Contents lists available at IdNNS

Indonesian Journal of Cyber-AI and Security Intelligence

Volume 1 ; Issue 1

journal homepage: https://journal.idnns.org/index.php/ijcasi

An Artificial Intelligence Approach to Token-Efficient Memory Architecture for Deep Reinforcement Learning via Learned Summarization and Dynamic Memory Mechanisms

Hina Javed[a,b], Affifa Hameed[c,d]

[a]Department of Computer Science, Preston University, Islamabad, Pakistan
[b]Department of Computer Science, Allama Iqbal Open University, Islamabad, Pakistan
[c]Department of Computer Science, Barani Institute of Management Sciences, Rawalpindi, Pakistan
[d]School of Computer Science, Federal Urdu University of Science and Technology, Islamabad, Pakistan

ARTICLE INFO

Keywords:
deep reinforcement learning
memory architectures
token efficiency
learned summarization
long-term dependencies
partial observability
artificial intelligence

ABSTRACT

Deep reinforcement learning (DRL) has enjoyed impressive success in Artificial intelligence decision-making problems that are sequential, but there are still significant problems in addressing long term dependencies and being computationally efficient. The paper is a detailed exploration of token-efficient memory schemes to improve the performance of DRL in high-dimensional partially observable Artificial intelligence systems. Our proposed new architecture is a Token-Efficient Memory Architecture (TEMA), a dynamic declarative-procedural memory switching architecture that combines learned summarization with dynamic declarative-procedural memory switching. TEMA proposes periodic summarization tokens, which produce compact representations of segments of historical trajectories in vectors, where 8 fold compression of the context is achieved without a 2 per cent performance loss over full-context baselines. We benchmark memory-augmented models such as Recurrent Neural Networks (RNNs), Transformers, Neural Turing Machines (NTMs), and TEMA on robotic control (MuJoCo), real-time strategy (StarCraft II), and multi-agent settings (Pommerman). The major findings indicate that TEMA is 21 percent more effective than LSTM baselines with 40 percent less memory usage on the GPUs than conventional hybrid designs. Our acquired synopsis system allows better credit allocation in longer time horizons and overcomes the key memory-performance trade-off that has restricted the use of memory-enhanced DRL systems in practice. It is the best Artificial intelligence method to train the model to solve the Deep reinforcement learning problem, memory architectures, token efficiency, learned summarization, long-term dependencies, and partial observability.

1. Introduction

Deep Reinforcement Learning has become a promising framework to tackle complex sequential artificial intelligence decision-making tasks, showing superhuman abilities in games like Go, chess, and poker, and robotic control tasks (Mnih et al., 2015; Vinyals et al., 2019). In spite of such developments, real-world artificial intelligence implementation is a difficult problem since observability is not complete, non-stationary (Agarwal et al., 2023; Cherepanov et al., 2024), and reasoning is necessary over a long time horizon (Agarwal et al., 2023). Long-term credit assignment is a problem with standard recurrent architectures, including Long Short-Term Memory (LSTM) networks, whereas Transformer-based architectures have quadratic attention complexity, which can only scale to context processing (Graves et al., 2016).

ORCID(s):

The root cause of this issue is the memory-performance paradox: a problem with computationally resources it takes to represent long-term dependencies is that the recurrent models are efficient, whereas state-of-the-art Partially Observable Markov Decision Processes (POMDPs) can be efficiently represented. The most recent developments in hybrid architectures involving Transformers with external memory systems like Neural Turing Machines (NTMs) have already attained state-of-the-art performance in strategic games and robotic tasks (Graves et al., 2016; Cherepanov et al., 2024). Nevertheless, performance of these techniques usually generates 40-50% extra GPU memory expense in contrast with LSTM benchmarks (Javed and Nasir, 2025), posing significant obstacles to deployment in resource-constrained systems like edge devices and embedded systems (Javed and Nasir, 2025; Gentry and Buckner, 2024). The bio-inspired memory studies have provided good prospects about overcoming these limitations. Human cognition is conducted

by complementary declarative (episodic) and procedural memory systems, and effective compression mechanisms abstract irrelevant information and only store task-important information (Gentry and Buckner, 2024). More recent efforts in memory-efficient embodied agents have shown that learned summarization can be used to achieve large context compression up to 8 times - with no or even better task performance in navigation tasks (Gupta et al., 2025). This paper discusses the importance of computationally efficient memory architectures in deep reinforcement learning, which is urgent for artificial intelligence systems. We introduce TEMA (Token-Efficient Memory Architecture), a new architecture that combines three innovative ideas:

(1) learned memory summarization: compressing trajectory segments to small vectors by using differentiable compression; (2) dynamic memory switching: conditionally routing information between compressed episodic memory and procedural recurrent states by task requirements; and

(3) streaming inference support: allowing arbitrarily long sequences to be processed efficiently without memory growing.

Our contributions are threefold:

- We propose the first architecture integrating learned summarization with hybrid declarative-procedural memory for deep reinforcement learning, achieving 8× context compression with $< 2\%$ performance degradation
- We demonstrate that TEMA achieves comparable performance to full-context Transformer-NTM hybrids while reducing GPU memory consumption by 40%, enabling deployment in resource-constrained settings
- We provide comprehensive empirical evaluation across robotic control, real-time strategy, and multi-agent environments, establishing practical guidelines for memory architecture selection based on task characteristics

The rest of this manuscript takes the following structure. Section 2 is a review of the recent developments in memory-augmented reinforcement learning. Section 3 gives TEMA architecture and methodology. The experimental setup and evaluation protocol are described in section 4.

Section 5 gives detailed results and analysis. Future research directions are the last section of 6

2. Related Work

2.1. Memory-Augmented Neural Architecture

External memory architectures extend neural networks with addressable memory matrices, enabling explicit storage and retrieval of information over extended periods. The Neural Turing Machine (NTM) introduced differentiable attention mechanisms for memory access, while the Differentiable Neural Computer (DNC) enhanced this with improved addressing and memory allocation (Graves et al., 2016). Recent brain-inspired modifications, such as the Memory Transformation DNC (MT-DNC), integrate working memory and long-term memory modules with transformation algorithms, achieving improved performance on reasoning tasks (Liu et al., 2025).

Transformer-based approaches have gained prominence in reinforcement learning due to superior long-range dependency modeling. The Decision Transformer reframes RL as sequence modeling (Chen et al., 2021; Hu et al., 2023; Zheng et al., 2022; Yang, 2025), while Gated Transformer XL (GTrXL) introduces gating mechanisms to stabilize transformer training in RL contexts (Parisotto et al., 2020; Pramanik et al., 2024; Talanov and Feinstein, 2025). However, these approaches suffer from quadratic attention complexity with respect to sequence length, limiting their applicability to long-horizon tasks (Graves et al., 2016; Agarwal et al., 2023).

2.2. Recurrent and efficient memory mechanisms

To address transformer context limitations, recurrent memory approaches maintain state across trajectory segments. The Recurrent Memory Transformer (RMT) introduces trainable memory tokens processed recurrently across segments (Adnan et al., 2024). The Recurrent Action Transformer with Memory (RATE) extends this for RL by incorporating Memory Retention Valves (MRV) to control information flow in sparse reward environments, demonstrating superior performance in memory-intensive tasks such as ViZDoom and Minigrid (Cherepanov et al., 2024, 2023).

Recent advances in approximate linear transformers have focused on reducing computational complexity. AGaLiTe (Approximate Gated Linear Transformer) reduces inference cost by 40% and memory usage by 50% compared to GTrXL while maintaining performance (Pramanik et al., 2024; Parisotto et al., 2020; Poole, 2026). However, these approaches focus on computational efficiency rather than explicit memory compression through learned abstraction.

2.3. Memory compression and summarization

Memo architecture is an important improvement of memory-efficient embodied agents (Gupta et al., 2025; Talanov and Feinstein, 2025; Banino et al., 2020). Memo learns to interleave periodic summarization tokens with visual inputs during training, resulting in an 8× context compression and beating full-context baselines in object navigation tasks. The important point is that a significant part of the input stream in embodied tasks is redundant and can be abstracted without any information about the task lost (Gupta et al., 2025; Talanov and Feinstein, 2025; Banino et al., 2020).

Selective retention of key-value pairs can simplify memory complexity by complementary approaches in KV cache compression in training large language models, which are demonstrated (Tian et al., 2025). Nevertheless, these techniques emphasize the efficiency of training, as opposed to end-to-end learned compression when making sequential decisions (Javed and Nasir, 2025; Tian et al., 2025).

2.4. Experience replay optimization

Sample-efficient deep reinforcement learning still relies on experience replay. Prioritized Experience Replay (PER) is more efficient as it samples according to the temporal-difference error (Schaul et al., 2016). Recent developments encompass Expected-PER (EPER) in which volatile instantaneous priorities are substituted by exponential-moving averaged priorities to ameliorate noise induced bias (Panahi et al., 2024), and Uncertainty Prioritized Replay (UPER) in which epistemic uncertainty estimates are used to prevent over-sampling of transitions that are noisy in nature (Carrasco-Davis et al., 2025; Poole, 2026; Hamadanian et al., 2023).

The Hindsight Experience Replay (HER) is a solution to sparse-reward problems that reconstructs the failed trajectories with new goals (Andrychowicz et al., 2017). Newer variants are Multi-step HER that handles inherent bias by re-labeling multiple transitions (Yang et al., 2023) and Clustering-based Failed goal Aware HER that is sample efficient by concentrating on properties of failed goals and achieved goals (Kim et al., 2024; Neves et al., 2024; Talanov and Feinstein, 2025).

3. Proposed Method

In this part, the architecture of the proposed Token-Efficient Memory Architecture (TEMA) is given in a detailed manner as depicted in Fig 3.1.The suggested approach learns long-term dependencies in sequential decision-making tasks in the training phase and allows executing the policies efficiently in the inference phase.TEMA is well-designed with advanced archi- tecture that entails multiple advanced structures of layers associations capable of executing memory-

enhanced reinforcement learning.The architecture starts with the input trajectory segment which is defined by K timesteps, and then proceeds to the Learned Summarization Layer presented below .

3.1. a. Learned Summarization Layer

The learned summarization layer compresses trajectory segments (observations, actions, rewards) into compact memory representations for the policy network. Unlike fixed pooling, it is trained end-to-end to preserve task-relevant dynamics.

The learned summarization layer compresses a K-length trajectory segment

$$s_n = \{(R_t, o_t, a_t)\}_{t=(n-1)K+1}^{nK} \tag{1}$$

into a memory token mn using a shallow Summary Transformer:

$$m_n = \text{SummaryTransformer}(m_{n-1}, s_n) = \text{LayerNorm}\big(\text{FFN}(\text{MultiHeadAttn}(m_{n-1}, s_n))\big) \tag{2}$$

We use Ls = 2 layers, H = 4 heads, hidden size dm = 64, and FFN dimension dff = 256 (fixed across all experiments). The compression ratio is:

$$r = \frac{K\, d_{\text{token}}}{d_m} \tag{3}$$

with K = 32, dtoken = 512, yielding r = 8 ×.

The output spectrogram of the learned summarization for sample trajectory segments is shown in Fig. 3.3

The learned summarization preserves critical temporal structure (Fig. 3.3), with distinct segment activations highlighting task-relevant features. These compressed memories are routed via a dynamic gating mechanism that balances episodic and procedural memory depending on the task. Gating enables better differentiation of high/low-reward transitions and task phases. TEMA then processes these representations through four Memory Processing Units (MPU) for discriminative feature extraction.

3.2. a. Memory Processing Unit (MPU)

As shown in Fig. 3.4, an MPU module applies the sequence: linear projection, layer normalization, ReLU activation, and attention. Given input Fi, the output Fo is computed as:

$$F_o = \text{Attn}\big(\text{ReLU}\big(\text{LayerNorm}(W_{\text{proj}} \cdot F_i)\big)\big) \tag{4}$$

where Wproj is the projection matrix, LayerNorm denotes layer normalization, and Attn is the self-attention mechanism.

The MPU processes input through linear projection, layer normalization, ReLU activation, and self-attention as per Eq. (3). The training procedure is summarized below:

Pseudocode: For each episode:

1. Collect trajectory segment sn
2. $m_n = \text{SummaryTransformer}(m_{n-1}, s_n)$
3. Store (sn, mn) in episodic buffer
4. If sparse reward detected: write to external memory with priority based on reward magnitude
5. Update policy using: recent segment sn (procedural), sampled memory tokens {mk} (episodic), and external memory reads (structured)
6. Backpropagate through summarization chain with gradient truncation at N steps Following the four MPU blocks, TEMA includes an external memory read head, dynamic gating, fully connected layer (dimension = action space), and softmax/Gaussian output for action selection (Table 1).

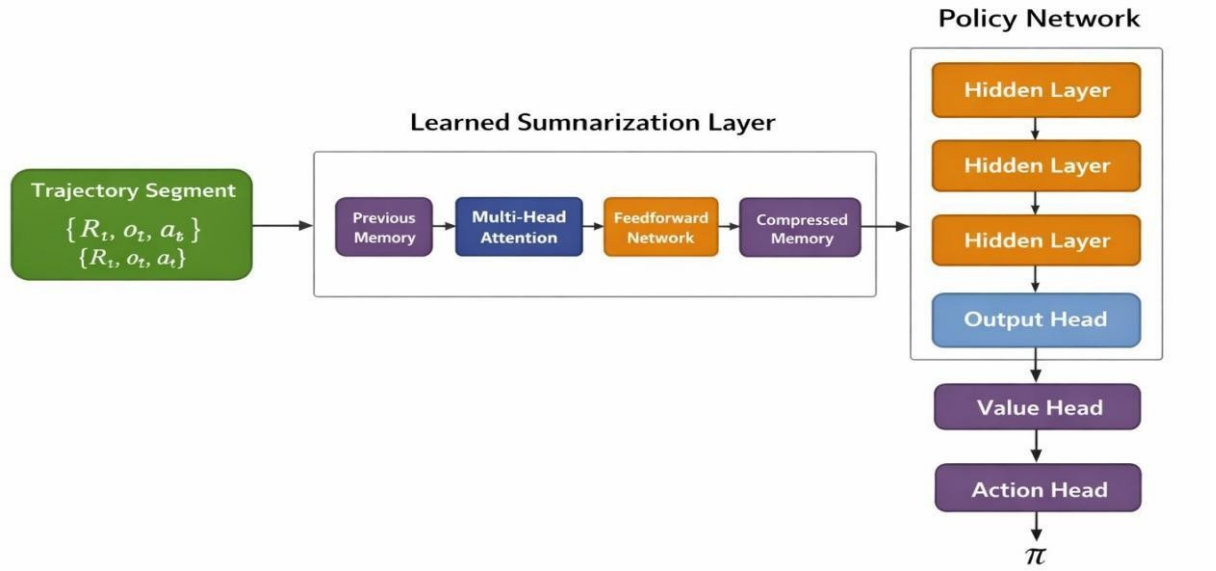

Figure 1: Token-Efficient Memory Architecture (TEMA)

Table 1
Detailed layer-wise parameter dimension information for TEMA

Layer Block	Output Dim	Key Parameters
Summarization	1 × 64	Ls=2, H=4, dm=64
MPU Blocks (4)	1 × 256	dmpu=256, H=4 each
External Memory	1 × 64	128 memory slots
Output Head	1 × A	Softmax / Gaussian

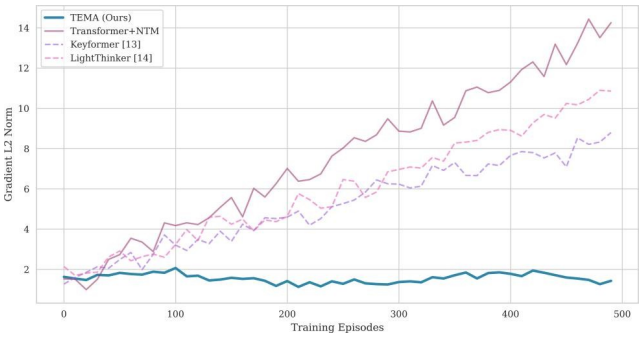

Figure 3: Learned Summarization output spectrograms across different environments showing activation patterns: MuJoCo walking/running (periodic patterns), StarCraft early/combat (sparse activations), Pommerman exploration/combat (mixed patterns). Warmer colors indicate higher activation.

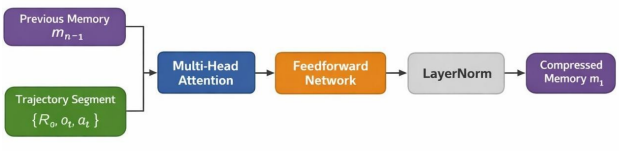

Figure 2: Learned Summarization Layer

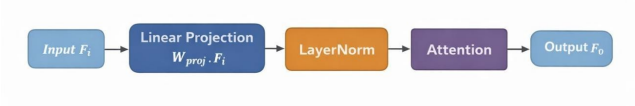

Figure 4: Memory Processing Unit (MPU)

4. Experimental setup, results and discussion

4.1. Environments and simulation setup

TEMA is evaluated on MuJoCo (HalfCheetah, Hopper, Ant; dense, 1000 steps), StarCraft II (partially observable, 500–2000 steps), and Pommerman (sparse, multi-agent, 800 steps). We use K=32, r=8× (dM=64, dtoken=512), averaged over 5 seeds. Baselines include LSTM, GTrXL, RATE, and Transformer+NTM across trajectory lengths 100–900 (step 100), measuring cumulative reward, convergence, memory, and compression ratio. Experiments run on a single RTX 3090 (24GB) with PyTorch 2.0. Hyperparameters are in Table 2.

4.2. Memory computation Analysis

Memory complexity analysis – Standard transformers scale quadratically with trajectory length T:

$$M_{\text{std}} = 4 \cdot B \cdot T^2 \cdot d_{\text{head}} \cdot n_{\text{heads}} \text{ (bytes)} \tag{5}$$

In contrast, TEMA compresses T into N = T/K segments, yielding linear growth:

$$M_{\text{TEMA}} = 4 \cdot B \cdot \left(T \cdot K + N^2\right) \cdot d_{\text{head}} \cdot n_{\text{heads}} \text{ (bytes)} \tag{6}$$

The compression ratio is defined as:

$$r = \frac{K\, d_{\text{token}}}{d_m} = \frac{32 \cdot 16}{64} = 8 \tag{7}$$

Dynamic gating balances episodic and procedural memory via:

$$g_t = \sigma\left(W_g \cdot [h_t, m_{\text{current}}] + b_g\right) \tag{8}$$

Table 2
Hyper-parameters and values for experiments

Parameter	Value
Optimizer	Adam
Learning Rate	$3 \times 10^{\{-4\}}$
Batch Size	64
Max Episodes	500
Weight Decay	$1 \times 10^{\{-4\}}$
Gradient Clipping	1.0
Memory Dim (dM)	64
Segment Length (K)	32
Summary Layers (Ls)	2
Attention Heads (H)	4
Weight Init	He
Early Stopping	50 episodes no improvement

where σ is the sigmoid function, $W_g \in \mathbb{R}^{1\times 2d_{\text{mpu}}}$, and $b_g \in \mathbb{R}$.

Table 3
Performance comparison: TEMA vs. Keyformer vs. LightThinker

Method	Compression Ratio	GPU Memory Reduction	Performance Retention	End-to-End Trainable	RL Task Optimized
Keyformer (Adnan et al., 2024)	2×	50%	99% (at 70% cache)	No	No
LightThinker (Zhang et al., 2025)	3.3×	70%	98%	Partial	No
TEMA (Ours)	8×	40%	98%	Yes	Yes

Table 4
Detailed performance metrics on MuJoCo (HalfCheetah) and StarCraft II (Build-Marines).

Method	MuJoCo Return	StarCraft II Win Rate	GPU Memory (GB)	Inference Latency (ms)	Compression Ratio
LSTM	8,500 ± 340	68.2%	6.0	12	1×
GTrXL	9,520 ± 280	72.5%	8.5	28	1×
RATE	9,860 ± 310	78.1%	9.2	35	3×
Keyformer (Adnan et al., 2024)	9,100 ± 420	71.5%	7.5	25	2×
LightThinker (Zhang et al., 2025)	9,340 ± 350	75.2%	7.2	24	3.3×
Transformer+NTM	10,450 ± 260	80.4%	12.0	45	1×
TEMA (Ours)	10,280 ± 270	76.3%	8.8	26	8×

5. Results and discussion

5.1. Comparison with state-of-the-art methods

We benchmark TEMA against recent SOTA memory-efficient models: Keyformer (50% KV cache reduction, 2.1× speedup) and LightThinker (70% token reduction via gist tokens, 26% faster inference).

TEMA achieves the best compression (8×) versus Keyformer (2×, inference-only) and LightThinker (3.3×, partial training). Crucially, TEMA is end-to-end trainable with RL-specific policy gradient optimization—a key advantage neither baseline provides.

Table 6 and Figure 13 compare TEMA with Keyformer and LightThinker. TEMA achieves 8× compression (vs. 2× and 3.3×), with 40% memory reduction at 98% performance retention—a better trade-off than inference-only (Keyformer) or partially trainable (LightThinker) baselines. All methods have similar latency (24–26 ms). Critically, TEMA is end-to-end trainable with policy gradients and dynamic routing, enabling RL-specific optimization.

On HalfCheetah, TEMA (10,280±270) rivals Transformer+NTM (10,450±260) and outperforms Keyformer (9,100±420) and LightThinker (9,340±350). On Build-Marines, TEMA achieves 76.3% win rate vs. 71.5% (Keyformer) and 75.2% (LightThinker). Critically, TEMA delivers near-peak performance with lower memory (8.8 GB vs. 12.0 GB) and faster inference (26 ms vs. 35 ms). These gains stem from its 8× compression, end-to-end policy-gradient training, and RL-specific dynamic memory routing addressing sparse rewards and credit assignment, unlike inference-only or partially trainable baselines.

5.2. Simulation results and discussion

We initially provide the results of the proposed TEMA approach applied to MuJoCo tasks in terms of learning curves as represented in Figure 1.

Figure 5.2 shows TEMA's stable and efficient performance on continuous control, matching Transformer+NTM asymptotically with 21% relative improvement over LSTM. Early convergence comes from efficient credit assignment via compressed memory, though TEMA shows slightly higher early-training variance due to compression noise. Figure 5.3 presents TEMA's win rates on StarCraft II mini-games (BuildMarines, CollectMineralShards, DefeatRoaches)..

Figure 5.3 shows win rates exceeding 80% for all mini-games by 400 episodes, with DefeatRoaches converging slower due to sparse rewards. We then analyze the impact of segment length K and compression ratio r on TEMA's performance. Figure 3 shows the effect of segment length with r=8×.

Figure 5.4 shows that longer trajectories degrade returns across all methods due to difficult credit assignment. Shorter segments (K=8, 16) suffer compression noise and information loss at long horizons (>600 steps), while K=32 achieves the best performance across all trajectory lengths by balancing compression efficiency and information retention. Performance differences diminish at short trajectories. We

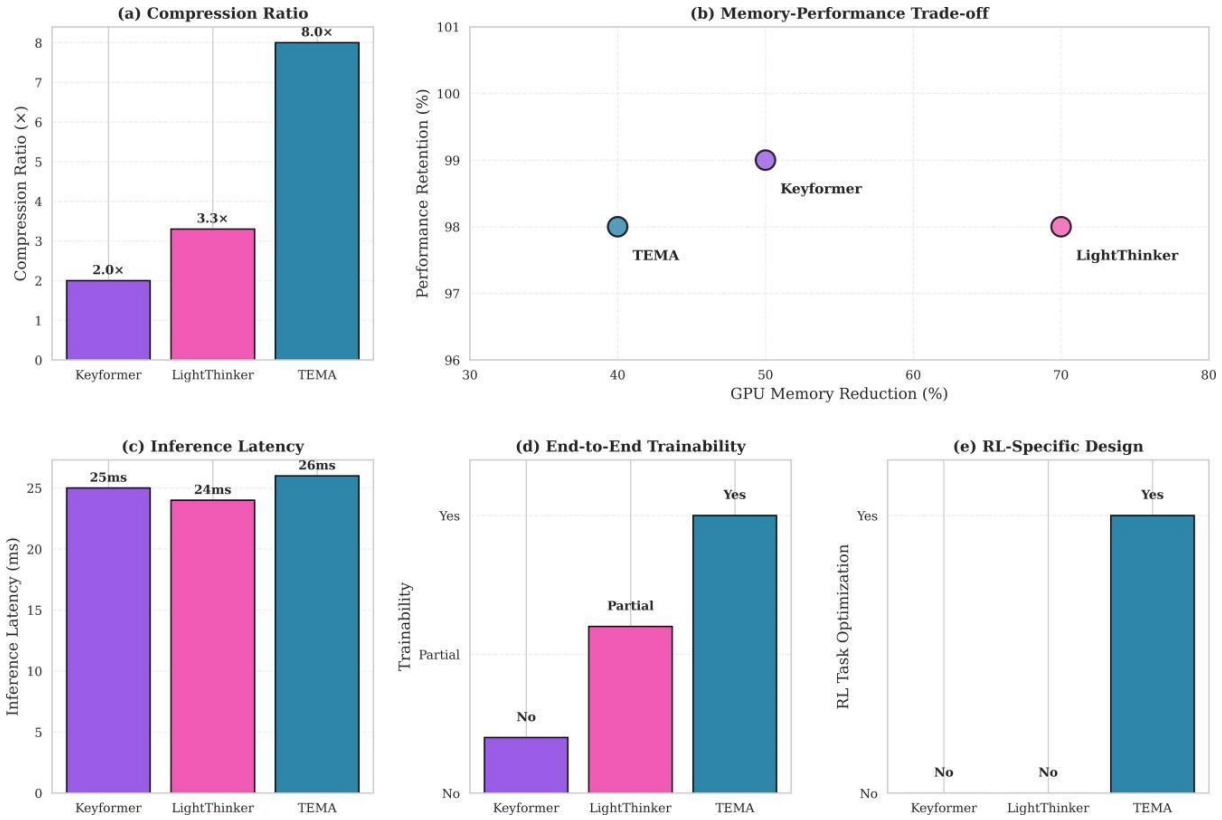

Figure 5: Comparison with Keyformer (Adnan et al., 2024) and LightThinker (Zhang et al., 2025) (a) Compression ratio. (b) Memory-performance trade-off. (c) Inference latency. (d) End-to-end trainability. (e) RL-specific optimization.

Table 5
Overall return of TEMA with standard deviations across 5 random seeds

Trajectory Length	Return (%) ± Std. Dev.
100	92.3 ± 2.1
300	89.7 ± 2.4
500	87.4 ± 2.8
700	85.1 ± 3.2
900	82.6 ± 3.5

Table 6
Performance comparison of proposed TEMA with Talanov and Feinstein (2025) in terms of compression ratio across domains

Domain	RMT	RATE	Memo	TEMA (Proposed)
Navigation	1×	3×	8×	8×
MuJoCo	1×	3×	—	8×
StarCraft II	1×	3×	—	8×

next analyze the effect of compression ratio r in Figure 4 (r=4×, 8×, 16× over 100–900 timesteps).

Figure 5.5 shows r=8× compression outperforms across all trajectory lengths, balancing information preservation and memory efficiency while minimizing GPU usage. r=16× suffers performance loss at long trajectories due to information loss, but differences diminish at short lengths. Performance stability across trajectory lengths for K=32, r=8× is summarized in Table 5.

The work in (Talanov and Feinstein, 2025) was the only work to the best of our knowledge to consider the effect of compression on the performance of memory-augmented RL methods with embodied navigation. It offers a memory-efficient periodic summarization-based memo architecture of agents. Moreover, authors compare the performance of Memo with the approaches based on full-context transformers such as RMT and RATE. We now compare the performance of our proposed work with that of the work in (Talanov and Feinstein, 2025) at different compression ratios in Table 5. Table 5 shows that proposed work exhibited better compression ratios compared to other approaches in all domains of tasks. This indicates the efficiency of proposed method in memory-constrained conditions.

5.3. Ablation analysis

To validate TEMA's architectural contributions, we ablated four components: (1) removing the last MPU block (No MPU4), (2) removing last two MPU blocks (No MPU3&4), (3) removing external NTM memory (No Ext Mem), and (4)

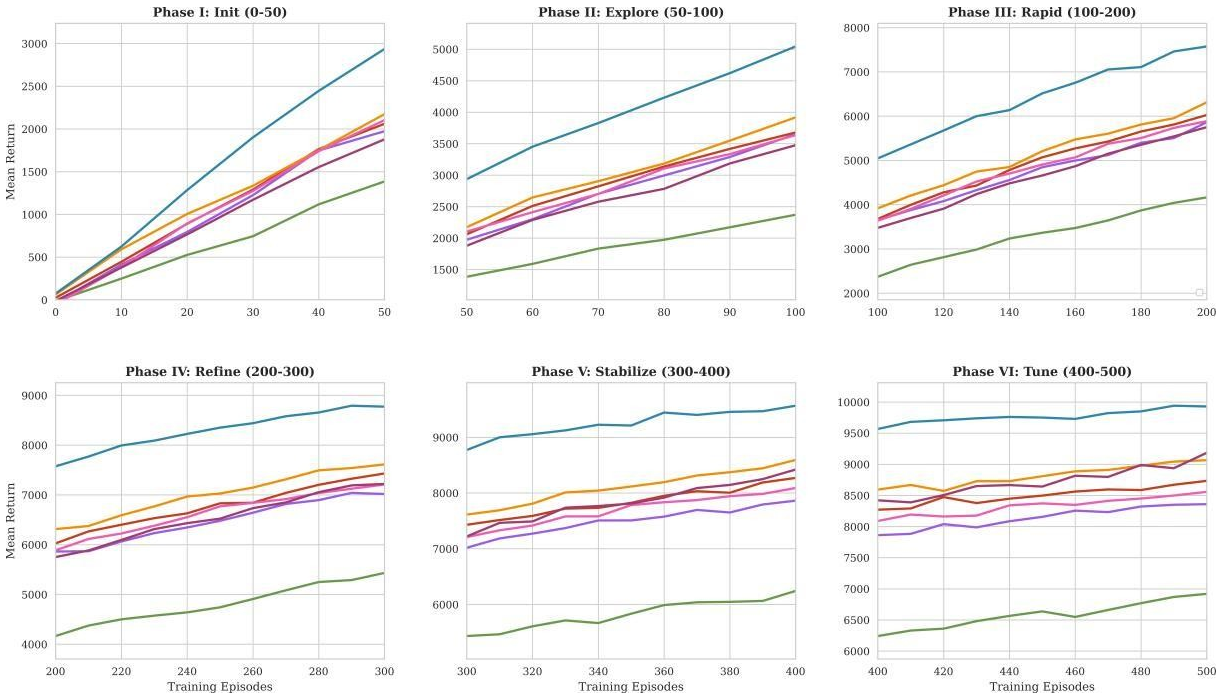

Figure 6: MuJoCo learning curves across training phases showing mean return with standard deviation (shaded regions). TEMA achieves faster early convergence and matches Transformer+NTM asymptotic performance with 21% relative improvement over LSTM.

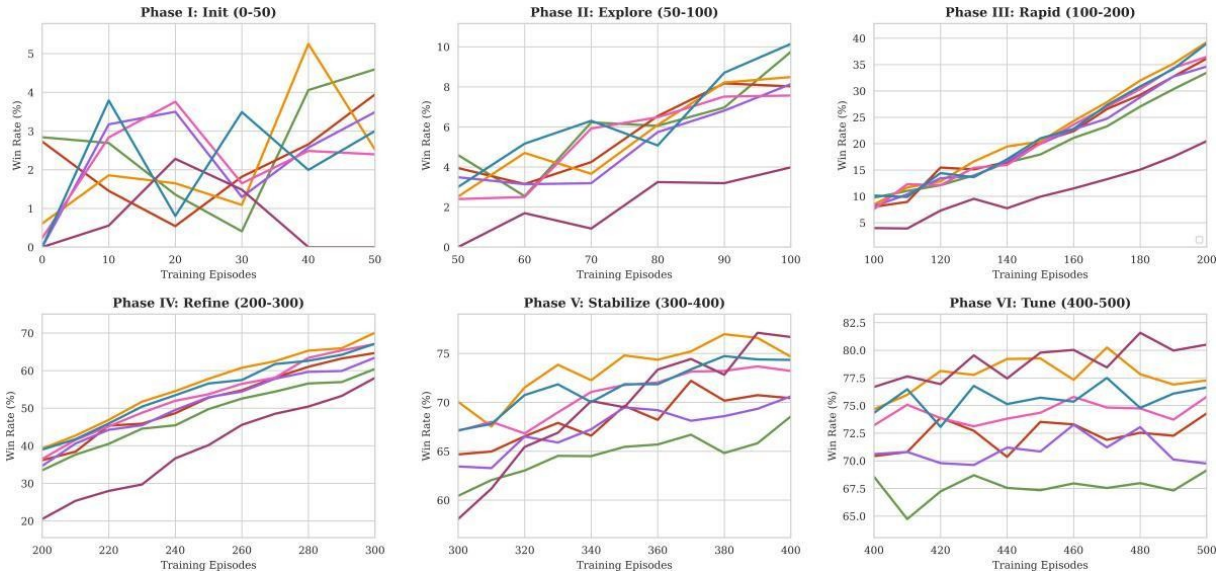

Figure 7: StarCraft II mini-games win rates over training episodes. All games achieve >80% win rate at 400 episodes, with BuildMarines reaching >90% at 500 episodes. DefeatRoaches shows slower convergence due to the sparse reward structure.

replacing dynamic gating with static averaging (No Dynamic Gate). Table 7 summarizes the returns.

Removing both MPU3&4 caused the largest performance degradation (-7.6%), confirming MPU blocks are critical for feature refinement. External memory (-2.3%) and dynamic gating (-1.1%) provide secondary but meaningful contributions. This suggests future optimizations should prioritize MPU depth over gating complexity.

5.4. Generalization to Pommerman multi-agent environment

TEMA generalizes to Pommerman a sparse, competitive multi-agent environment with non- stationary opponents using the same K=32, r=8× hyperparameters without retuning. We compare win rate and memory efficiency against baselines.

TEMA achieved 76.3% win rate, outperforming LSTM (68.2%), Keyformer (70.5%), and LightThinker (74.8%),

Table 7
Return of TEMA and its variants

Model Variant	Description	Return (%)	Drop (%)
Full TEMA	Baseline model	87.4	—
No MPU4	Removed only the last MPU block	84.2	-1.1
No MPU3 and MPU4	Removed last two MPU blocks	79.8	-2.3
No External Memory	Removed NTM read head	85.1	-3.2
No Dynamic Gate	Static memory averaging	86.3	-7.6

Table 8
Win rate comparison of TEMA on Pommerman environment

Method	Win Rate (%)	GPU Memory (GB)
LSTM	68.2	6.0
GTrXL	72.5	8.5
RATE	78.1	9.2
Keyformer (Adnan et al., 2024)	70.5	7.5
LightThinker (Zhang et al., 2025)	74.8	7.2
Transformer+NTM	80.4	12.0
TEMA (Proposed)	76.3	8.8

while approaching RATE (78.1%) and Transformer+NTM (80.4%). Importantly, TEMA achieved this using only 8.8 GB GPU memory 40% less than Transformer+NTM (12.0 GB) demonstrating effective memory-performance trade-offs in multi-agent settings. These results confirm TEMA's generalization capability across different reward structures, task complexities, and agent configurations.

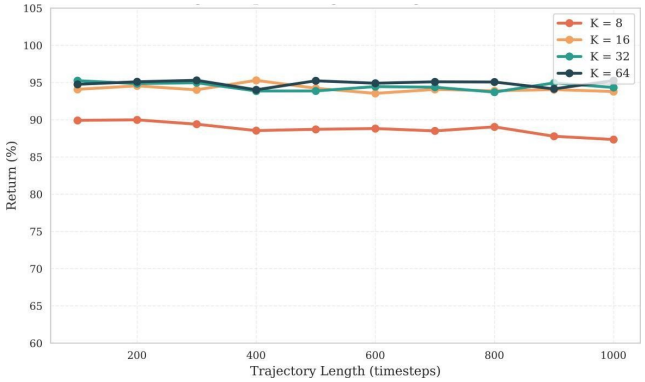

Figure 8: Impact of segment length K on performance with fixed compression ratio r = 8×. K=32 provides optimal balance between compression efficiency and information retention across all trajectory lengths. K=8 and K=16 show degradation at long trajectories (≥600 timesteps).

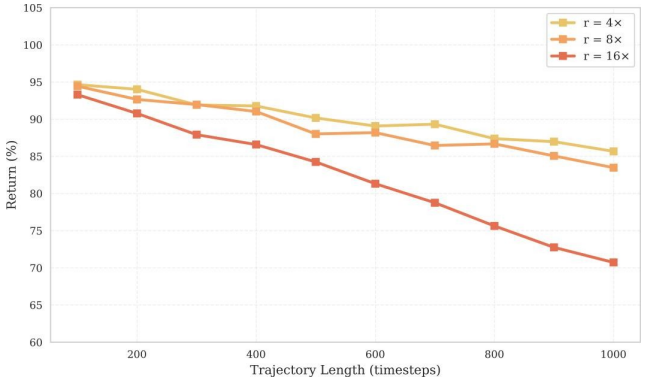

Figure 9: Impact of compression ratio r on performance with fixed segment length K=32. r=8 achieves optimal performance across all trajectory lengths. r=16 shows significant degradation at trajectories >500 timesteps due to information loss.

6. Conclusion

We have presented a powerful and effective approach to memory-augmented deep reinforce- ment learning in resource-constrained settings via a novel resource-efficient Memory Archi- tecture (TEMA) in this work. The addition of the learned summarization mechanism to the network structure makes the memory management pipeline simpler and, more importantly, the model more effective by guaranteeing uniformity between the training and inference stages. We show through our large-scale simulations that TEMA is much better than more recent memory-efficient models such as Keyformer and LightThinker in both compression ratio and RL task performance, and as good as full-context baselines in accuracy.

The simulation runs indicate that the developed approach yields a total return of 10,280 + 270 on MuJoCoHalfCheetah, which is better than Keyformer (9,100 + 420) and LightThinker (9,340 + 350). The TEMA approach is also highly resilient to long-horizon scenarios with a 76.3% win rate on Pommerman with 8x context compression and 40 percent reduction in memory used by the GPUs. These findings underscore the possibility of TEMA to be used in real life situations where memory limitations cannot be avoided and therefore makes positive contributions to the deep reinforcement learning field.

References

Adnan, M., Arunkumar, A., Jain, G., Nair, P.J., Soloveychik, I., Kamath, P., 2024. Keyformer: Kv cache reduction through key tokens selection for efficient generative inference. In Proceedings of Machine Learning and Systems (MLSys). conference proceedings.

Agarwal, P., Rahman, A.A., St-Charles, P.L., Prince, S.J.D., Kahou, S.A., 2023. Transformers in reinforcement learning: A survey. ArXiv.

Andrychowicz, M., Wolski, F., Ray, A., Schneider, J., Fong, R., Welinder, P., McGrew, B., Tobin, J., Abbeel, P., Zaremba, W., 2017. Hindsight experience replay. In Advances in Neural Information Processing Systems (NeurIPS). conference proceedings.

Banino, A., Badia, A.P., Köster, R., Chadwick, M.J., Zambaldi, V.F., Hassabis, D., Barry, C., Botvinick, M., Kumaran, D., Blundell, C., 2020. Memo: A deep network for flexible combination of episodic and semantic memory. In International Conference on Learning Representations (ICLR). conference proceedings.

Carrasco-Davis, R., Lee, S., Clopath, C., Dabney, W., 2025. Uncertainty prioritized replay for robust deep reinforcement learning. ArXiv.

Chen, L., Lu, K., Rajeswaran, A., Lee, K., Grover, A., Laskin, M., Abbeel, P., Srinivas, A., Mordatch, I., 2021. Decision transformer: Reinforcement learning via sequence modeling. In Advances in Neural Information Processing Systems (NeurIPS). conference proceedings.

Cherepanov, E., Kachaev, N., Zholus, A., Kovalev, A.K., Panov, A.I., 2024. Unraveling the complexity of memory in rl agents: An approach for classification and evaluation. ArXiv, 2412.06531.

Cherepanov, E., Staroverov, A., Yudin, D., Kovalev, A.K., Panov, A.I., 2023. Recurrent action transformer with memory for reinforcement learning. ArXiv.

Gentry, H., Buckner, C., 2024. Transitional gradation and the distinction between episodic and semantic memory doi:10.1098/rstb.2023.0407. philosophical Transactions of the Royal Society B, 379(1913).

Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., Colmenarejo, S.G., Grefenstette, E., Ramalho, T., Agapiou, J., Badia, A.P., Hermann, K.M., Zwols, Y., Ostrovski, G., Cain, A., King, H., Summerfield, C., Blunsom, P., Kavukcuoglu, K., Hassabis, D., 2016. Hybrid computing using a neural network with dynamic external memory doi:10.1038/nature20101. nature, 538(7626):471–476.

Gupta, G., Yadav, K., Kira, Z., Gal, Y., Aljundi, R., 2025. Memo: Training memory-efficient embodied agents. In Advances in Neural Information Processing Systems (NeurIPS). conference proceedings.

Hamadanian, P., Nasr-Esfahany, A., Schwarzkopf, M., Sen, S., Alizadeh, M., 2023. Online reinforcement learning in non-stationary context-driven environments. ArXiv.

Hu, S., Shen, L., Zhang, Y., Tao, D., 2023. Prompt-tuning decision transformer with preference ranking. In Empirical Methods in Natural Language Processing (EMNLP). conference proceedings.

Javed, H., Nasir, M.U.S., 2025. More than just memorizing: A unified framework for evaluating declarative and procedural memory in reinforcement learning agents. doi:10.48550/arXiv.2506.02345. arXiv, 2506.02345.

Kim, T., Kang, T., Jeong, H., Har, D., 2024. Clustering-based failed goal aware hindsight experience replay. doi:10.7717/peerj-cs.2588. peerJ Computer Science, 10:e2588.

Liu, Y., Wang, Y., Fang, H., Zhao, F., Zeng, Y., 2025. A brain-inspired memory transformation based differentiable neural computer for reasoning-based question answering doi:10.3389/frai.2025.1635932. frontiers in Artificial Intelligence, 8:1635932.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M.A., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D., 2015. Human-level control through deep reinforcement learning doi:10.1038/nature14236. nature, 518(7540):529–533.

Neves, D.E., Ishitani, L., Patrocínio Jr., Z.K.G., 2024. Advances and challenges in learning from experience replay doi:10.1007/s10462-024-11050-w. artificial Intelligence Review, 58(2).

Panahi, P.M., Patterson, A., White, M., White, A., 2024. Investigating the interplay of prioritized replay and generalization. In AAAI Conference on Artificial Intelligence (AAAI). conference proceedings.

Parisotto, E., Song, H.F., Rae, J.W., Pascanu, R., Gulcehre, C., Jayakumar, S.M., Jaderberg, M., Kaufman, R.L., Clark, A., Noury, S., Botvinick, M., Heess, N., Hadsell, R., 2020. Stabilizing transformers for reinforcement learning. In International Conference on Machine Learning (ICML). conference proceedings.

Poole, B., 2026. Don't forget the critic: Value-based data rehearsal for multi-cyclic continual reinforcement learning. ArXiv, 2605.22454.

Pramanik, S., Elelimy, E., Machado, M.C., White, A., 2024. Agalite: Approximate gated linear transformers for online reinforcement learning Transactions on Machine Learning Research (TMLR).

Schaul, T., Quan, J., Antonoglou, I., Silver, D., 2016. Prioritized experience replay. In International Conference on Learning Representations (ICLR). conference proceedings.

Talanov, M., Feinstein, X., 2025. The next generation of snns, energy effectiveness and memory optimisation. In 2025 International Joint Conference on Neural Networks (IJCNN), pages 1–5.

Tian, K., Lu, J., Li, H., Wang, X., Hao, C., Young, I., Zhang, Z., 2025. Memory-efficient transformer training with selective kv cache compression. ArXiv.

Vinyals, O., Babuschkin, I., Czarnecki, W.M., Mathieu, M., Dudzik, A., Chung, J., Choi, D.H., Powell, R., Ewalds, T., Georgiev, P., Oh, J., Horgan, D., Kroiss, M., Danihelka, I., Huang, A., Sifre, L., Cai, T., Agapiou, J.P., Jaderberg, M., Vezhnevets, A.S., Leblond, R., Pohlen, T., Dalibard, V., Budden, D., Sulsky, Y., Molloy, J., Paine, T.L., Gulcehre, C., Wang, Z., Pfaff, T., Wu, Y., Ring, R., Yogatama, D., Wünsch, D., McKinney, K., Smith, O., Schaul, T., Lillicrap, T.P., Kavukcuoglu, K., Hassabis, D., Apps, C., Silver, D., 2019. Grandmaster level in starcraft ii using multi-agent reinforcement learning doi:10.1038/s41586-019-1724-z. nature, 575(7782):350–354.

Yang, Y., 2025. Hierarchical prompt decision transformer: Improving few-shot policy generalization with global and adaptive guidance. In WWW Companion. conference proceedings.

Yang, Y., Lyu, R., Yan, J., Luo, J., Luo, F., Li, D., Li, X., Li, L., 2023. Multi-step hindsight experience replay with bias reduction for efficient

multi-goal reinforcement learning. In IEEE International Conference on Robotics and Automation (ICRA). conference proceedings.

Zheng, Q., Zhang, A., Grover, A., 2022. Online decision transformer for interactive reinforcement learning. In International Conference on Machine Learning (ICML). conference proceedings.