ExplicitLM

Decoupling Knowledge from Parameters via Explicit Memory Banks

Paper authors: Chengzhang Yu, Zening Lu, Chenyang Zheng, Chiyue Wang, Yiming Zhang, and Zhanpeng Jin

arXiv:2511.01581v1 · Reading: 8 October 2026
Research group discussion · approximately 25 minutes

A readable memory interface improves access to preloaded facts.

Give factual knowledge an address

Research question

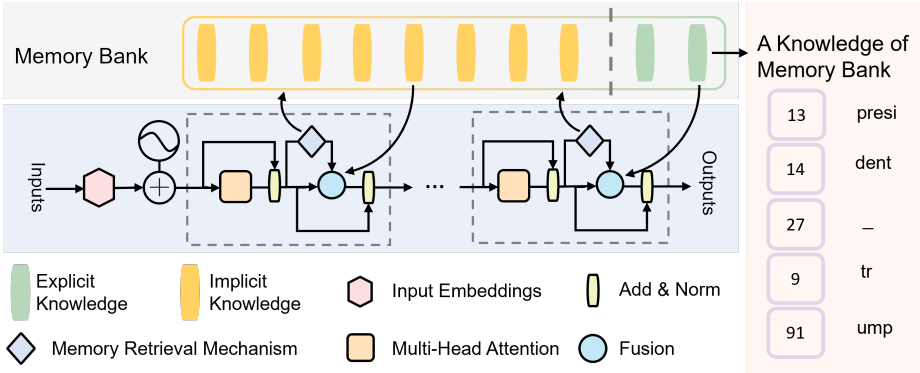
Can a language model read inspectable knowledge entries while learning how to retrieve and use them?

- Distributed parameters provide no direct record for each fact.
- ExplicitLM stores facts as token sequences in an external bank.
- Learned retrieval connects those records to network computation.

The experiments primarily test access to preloaded facts.

Paper, Secs. 1 and 3, PDF pp. 1–6; evaluation scope from Sec. 4.2, p. 7.

All layers access a shared memory bank



One shared bank $\longrightarrow$ layer-specific reads $\longrightarrow$ fusion with network computation.

Paper, Fig. 1 and Sec. 3.2, PDF p. 4. The exact fusion operator is not specified.

Readable content and retrieval keys are distinct

$$\mathbf{m}_i = \mathrm{Tokenize}(s_i), \qquad \mathbf{E}_i = \mathrm{Embed}(\mathbf{m}_i) \in \mathbb{R}^{d \times L}$$

Stored content

$N = 10^6$ entries
Up to $L = 16$ token IDs per entry

Decodable into text

Learned addressing

Query $q(x) \in \mathbb{R}^d$
Key $\mathbf{k}_i \in \mathbb{R}^d$

Selects a memory address

Readability applies to the stored token sequence, not the key vector.

Paper, Secs. 3.2 and 3.4, PDF pp. 4–5.

Freeze facts; reserve capacity for learned patterns

$$\mathcal{M} = \mathcal{M}_f \cup \mathcal{M}_u, \qquad |\mathcal{M}_f| = \rho N, \qquad |\mathcal{M}_u| = (1-\rho)N$$

Frozen partition: 20% by default

200,000 entries
Curated explicit facts
Content preserved during training

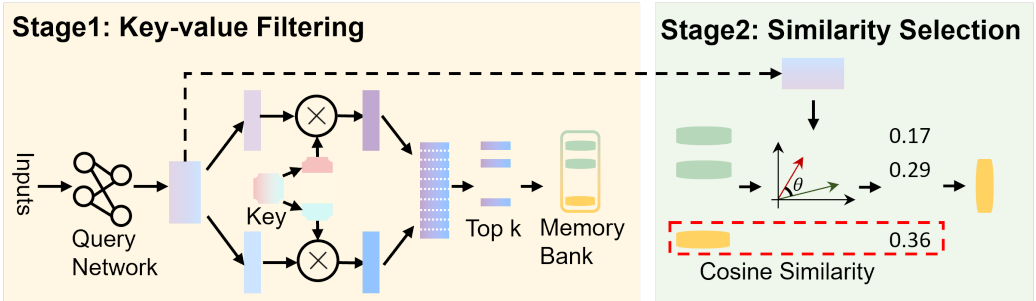
Updatable partition: 80%

800,000 entries
Intended to learn implicit patterns
EMA proposed for gradual updates

Fixed slot count; updates replace content in place.

Paper, Sec. 3.3, PDF pp. 4–5. EMA statistics and conversion back to token IDs are unspecified.

Retrieve candidates, then select a discrete entry



1. Product keys reduce the initial search.
2. Cosine scores and Gumbel-Softmax drive the final choice.

Paper, Fig. 2 and Sec. 3.4, PDF p. 5.

Product keys factor a million-way search

$$\mathbf{k} = [c; c'], \quad q = [q^{(1)}; q^{(2)}], \quad q^\top \mathbf{k} = (q^{(1)})^\top c + (q^{(2)})^\top c'$$

- Two sets of $\sqrt{N} = 1{,}000$ subkeys address $N = 10^6$ entries.
- Retain high-scoring subkeys and combine their scores.
- Teaching example: top 16 from each set gives $16^2 = 256$ pairs.

Scope of the efficiency claim

Paper: $O(N|I|) \to O(\sqrt{N}|I|)$ for retrieval.
Subkey scoring, candidate selection, and memory fusion still cost work. No end-to-end latency benchmark is reported.

Paper, Sec. 3.4, p. 5; 16 candidates in Sec. 4.6, p. 9. Score expansion is a teaching derivation.

Hard choices use a soft gradient approximation

$$p_i = \frac{\exp((cs_i + g_i)/\tau)}{\sum_{j \in I} \exp((cs_j + g_j)/\tau)}, \qquad cs_i = \cos(q, \mathbf{k}_i)$$

$$g_i = -\log(-\log \epsilon_i), \quad \epsilon_i \sim U(0,1), \qquad \hat{i} = \arg\max_i p_i$$

- Forward: read the discrete entry $\mathbf{m}_{\hat{i}}$.
- Backward: use the soft selection distribution as a surrogate.
- Candidate set I; temperature τ controls the soft weights.

The first-stage candidate indices remain discrete.

Paper, Eq. (1) and Sec. 3.4, PDF p. 5. Inference noise policy is not specified.

Three objectives couple prediction and memory use

$$\mathcal{L} = \mathcal{L}_{\mathrm{CE}} + \lambda_{\mathrm{sim}}\mathcal{L}_{\mathrm{sim}} + \lambda_{\mathrm{div}}\mathcal{L}_{\mathrm{div}}$$

Component	Intended effect
Cross-entropy	Predict the next token using context and memory.
Relevance	Favor memories with high query–content similarity.
Diversity	Reduce similarity among retrieved candidates.

Relevance and diversity create a trade-off controlled by the weights.

Paper, Sec. 3.5, Eqs. (2)–(4), PDF p. 6. Exact notation issues are discussed in backup slides.

Test facts are available inside the memory bank

- **Corpus:** 10M entries from Wikipedia, Gutenberg, OpenWebText.
- **Memory initialization:** Wikipedia only; source UUIDs retained.
- **Split:** test examples derive from frozen memory entries; training excludes those entries' tokens, as stated in the paper.
- **Comparison:** ExplicitLM vs. a Transformer without memory; SFT data varies from 10k to 100k examples.

What this protocol measures

Learning to use externally supplied facts beyond the training examples. The baseline has no equivalent retrieval access.

Paper, Secs. 4.1–4.3, PDF pp. 6–7. Model size and matched compute budgets are not specified.

Three tasks probe different uses of stored facts

Task	Input → output	Main capability
Object prediction	Subject, relation → object	Exact factual recall
Relation reasoning	Entity pair → relation	Relation identification
Fact verification	Statement → true/false	Fact discrimination

All results report accuracy (%, higher is better).

- Object prediction includes five distractors in the stated protocol.
- Fact verification uses a 50:50 positive/negative balance.

Paper, Sec. 4.2, PDF p. 7. No separate multi-hop reasoning benchmark is reported.

Object prediction shows the most persistent gains

SFT data	Baseline (%)	ExplicitLM (%)	Gain (pp)
10k	7.86	28.42	**20.56**
25k	22.16	63.12	**40.96**
50k	30.23	73.90	**43.67**
75k	40.64	79.76	**39.12**
100k	56.80	80.94	**24.14**

Two different quantities

10k: $28.42/7.86 = 3.62\times$ the baseline accuracy.
50k: $73.90 - 30.23 = 43.67$ percentage points.

Paper, Table 1, PDF p. 7. The abstract's 43.67% improvement is an absolute accuracy difference.

Other task gaps narrow with more supervision

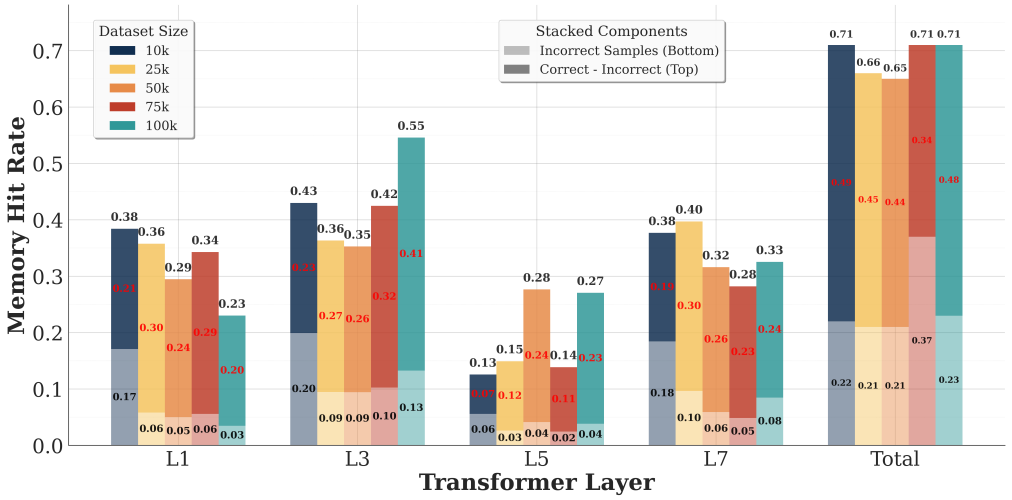
	Relation reasoning		Fact verification	
SFT data	Baseline	ExplicitLM	Baseline	ExplicitLM
10k	38.27	70.02	61.71	66.03
25k	79.99	87.85	71.49	79.79
50k	83.80	90.41	83.34	86.25
75k	87.66	92.12	86.40	88.74
100k	91.91	92.73	88.92	89.75

Relation gain: **31.75 pp at 10k** $\rightarrow$ **0.82 pp at 100k**.

Fact verification gain at 100k: **0.83 pp**.

Paper, Table 1, PDF p. 7. All entries are accuracy (%). At 25k, fact verification improves by 8.30 pp; the printed delta is 8.33.

Memory access varies across network depth



Paper, Fig. 3, PDF p. 8. Full bar: correct. Bottom: incorrect. Top: difference.

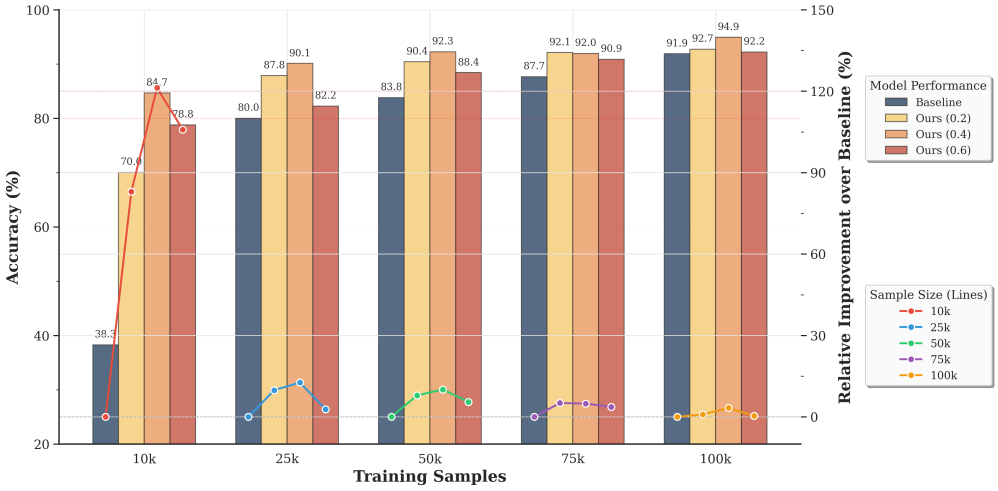
Correct predictions have higher memory hit rates

SFT data	Hit | correct	Hit | incorrect	Gap (pp)
10k	71%	22%	49
25k	66%	21%	45
50k	65%	21%	44
75k	71%	37%	34
100k	71%	23%	48

- Total hit: at least one layer retrieves relevant memory.
- These are $P(\text{hit} \mid \text{correct})$ comparisons: evidence of association, not a causal effect estimate.

Paper, Fig. 3, PDF p. 8. Values follow figure labels; Sec. 4.4 prose says 5k where the figure says 75k.

The frozen fraction creates a nonmonotonic trade-off



Paper, Fig. 4, PDF p. 9. Left: accuracy (starts at 20%). Right: relative gain over baseline.

A 40% frozen bank often beats the 20% default

SFT data	Baseline	$\rho = 0.2$	$\rho = 0.4$	$\rho = 0.6$
10k	38.3	70.0	**84.7**	78.8
25k	80.0	87.8	**90.1**	82.2
50k	83.8	90.4	**92.3**	88.4
75k	87.7	**92.1**	92.0	90.9
100k	91.9	92.7	**94.9**	92.2

At 10k, $\rho = 0.4$ adds **14.7 pp** over the default.

Hyperparameter choice matters even when every setting beats baseline.

Paper, Fig. 4, PDF p. 9; values as labeled to one decimal. Whether test facts are held fixed as the frozen partition changes is not specified.

Oracle memory replacement improves accuracy

Replace the top-ranked of 16 candidates at **L1 and L3** with the answer-relevant oracle entry.

Gain over Retain (pp)	50k	75k	100k
Object prediction	3.62	2.73	1.88
Relation reasoning	2.25	1.62	1.75
Fact verification	2.12	2.25	0.75

Mean gain: **2.11 percentage points** across nine comparisons.

- Direct intervention supports a contribution from retrieval content.
- Only two layers are changed; this is not a universal upper bound.

Paper, Sec. 4.6 and Table 2, PDF p. 9. Retain values differ from Table 1; comparisons stay within Table 2.

The strongest evidence concerns knowledge access

Design and reported evidence	What the broader claims still need
Better accuracy with preloaded memory	Comparisons with equal knowledge access
Retrieval content affects predictions	Actual post-training fact-editing tests
Readable token entries	Faithfulness of reasoning explanations
Factorized candidate search	Measured latency and memory costs

The backbone still has parameters that can store knowledge.

Reading assessment based on Secs. 3–5, PDF pp. 3–10. Accuracy and intervention evidence: Tables 1–2.

Three experiments would strengthen the central claim

1. **Equal knowledge access.**
   Compare RAG, context insertion, and layer-wise memory on the same facts, backbone, and resource budget.

2. **Controlled fact editing.**
   Change one record; test update success, paraphrases, and side effects.

3. **Complete the implementation and cost picture.**
   Specify EMA-to-token updates, pooling, and fusion; measure latency and ablate auxiliary losses.

Proposed follow-up experiments from this reading. The paper identifies manual knowledge curation as a limitation in Sec. 5, PDF p. 10.

A useful interface for readable external knowledge

1. **Architecture:** token memory, product-key search, and trainable discrete selection.
2. **Evidence:** strong factual recall gains in the reported memory-backed protocol; oracle reads add another 2.11 pp on average.
3. **Open problem:** connect inspectable storage to reliable editing and well-specified, efficient computation.

Chengzhang Yu et al., *ExplicitLM: Decoupling Knowledge from Parameters via Explicit Memory Banks*.
arXiv:2511.01581v1, 3 November 2025.

Main discussion ends here. Four backup slides follow.

Backup: a consistent straight-through surrogate

$$h_i = \mathbf{1}[i = \hat{i}], \qquad z_i = \mathrm{sg}(h_i - p_i) + p_i, \qquad y = \sum_{i \in I} z_i v_i$$

- Forward: $z = h$, so the read is the selected value $v_{\hat{i}}$.
- Backward: differentiate through p; treat $\mathrm{sg}(\cdot)$ as constant.

$$\frac{\partial p_i}{\partial cs_j} = \frac{p_i}{\tau}(\delta_{ij} - p_j)$$

Here v_i is a pedagogical memory vector; candidates, noise, and values are held fixed for the score derivative.

Teaching derivation for Sec. 3.4, PDF p. 5. The paper's displayed query-gradient expression omits the selection Jacobian and differentiates discrete IDs.

Backup: two loss definitions need clarification

Relevance loss, Eq. (3):

$$-\mathbb{E}_x \left[\sum_{i \in I} p_i \frac{q(x)^\top \mathbf{E}_i}{\|q(x)\|_2 \|\mathbf{E}_i\|_2} \right]$$

$q \in \mathbb{R}^d$, but $\mathbf{E}_i \in \mathbb{R}^{d \times L}$. An aggregation into a compatible vector is needed.

Diversity loss: $K = |I|$, $\hat{\mathbf{E}}_i$ is normalized.

$$\frac{2}{K(K-1)} \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq i}} \cos(\hat{\mathbf{E}}_i, \hat{\mathbf{E}}_j)$$

Ordered pairs are counted twice. This equals twice the usual mean pairwise cosine, affecting the effective regularization weight.

Paper, Sec. 3.2, p. 4, and Eqs. (3)–(4), p. 6. Both observations are this reading's analysis.

Backup: full oracle intervention results

SFT	Condition	Object	Relation	Fact
50k	Retain	70.87	89.87	85.12
	Replace	74.49	92.12	87.24
75k	Retain	77.12	90.25	88.00
	Replace	79.85	91.87	90.25
100k	Retain	79.12	90.50	90.37
	Replace	81.00	92.25	91.12

Accuracy (%). Average gain by data scale: 2.66, 2.20, 1.46 pp.

Paper, Table 2, PDF p. 9. At 50k, Table 1 object accuracy is 73.90%, whereas this table's Retain is 70.87%; the run/configuration difference is unexplained.

Backup: details that affect reproduction

- **Task scoring:** five distractors would suggest six choices; uniform chance would be 16.67%, above the 10k baseline's 7.86%. Candidate construction and scoring need clarification.
- **Model/training:** model size, optimizer settings, temperature schedule, loss weights, and EMA-to-ID mapping are not specified.
- **Figure 3:** use the plotted 75k label; the prose says 5k. The internal stacking legend also differs from the caption.

Appendix A.1 states NVIDIA A100 hardware and links an anonymous repository.

Paper, Secs. 3–4, PDF pp. 4–9, and Appendix A.1, p. 12. Statements concern the paper's specification.