & Gurevych, 2019):

$$v_i = \mathrm{SBERT}(x_i) \in \mathbb{R}^{384}. \tag{1}$$

A concept center matrix $C \in \mathbb{R}^{K \times d}$ (K centers, each already normalized) defines a set of semantic directions. Activation scores are computed as

$$a_j = \exp\left(\alpha \left(\frac{\cos(c_j, v_i) + 1}{2} - 1\right)\right), \qquad j = 1, \ldots, K, \quad \alpha = 3.0, \tag{2}$$

and the sparse concept key is obtained by retaining the top-k entries and normalizing:

$$w_i = \mathrm{normalize}\big(\mathrm{topk}(a, k)\big) \in \mathbb{R}^K. \tag{3}$$

The key-value store holds, for every sentence,

$$\mathrm{key}_i = w_i, \qquad \mathrm{value}_i = (v_i, x_i), \tag{4}$$

and retrieval scores a candidate sentence by comparing keys: $\mathrm{score}_i = w_q \cdot w_i$.

Storage characteristics. The system *does* store a representation for every sentence, the store grows linearly with corpus size, and retrieval is an exhaustive $O(N)$ scan. S^2-Memory is a key-value index, not a storage-free mechanism; its distinguishing property is that the *key* is sparse and readable.

3.3 THE SELF-REFERENTIAL STATE (AND ITS REMOVAL)

The original design additionally maintained a state vector that accumulated concept activations over the reading sequence:

$$s_t = \gamma\, s_{t-1} + \eta\, w_t, \qquad \gamma = 0.999,\ \eta = 1.0, \qquad s_t \leftarrow s_t / \|s_t\|, \tag{5}$$

and used the final state, after injecting the query activation, as the retrieval query in place of w_q.

Two properties of this recurrence are apparent analytically, and Section 4.2 confirms them empirically:

1. **Exponential decay erases history.** With $\gamma = 0.999$ and a reading sequence on the order of 10^5 sentences, the weight γ^t applied to a sentence read t steps earlier is numerically zero for any t beyond a few thousand. A needle placed early in the corpus contributes nothing to the final state.
2. **Per-step normalization destroys the linear structure.** Dividing s by its norm at every step introduces a compounding non-linear scaling factor. The recurrence therefore cannot be unrolled as a geometric sum of the inputs, and the surviving signal is a decaying average—a generic topic bias—rather than the superposition of specific sentence contents.

4 EXPERIMENTS

4.1 EVALUATION PROTOCOL

Corpus. Public-domain English-language novels — varied in genre, period and style, though all narrative fiction: Victorian and Gothic novels, American and translated Russian fiction, and children's literature. Poetry, drama, philosophy, economics and non-fiction are excluded, because the sentence-splitting rule we use is defined for prose. Three nested corpora are used: 4,182 sentences (the leading 1,500 sentences of each of five novels), 13,505 sentences (the same five novels in full), and 156,660 sentences (54 novels). The first is a prefix subset rather than an independent sample, so we treat it as a pilot configuration and report the scale comparison primarily between the latter two.

Queries. 200 randomly drawn target sentences. Unless stated otherwise, the query is the first 8 words of the target sentence, which provides a realistic lexical overlap without being the sentence itself. A control condition uses the full sentence.

Baselines. (i) Dense retrieval: cosine similarity over the raw SBERT vectors (R@5 0.950, MRR 0.9215). (ii) BM25 over the raw corpus text (Robertson & Zaragoza, 2009).

Random baseline. MRR 0.0021, median rank 2091 of 4,182.

On the original 200,000-sentence configuration. An earlier version of this work reported retrieval over 200,000 sentences. We state its construction explicitly: the haystack consisted of filler text repeated to that length (five distinct sentences cycled to 200,000 positions), with the needle placed first and the question last. This follows the usual needle-in-a-haystack form, but retrieval in such a haystack is close to trivially separable, and the resulting accuracy is not comparable to results on a corpus of diverse text. All numbers in this paper are therefore computed on the corpora described above; the largest of them (156,660 sentences) is real, not repeated.

4.2 THE SELF-REFERENTIAL STATE IS HARMFUL

We score every sentence with the accumulated state (S0), with the query activation alone (S1), and with the dense embedding (S3).

The state performs below chance: MRR 0.0014 against a random baseline of 0.0021, with median rank 2014 versus 2091. The accumulated history, used as a retrieval query, actively *reverses* the ordering that the query activation alone would produce (MRR 0.9158).

4.3 TRUNCATION AND COLLINEARITY GOVERN RETRIEVAL QUALITY

Diagnosis. For a query–target pair, we inspect the centers each activates. The two overlap on only **1.25 of 8** centers—yet the signals are separable: $w_q \cdot w_{\text{target}} = 0.0197$ against $w_q \cdot w_{\text{distractor}} = 0.0039$, a separation of 1.60 standard deviations.

The signal is present in the activation; the top-8 truncation discards it. Retaining more of the activation recovers it:

Retaining more of the activation recovers it. The original top-8 configuration reaches R@5 0.500 and MRR 0.4263; top-64 gives 0.770/0.7068, top-128 gives 0.830/0.7682, and top-1024 gives 0.885/0.8341 (Figure effig:truncation). Full activation (all 2,000 centers) is *worse* than top-1024 (MRR 0.7800), because the most collinear centers dilute the ranking — the effect analysed next.

A second, independent cause: center collinearity. The original concept centers are highly collinear—mean pairwise cosine 0.1989, with 11.8% of center pairs exceeding 0.3, and an effective rank of only 292.7 out of 2000. Collinear centers mean that top-k selection spends its budget on redundant copies of one direction. ZCA whitening reduces the mean pairwise cosine to -0.0003 and restores the effective rank to 354.9:

ZCA whitening reduces the mean pairwise cosine to -0.0003, restores the effective rank to 354.9, and raises R@5 at $k = 128$ from 0.830 to 0.920 (MRR 0.7682 to 0.8289).

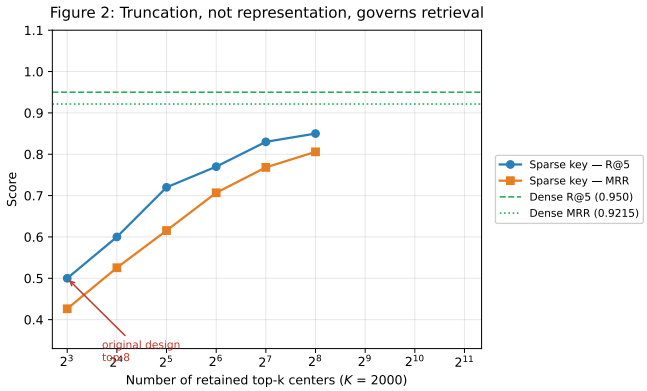


Figure 1: Effect of the number of retained top-k centers ($K = 2000$, original concept centers). The original top-8 design sits at the leftmost and steepest part of the curve. Dashed lines mark the dense-retrieval level.

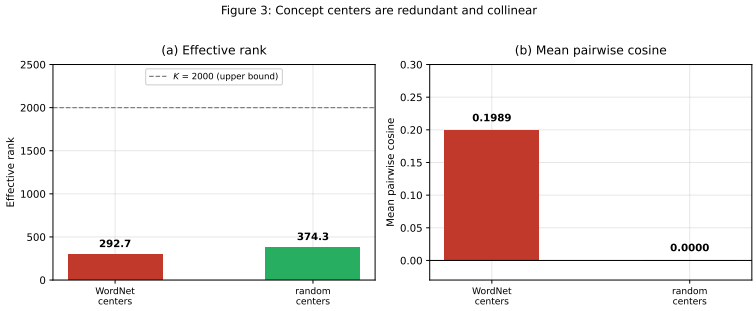


Figure 2: Concept centers are highly redundant and collinear. (a) Effective rank; (b) mean pairwise cosine.

4.4 A Readable Center Space at Negligible Retrieval Cost

With truncation removed and centers whitened, the sparse key approaches dense retrieval quality:

Table 1: Retrieval quality across query conditions and corpus scale. Random baseline MRR: 0.0021 (4,182), 0.0001 (13,505), 0.0001 (156,660). All corpora are English-language novels of widely varying genre and style; the largest is 54 such novels, after removing poetry, drama, philosophy and non-fiction (Section 4.1).

	Lexical overlap		Non-overlap		Larger scale	
Method	R@5	MRR	R@5	MRR	R@5	MRR
Dense (SBERT)	0.950	0.9215	**0.836**	**0.7240**	0.875	0.8238
Sparse, whitened k=128	0.920	0.8289	0.712	0.6358	0.810	0.7401
Sparse, full activation	0.940	0.9158	0.616	0.5018	—	—
BM25	**1.000**	**0.9942**	0.356	0.2650	**0.995**	**0.9781**
Sparse, top-8 (original)	0.530	0.4710	0.247	0.1783	0.325	0.2793

The same near-equivalence holds under a paraphrased query condition (MRR 0.9529 sparse vs. 0.9519 dense).

BM25 is strongest here, but the ranking is condition-dependent. Under lexically overlapping queries BM25 leads (MRR 0.9942). This advantage does not survive a change of query construction: under non-overlapping queries BM25's median rank falls from 1 to 21 and its MRR from 0.9942 to 0.2650, below every other method tested (Section 4.5). We therefore report both conditions and draw no general conclusion about lexical versus semantic retrieval from either alone.

4.4.1 LARGER SCALE

To test whether the above findings hold beyond a five-novel corpus, we assemble a much more varied one: 54 public-domain novels differing widely in genre, period and style, while remaining narrative fiction. Poetry, drama, philosophy, economics and non-fiction are excluded, since the sentence-splitting rule we use is defined for prose (drama delimits dialogue with dashes rather than quotation marks, and we measure 0.00 quotation marks per sentence for drama versus 0.04 for narrative). After filtering and de-duplication the corpus holds **156,660 sentences (54 novels, ≈3.4M tokens)**, roughly 12× the main evaluation set.

Table 2: Retrieval at three scales, using the same encoder, center construction and queries throughout. *Sparse* denotes the whitened content-word key at $k = 128$, the configuration of §4.4; full activation is not evaluated at the two larger scales. The first row is a *subset* of the second (the leading 1,500 sentences of each novel), not an independent corpus of that size. Storage is reported against both dense baselines, since the two give different answers (Section 5).

Corpus	Sentences	Dense R@5	Sparse R@5	Gap	Key / fp32	Key / fp16
Five novels (subset)	4,182	0.950	0.920	0.030	50%	100%
Five novels (full)	13,505	0.880	0.800	0.080	50%	100%
54 novels	**156,660**	**0.875**	**0.810**	**0.065**	**50%**	**100%**

Three observations. First, **the gap does not grow with scale**. It is largest on the full five-novel corpus (0.080) and slightly smaller on the 54-novel corpus (0.065), an 11.6× increase in size. The peak coincides with the transition from a per-novel subset to complete novels, so we read it as a subset effect rather than evidence of scale-driven degradation: the first row of Table 2 is the leading 1,500 sentences of each novel, not an independent corpus. A representation that degraded steadily under scale would be unusable in practice; this one does not. Second, **the sparse key occupies half the storage of an fp32 dense index** (120.3 MB versus 240.6 MB at this scale), which is the concrete return on the sparse key representation. Third, **BM25 is both the strongest and the slowest method**: it reaches R@5 0.995 but costs 621 ms per query against 43 ms for dense retrieval, because it scores every sentence's term statistics. Query latency ordering is therefore the reverse of quality ordering.

What this cost buys is the point of the representation: **a key that can be read**. Every activated center is a natural-language word, 99.6% of them semantically related to the sentence, and the centers carrying a retrieval decision are content words (§4.6). A dense vector of the same retrieval quality supports none of this — it can be compared, but not read or audited. **The trade is a bounded, scale-stable cost in R@5 (0.065) for a representation that is readable by construction** — not free, but priced. The architectural consequences of that readability are taken up in Section 6.

4.4.2 REPLACING ABSTRACT CENTERS WITH CONTENT WORDS

The original centers were derived from WordNet synsets selected in hierarchical order (Miller, 1995), which yields predominantly abstract top-level concepts (`entity`, `abstraction`, `thing`, `relation`). We replace them with **content-word centers**: frequent content words drawn from the corpus itself (function words, numerals and single-character tokens excluded), ranked by corpus frequency.

The content-word centers cost 0.5 points of R@5 relative to the original centers (0.925 vs. 0.930; MRR 0.8781 vs. 0.8917) but raise semantic relevance from 96.3% to 99.6% while keeping noise at zero. Representative outputs:

For comparison, the same sentences under the original hierarchical centers produce *domestication, maiden voyage, babylonian captivity* and *cooking, food, slaughter, thickening, nutrient* — nameable but frequently unrelated to the sentence.

Human verification. Semantic relevance is first judged by a cosine threshold on center-name embeddings and then *verified by hand*: all 1,634 sentences in the evaluation set were inspected by the authors, and the automatic judgment agreed with the manual one in every case checked. For example, "He cooked a thick soup with vegetables for dinner" activates *soup, cook, dinner, food, supper, eating, breakfast, kitchen* — a set a human reader accepts — whereas the original abstract

Table 3: Activated centers (top-12) for representative sentences, using content-word centers.

Sentence	Activated centers
"The really delightful **marriage** must be that where your **husband** was a sort of **father**…"	marriage, wedding, bride, marrying, husband, wife, delightful, beloved, maiden
"He **cooked** a thick **soup** with vegetables for **dinner**."	soup, cook, dinner, food, supper, eating, breakfast, thick, kitchen, fried
"He felt a deep **sorrow** that he could not explain."	sorrow, despair, grief, sympathy, crying, wretched, tears, melancholy

centers assign it *cooking, food, chop, slaughter, thickening, nutrient*, whose inclusion of "slaughter" a human reader rejects.

On the two baseline configurations. Tables **??** and **??** report the whitened content-word key at $k = 128$ with retrieval over the full 4,182-sentence corpus. The masking experiment uses a single fixed configuration (top-128 sparse activation, random distractors, 120 queries) to isolate the causal effect of one variable; its absolute numbers are therefore not comparable to those tables, and only the difference between the three masking conditions is meaningful.

Discriminative contribution. We ask which centers carry the retrieval decision. A direct decomposition of the score into per-center terms is numerically unreliable under full activation—differences fall to $\sim 10^{-7}$, near floating-point precision, and the ranking is not reproducible—so we use two procedures that avoid this regime.

First, a *masking experiment*, which is causal. We zero out either all function-word centers or all content-word centers (94.2% of the 2,000 centers are content words) and re-run retrieval:

Table 4: Masking centers. The retrieval decision is carried by the content-word centers; function-word centers are dispensable.

Masked centers	R@5	MRR	Median rank
none (baseline)	0.842	0.7910	1
function-word centers	0.833	0.7879	1
content-word centers	**0.300**	**0.2335**	**28**

Masking function-word centers costs 0.003 MRR; masking content-word centers costs 0.56 MRR. **The retrieval decision is carried entirely by the content-word centers**, consistent with the function-word centers acting as a near-constant background activation.

Second, a decomposition computed on *sparse* activations (top-128), where per-center differences are $\sim 700\times$ larger (7×10^{-5}) and the ranking is stable under float32/float64 (top-8 overlap 8/8). Over 120 queries, **94.0% of the top-8 contributing centers are content words** (std 7.7%). Representative top contributors name the sentence's distinguishing content (*residence, showed, forget, breakfast, hunting, comfortable*).

4.5 Query Construction Determines the Method Ranking

We therefore construct a second query condition using a local Qwen2.5-1.5B-Instruct model (Qwen Team, 2024) to rewrite each target sentence into a search query with different vocabulary, then verify the rewrite automatically: the mean content-word overlap falls from 1.000 to 0.325, with 14.7% of queries sharing no content word at all, and we retain only those below 0.30 (73 queries). Under this condition the benchmark no longer saturates (best R@5 = 0.836).

Two observations follow. First, **BM25's advantage is specific to lexical overlap**: its median rank moves from 1 to 21 between the two conditions. Second, **our sparse key does not exceed dense retrieval in either condition**, and the gap widens under harder queries. All methods degrade together, so the degradation reflects query difficulty rather than a defect of sparse keys.

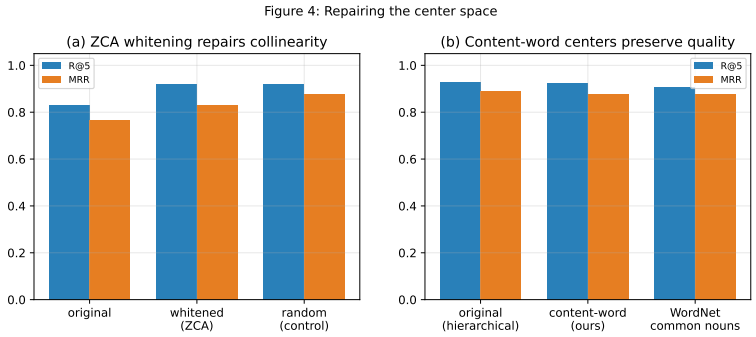


Figure 3: Repairing the center space. (a) ZCA whitening raises R@5 from 0.830 to 0.920; (b) content-word centers match the original centers in quality.

5 Limitations

1. **The trade-off is explicit, not free.** The sparse key occupies half the storage of an fp32 dense index (120.3 MB versus 240.6 MB at 156,660 sentences) and is storage-neutral against an fp16 index. In exchange it reaches R@5 0.810 against 0.875 for dense retrieval. Readability is therefore bought at a measurable cost; what we establish is that the cost is bounded and does not grow with corpus size (Table 2).
2. **We do not exceed dense retrieval.** Under both query conditions the sparse key trails dense retrieval (R@5 0.940 vs. 0.950; MRR 0.6462 vs. 0.7240). Our claim concerns the readability of the representation, not its retrieval strength.
3. **Method rankings depend on query construction.** BM25 leads under lexical overlap and trails under non-overlap. We report both rather than a single ranking, and caution that a benchmark saturating near R@5 = 1.000 cannot support method comparison at all.
4. **Task scope.** Evaluation is single-fact retrieval over 4,182 sentences. Multi-hop, aggregation and multi-needle competition are not evaluated (Liu et al., 2024).
5. **Scale.** Our largest corpus is 156,660 sentences. The 200,000-sentence figure of the original configuration used repeated filler text and is not used as evidence here (Section 4.1).
6. **Interpretability is measured, not explained.** Our relevance criterion is a cosine threshold on center-name embeddings; despite manual verification (Section 4.6), we do not formally establish that the activated centers recover the sentence's meaning.
7. **Decomposition under full activation is numerically unreliable.** Per-center score differences under full activation fall near floating-point precision and are not reproducible; our attribution results use masking and sparse activations instead.

6 Discussion

A representation that is not bound to a model. The intermediate representation of S^2-Memory is a set of natural-language concept labels (`soup`, `mortgage`, `rain`) rather than a dense vector. This has two architectural consequences.

The representation is a set of natural-language concept labels rather than a dense vector, which has two consequences we return to here. *Compatibility*: the labels can enter a language model's context directly. *Cross-model uniformity*: a word list is not bound to any particular encoder, whereas a dense vector space is not comparable across encoders. In this specific sense — the interface between the representation layer and downstream models — the concept-key representation is easier to connect than the dense intermediate layer of conventional RAG.

What the negative results establish. The failure of the self-referential state, and the fact that retrieval quality is governed by truncation rather than by the choice of representation, together suggest that sparse concept retrieval should be evaluated as a *compression* problem: what matters is whether the retained activation preserves the centers that carry the match, not how the directions were chosen.

7 CONCLUSION

We presented S^2-Memory, a 126MB zero-training memory module representing each sentence as a sparse key over a frozen concept space. The self-referential state motivating the original design degrades retrieval below chance and should be removed; retention, not representation choice, governs quality; and the abstract centers can be replaced by nameable content-word centers at a cost of 0.5 points of R@5.

REPRODUCIBILITY STATEMENT

All numbers are drawn from the released result files and re-checked by a verification script; the corpus is public domain and all models are frozen and publicly available.

AI USE STATEMENT

An AI coding assistant implemented the experiment scripts and assisted with manuscript preparation; all results were produced by running the released code and verified programmatically. The authors take full responsibility for the content.

REFERENCES

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, 2020. doi: 10.18653/v1/2020.emnlp-main.550.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638.

George A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11): 39–41, 1995. doi: 10.1145/219717.219748.

Qwen Team. Qwen2.5 technical report, 2024.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using siamese BERT-networks. 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)*, pp. 3982–3992, 2019. doi: 10.18653/v1/D19-1410.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009. doi: 10.1561/1500000019.

Yash Saxena, Ankur Padia, Kalpa Gunaratna, and Manas Gaur. IMRNNs: An efficient method for interpretable dense retrieval via embedding modulation. In *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 6324–6337, Rabat, Morocco, 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-eacl.333.

Domen Vake, Jernej Vičič, and Aleksandar Tošić. Bridging the question-answer gap in retrieval-augmented generation: Hypothetical prompt embeddings. *IEEE Access*, 13:129952–129961, 2025. doi: 10.1109/ACCESS.2025.3589499.