From Tokens to Latent States: Leveraging Pre-trained Language Models for Improving Partially Observable Reinforcement Learning

Meiju Li[1*], **Ruixiang Sun**[1*], **Xin Li**[1†], **Mingzhong Wang**[2]

[1] Beijing Institute of Technology
[2] University of the Sunshine Coast
{meijuli, 3120240918, xinli}@bit.edu.cn, mwang@usc.edu.au

Abstract

Partially observable Markov decision processes (POMDPs) present significant challenges for reinforcement learning, as agents must learn optimal policies while maintaining belief states over unobserved environment states based on partial observations. We observe a compelling analogy: large language models (LLMs) autoregressively generate token probability distributions based on preceding context, mirroring how belief states are maintained and updated in POMDPs. This insight motivates leveraging the rich prior knowledge embedded in pre-trained LLMs for latent states estimation from observation-action histories. However, two critical challenges emerge: on the one hand, modality misalignment prevents LLMs from directly encoding visual observations and discrete actions; on the other hand, semantic misalignment exists between observation-action sequences and token sequences. To address these challenges, we introduce a novel framework ELSLLM that employs a Johnson-Lindenstrauss projection (JLP) module to transform input dimensions while preserving state similarity with theoretical guarantees, and utilizes modern Hopfield networks (MHN) to store all word embeddings from pre-trained LLMs as a knowledge repository. Through retrieval and querying mechanisms, ELSLLM achieves token-level knowledge alignment without requiring fine-tuning of the pre-trained LLMs. Extensive experiments on partially observable environments demonstrate that ELSLLM achieves state-of-the-art performance, significantly outperforming baseline methods with and without LSTM memory mechanisms. Our work opens new avenues for integrating pre-trained LLMs with reinforcement learning in partially observable settings.

Introduction

Environments and tasks under the setting of partially observable Markov decision processes (POMDP) (Åström 1965; Kaelbling, Littman, and Cassandra 1998) is a widely studied problem in reinforcement learning (RL), reflecting many real-world scenarios where agents must make decisions with incomplete or noisy sensory information. The core difficulty lies in the dual task of inferring the environment's latent state from a history of partial observations while simultaneously learning a policy that maximizes long-term rewards based on this inference. Traditional approaches relied on Recurrent Neural Networks (RNNs) (Zaremba, Sutskever, and Vinyals 2014; Graves and Graves 2012) to compress history observations into compact memory representation (Hausknecht and Stone 2015), or have focused on constructing explicit world models to approximate the belief states, a probability distribution over the latent states (Hafner et al. 2019b,a, 2020). However, these methods, particularly those based on RNNs, often struggle with long-term dependencies and can exhibit limited generalization and sample efficiency(Igl et al. 2018).

To overcome these limitations of RNNs, recent research has naturally turned to the Transformer architecture(Vaswani et al. 2017), which has achieved revolutionary success in sequence modeling tasks (Devlin et al. 2019). With its multi-head self-attention mechanism and efficient parallel training capabilities, the Transformer offers a new paradigm for addressing long-range dependencies (Dai et al. 2019). Consequently, many methods that previously relied on RNNs for constructing memory or world models have begun to transition towards Transformer-based solutions(Esslinger, Platt, and Amato 2022; Chen et al. 2022). Despite their empirical advantages, the complex network structures and deeper model hierarchies of Transformers introduce new challenges(Han, Mao, and Dally 2015). Particularly in the online RL setting, where high-quality interaction samples are inherently sparse and difficult to obtain, Transformer models often require extended training periods and exhibit lower sample efficiency, thereby limiting their application in data-constrained scenarios (Janner, Li, and Levine 2021; Zheng, Zhang, and Grover 2022).

Recently, pre-trained Large Language Models (LLMs) have shown impressive zero-shot and few-shot generalization in natural language processing tasks (Brown et al. 2020). Their strengths in text summarization, reasoning, and knowledge integration (Wei et al. 2022) suggest new possibilities for constructing efficient and generalizable memory or world models for RL. However, directly applying LLMs to RL tasks presents several challenges. First, the success of LLMs relies on pre-training with large-scale text corpora, while most RL tasks lack sufficient expert data, and available datasets may not reflect optimal policies (Zheng, Zhang, and Grover 2022; Ding et al. 2024). This makes it impractical to pre-train or fine-tune large models for specific POMDP

[*]These authors contributed equally.
[†]Corresponding author.

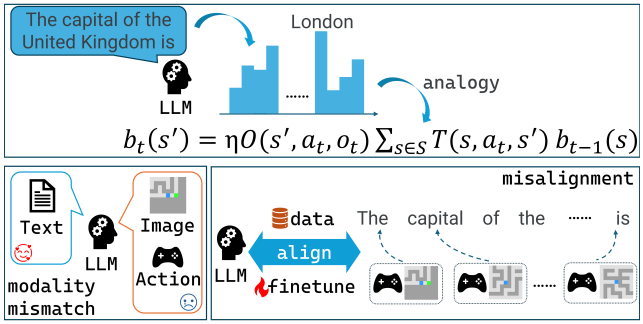

Figure 1: We establish an analogy between LLM token probability distributions and POMDP belief states, and between autoregressive token generation and belief state updates. However, pre-trained LLMs cannot directly process visual observations and discrete actions, creating modality mismatch and misalignment that we try to address.

tasks. Second, there is a significant modality gap: LLMs are designed for textual input, whereas RL environments typically involve image observations and actions, complicating modality alignment (Paischer et al. 2022b, 2023). Therefore, a promising direction is to leverage the prior knowledge and reasoning capabilities of existing pre-trained LLMs as plug-and-play inference engines for decision-making, with minimal or no fine-tuning (Du et al. 2023; Yuan et al. 2023).

In this paper, we introduce a novel framework for **E**stimating **L**atent **S**tates via pre-trained **L**arge **L**anguage **M**odel (**ELSLLM**) to address RL tasks under the POMDP setting. Our approach treats the LLM as a knowledge base that can process a sequence of observation-action history to estimate the current "belief state", which is then consumed by a downstream actor-critic module (Schulman et al. 2017) for decision-making. To bridge the modality gap between the observation-action space and the LLM's text space, we employ the Johnson-Lindenstrauss (JL) lemma(Johnson, Lindenstrauss et al. 1984; Dasgupta and Gupta 2003) to project observation-action pairs into the LLM's word embedding space, a method that preserves metric similarity with high probability. To efficiently query the vast knowledge embedded in the LLM's vocabulary, we utilize a modern Hopfield network (Ramsauer et al. 2020; Krotov and Hopfield 2016) as an associative memory. Finally, inspired by learnable positional embedding in Transformers (Vaswani et al. 2017), we integrate a perceptual encoder, implemented via CNN (LeCun et al. 1989) or ViT (Dosovitskiy 2020), to capture local temporal patterns from the observation history, thereby providing a structured input to our LLM-based belief module. Through extensive experiments, we validate the effectiveness of ELSLLM across multiple challenging POMDP environments, demonstrating superior performance compared to existing baseline methods.

Our main contributions can be summarized as follows:

- We propose a novel framework ELSLLM that leverages pre-trained LLMs for latent states estimation in POMDPs without requiring fine-tuning, effectively addressing the fundamental challenge of limited expert data availability in reinforcement learning environments.
- We introduce a JLP module based on the JL lemma that maps observation-action pairs, which constitute sufficient statistics for the belief state update equation, into the word embedding space while guaranteeing metric similarity. This achieves modality alignment and preserves all information required for belief state updates.
- We propose a token-level semantic alignment method analogous to sampling from a belief state, instantiated using a modern Hopfield network to achieve a balance between efficiency and accuracy.
- Through comprehensive experiments across multiple POMDP environments, we demonstrate that our approach achieves state-of-the-art performance with significant improvements over baseline methods, while exhibiting superior sample efficiency and enhanced generalization capabilities.

Related Work

Early solutions to DQN's limitations in POMDP introduced recurrent neural networks to provide memory and improve latent state estimation. The Deep Recurrent Q-Network (DRQN) (Hausknecht and Stone 2015) utilized LSTMs to process observation histories. This framework was subsequently enhanced by Action-based DRQN (ADRQN) (Zhu et al. 2017), which incorporates action sequences, and Deep Distributed Recurrent Q-Network (DDRQN) (Foerster et al. 2016), which extends DRQN to multi-agent scenarios. Additionally, bidirectional RNNs have been applied to further improve sequential decision-making (Chen, Guo, and Gao 2021).

Transformers and self-attention mechanisms have become increasingly prominent in reinforcement learning. The introduction of the Transformer architecture (Vaswani et al. 2017) has inspired new approaches, such as Decision Transformer (Chen et al. 2021), which reframes reinforcement learning as a sequence modeling problem. Deep Transformer Q-Network (DTQN) (Esslinger, Platt, and Amato 2022) leverages Transformer encoders to address POMDP, while Deep Attention Recurrent Q-Network (DARQN) (Sorokin et al. 2015) enhances LSTM-based history representations with attention mechanisms. Attention-Augmented Agents (AAA) (Mott et al. 2019) employ visual attention to create more interpretable reinforcement learning algorithms.

To address the challenges of training Transformer modules from scratch as memory mechanisms, researchers have increasingly leveraged pre-trained large language models with strong generalization capabilities for reinforcement learning tasks, achieving promising results. In text-based environments, methods like ELLM (Du et al. 2023) and GATA (Adhikari et al. 2020) directly utilize LLM reasoning for policy guidance and state estimation. However, non-textual tasks require semantic alignment. HELM (Paischer et al. 2022a) and SHELM (Paischer et al. 2023) employ Hopfield networks and CLIP to align visual observations with LLM embeddings, while Time-LLM (Jin et al. 2023) uses linear mappings to adapt LLMs for time-series processing.

Preliminaries

A partially observable Markov decision process (POMDP) is a framework for modeling decision-making in environments where the agent cannot directly observe the true state. Formally, a POMDP is defined by the tuple $(S, A, O, T, M, R, \gamma)$, where S is the set of latent states, A is the set of actions, O is the set of observations, $T(s'|s, a)$ is the state transition probability, $M(o|s, z)$ is the observation function with noise z, $R(s', a, s)$ is the reward function, and $\gamma \in (0, 1]$ is the discount factor.

At each time step, the agent selects an action a_t based on its history of observations and actions, receives a new observation o_t, and obtains a reward r_t. Due to partial observability, the agent maintains a belief state b_t, a probability distribution over possible latent states, which is recursively updated using the history of actions and observations:

$$b_t(s') = \eta \cdot O(s', a_t, o_t) \sum_{s \in S} T(s, a_t, s') b_{t-1}(s) \quad (1)$$

Here, $b_t(s')$ denotes the belief in state s' at time t, $O(s', a_t, o_t)$ is the probability of observing o_t after taking action a_t in state s', $T(s, a_t, s')$ is the probability of transitioning from state s to s' after taking action a_t, $b_{t-1}(s)$ is the belief in state s at the previous time step, and η is a normalization factor.

The objective in POMDP is to learn a policy $\pi(a|s_t)$ that maximizes the expected cumulative reward:

$$J(\pi) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t-1}) \right] \quad (2)$$

Efficient belief state estimation and policy optimization are crucial for solving POMDP, especially in complex environments with high-dimensional observations and limited access to expert data.

Methods

The primary challenge in applying reinforcement learning to POMDPs lies in effectively estimating belief states from partial observations. We propose a novel framework that leverages pre-trained LLMs to compress observation-action trajectories into belief state representations without requiring domain-specific fine-tuning. Our approach addresses the fundamental modality and semantic alignment challenges through a theoretically-grounded pipeline that integrates several key components, as illustrated in Figure 3.

Framework Architecture

Our end-to-end framework, as depicted in Figure 2, consists of three principal components working in concert: (1) a Knowledge-Enhanced Belief State Estimation (KEBSE) module that leverages pre-trained LLMs, (2) a perceptual encoder that extracts spatial features from observations, and (3) a policy optimization module based on any online RL algorithm for action selection and learning.

At the core of our approach is the theoretical insight that the sequence of actions and observations constitutes a sufficient statistic for belief state estimation in POMDPs.

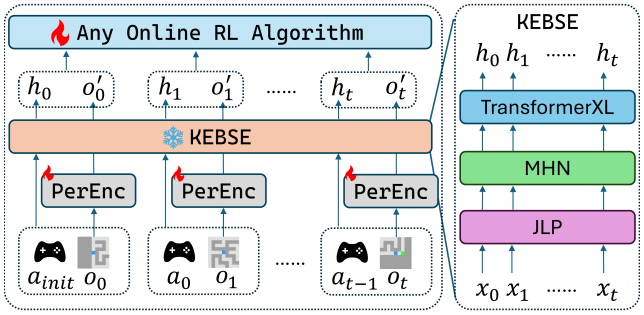

Figure 2: Complete ELSLLM pipeline, including the fully frozen KEBSE module, trainable downstream online reinforcement learning algorithm, and perceptual encoder.

The KEBSE module leverages this principle by encoding the trajectory $(a_{init}, o_0, a_0, o_1, \ldots, a_{t-1}, o_t)$ using a pre-trained LLM to produce a sequence of latent representations $(h_0, h_1, \ldots, h_t)$, which function as belief state estimates. This process effectively mirrors the recursive belief update in Equation 1 but within the LLM's representational space. The primary challenge in this approach lies in bridging the modality gap between non-textual observations/actions and the LLM's natural language input space, which we address through our theoretically-grounded projection and alignment mechanisms. a_{init} is initialized as an all-zero vector, is distinct from the discrete actions a_t that are one-hot encoded, and denotes the start of the sequence.

Modality Bridging via Johnson-Lindenstrauss Projection

To bridge the dimensional mismatch between observation-action feature vectors and the LLM's embedding space, we implement a Johnson-Lindenstrauss Projection Layer (JLP), as illustrated in Figure 3. According to Equation 1, both past observations and actions are essential for estimating the belief state, but prior works have largely ignored the role of actions (Esslinger, Platt, and Amato 2022; Paischer et al.

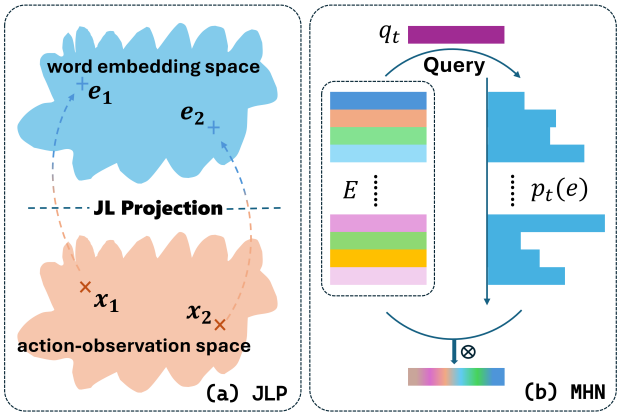

Figure 3: Illustration of Johnson-Lindenstrauss Projection (JLP) and Modern Hopfield Networks (MHN).

2022a, 2023). Treating observations and actions as separate tokens doubles an already long input sequence and increases the encoding burden on pre-trained LLMs. Intuitively, concatenating each observation with its corresponding action into a single token before feeding it to the model is preferable. Our JLP maps concatenated observation-action pairs into the LLM's word embedding space while preserving metric relationships with theoretical guarantees:

$$q_t = JLP(x_t), \quad x_t = concat(a_{t-1}, o_t) \tag{3}$$

This projection is based on the Johnson-Lindenstrauss lemma (Johnson, Lindenstrauss et al. 1984; Dasgupta and Gupta 2003), which ensures that for any pair of vectors $x_t = \text{concat}(a_{t-1}, o_t)$ and $x_{t'} = \text{concat}(a_{t'-1}, o_{t'})$ with difference $d = x_t - x_{t'}$, the following holds with probability at least $1 - \delta$:

$$\left| \frac{\|Pd\|_2^2}{\|d\|_2^2} - 1 \right| \leq \epsilon, 0 < \epsilon < 1$$

where $P \in \mathbb{R}^{d_{emb} \times (d_a + d_o)}$ is a random projection matrix with entries independently sampled from a Gaussian distribution $\mathcal{N}(0, \frac{d_a+d_o}{d_{emb}})$. $\| \cdot \|_2$ denotes the L_2 norm, and

$$\delta = 2 \exp\left(-\frac{d_{emb}(\epsilon^2/2 - \epsilon^3/3)}{2} \right).$$

Token-level Semantic Alignment

While the JLP bridges the dimensional gap, further semantic alignment is necessary to ensure that projected vectors correspond to meaningful representations within the LLM's knowledge base. One effective approach is to use an attention mechanism to compute similarities and aggregate word vectors(Jin et al. 2023). Given the belief state $b_t(s)$ at time t, we typically obtain an estimate $\hat{s}_t$ by

$$\hat{s}_t = \begin{cases} \arg\max_s b_t(s), & \text{state space is discrete,} \\ \mathbb{E}[s_t] = \int s\, b_t(s)\, \mathrm{d}s, & \text{state space is continuous.} \end{cases}$$

We store all word embeddings of the pre-trained LLMs as $E = (e_0, e_1, \ldots, e_{|V|})$, which serve as a basis for the state space. For the query $q_t = Px_t$, we compute the corresponding vector e_t in the word-embedding space as described below, using a temperature coefficient β to control the shape of the distribution.

$$e_t = \sum_{i=1}^{|V|} p_t(e)_i\, e_i, \text{ where } p_t(e)_i = \frac{\exp\left(\beta\,(Eq_t)_i\right)}{\sum_{j=1}^{|V|} \exp\left(\beta\,(Eq_t)_j\right)}$$

This formulation adapts to different retrieval needs:

$$e_t(\beta) = \begin{cases} \dfrac{1}{|V|} \displaystyle\sum_{i=1}^{|V|} e_i, & \beta = 0, \\ \displaystyle\sum_{i=1}^{|V|} \frac{\exp\left(\beta\,(Eq_t)_i\right)}{\sum_j \exp\left(\beta\,(Eq_t)_j\right)}\, e_i, & 0 < \beta < \infty, \\ e_{k^*}, \quad k^* = \arg\max_i (Eq_t)_i, & \beta \to \infty. \end{cases}$$

The computation of Eq_t is the most important. We use Modern Hopfield Networks(Ramsauer et al. 2020) in Figure 3 as an example to balance time complexity and retrieval accuracy; other faster retrieval methods can be used as substitutes or optimizations.

Latent States Estimation and Policy Learning

With the aligned token embeddings e_t, we leverage the pre-trained LLMs to process the sequence $(e_0, e_1, \ldots, e_t)$, obtaining corresponding belief state representations $(h_0, h_1, \ldots, h_t)$. These belief states are then concatenated with spatial features extracted by the perceptual encoder from the current observation, forming comprehensive state representations for the policy optimization module.

The PPO-based actor-critic networks consume these representations to learn value functions and policy distributions, enabling effective decision-making in partially observable environments. Crucially, our approach requires no expert data for fine-tuning the LLM, as the belief state estimation function utilizes the frozen pre-trained model's capabilities, resulting in superior sample efficiency and generalization.

This integrated framework addresses the fundamental challenges of POMDP reinforcement learning by combining the rich prior knowledge of pre-trained LLMs with theoretically-grounded projection and alignment mechanisms, enabling effective latent states estimation and policy learning in complex partially observable environments.

Experiments

We conduct comprehensive experiments across a diverse set of partially observable environments to validate the effectiveness of our proposed framework. The empirical substantiate three key findings:

- Our method consistently outperforms both memory-free and memory-based baselines, including those using LSTM, Transformer, or pre-trained models as memory mechanisms, demonstrating superior sample efficiency and asymptotic performance;
- The prior knowledge from pre-trained LLMs provides substantial benefits for processing observation-action sequence histories, leading to more informed decision-making under partial observability;
- The ELSLLM components facilitate more efficient modality alignment and token-level semantic alignment than conventional learnable approaches.

We compare against four baseline approaches: standard PPO (Schulman et al. 2017) without memory mechanisms (PPO), an extension of PPO augmented with LSTM-based memory module (LSTM-PPO) (Cobbe et al. 2020), DTQN (Esslinger, Platt, and Amato 2022), which trains Transformers from scratch as a memory mechanism, and HELM (Paischer et al. 2022a), which uses a frozen Hopfield network together with pre-trained models to compress historical observation sequences. Our proposed method, ELSLLM, incorporates a frozen pre-trained TransformerXL (Dai et al. 2019)

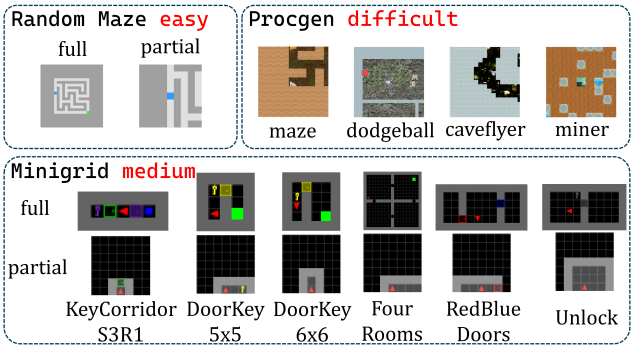

Figure 4: Overview of experimental environments and tasks. Red text indicates the difficulty level of each environment. In RandomMaze and Minigrid, "full" denotes tasks with complete observability, while "partial" denotes tasks with partial observability. We exclusively use the "partial" visual observations in our subsequent experiments. The Procgen environment is inherently partially observable.

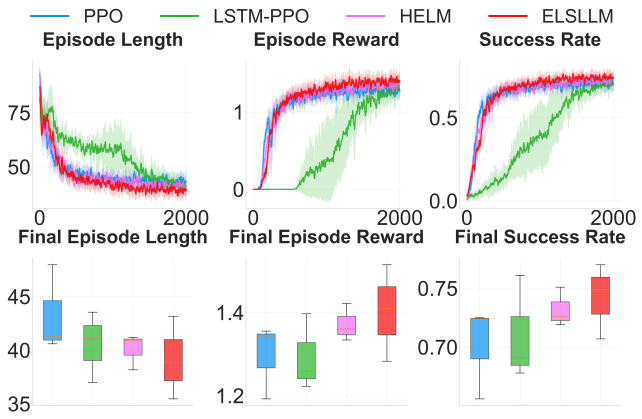

Figure 5: Performance analysis on the RandomMaze environment. The top row shows training curves for episode reward, episode length, and success rate over training epochs. The bottom row summarizes the average performance metrics computed over the final 20,480 environment interaction steps. Episode reward refers to the average cumulative reward per episode, episode length denotes the average number of steps per episode, and success rate measures the average rate at which the agent successfully completes the navigation task. To prevent negative values from distorting visual comparisons, we apply an exponential transformation to episode rewards, $R' = \exp(R)$. This transformation is applied only for the RandomMaze.

as the memory module and employs PPO for policy optimization. Beyond architectural differences in memory design, ELSLLM further distinguishes itself through the way it models historical context: it integrates both past observations and past actions to form a belief-state-like representation. This encoding strategy aligns with the Bayesian update formulation used in POMDPs for belief state estimation, enabling ELSLLM to more effectively infer the underlying latent state. We evaluate all methods on multiple tasks across three increasingly challenging environments (Figure 4).

RandomMaze RandomMaze (Zuo 2018) is a partially observable maze navigation task with procedurally generated mazes. Each episode samples a maze size uniformly from $\{5,...,25\}$. The agent receives an egocentric 9×9 RGB observation and selects one of four discrete actions (up, down, left, right). Episodes last up to 100 timesteps. Illegal moves (out of bounds or into walls) give reward -1 and terminate the episode; legal moves that do not reach the goal yield -0.01; reaching the goal gives $+1$ and ends the episode.

Figure 5 demonstrates the training dynamics and final performance of all three methods in the RandomMaze environment. Our proposed method, ELSLLM, converges at a rate comparable to memory-free PPO, which exhibits the fastest convergence. However, ELSLLM significantly outperforms PPO in terms of final average cumulative reward and success rate, while also achieving shorter average episode lengths, indicating more efficient and effective navigation behavior. In contrast, LSTM-PPO demonstrates the slowest convergence rate, likely due to the difficulty of fitting observation sequences with LSTM architectures in high-dimensional input spaces. This results in suboptimal learning dynamics under the limited training budget, leading to performance inferior to both PPO and ELSLLM.

The superior performance of ELSLLM can be attributed to its use of pre-trained large language models, which provide strong prior knowledge for encoding observation-action histories. By generating richer latent representations aligned with belief-state estimation in POMDPs, ELSLLM facilitates better temporal abstraction and improves the downstream reinforcement learning process, enabling both faster convergence and higher asymptotic performance.

Minigrid Minigrid (Chevalier-Boisvert et al. 2023) is a goal-oriented 2D gridworld. we evaluate seven representative tasks: DoorKey-5x5(DK5), DoorKey-6x6(DK6), Dynamic-Obstacles-Random-6x6(DOR), FourRooms(FR), KeyCorridorS3R1(KC), RedBlueDoors-6x6(RBD) and Unlock. The names in parentheses are task abbreviations, which we will use in the presentation of experimental results. We use a partial-observability wrapper that limits the agent's view to a 7×7 front-facing window, so the agent cannot see beyond doors. The agent has seven actions (left, right, forward, pickup, drop, toggle, done). Success yields a time-discounted reward $1 - 0.9 \times (\text{step\_count}/\text{max\_steps})$, while failure within the step limit gives zero. Tasks are challenging due to sparse rewards, constrained visibility, and the need for exploration, hazard avoidance, and long-term memory.

Tasks	DK5	DK6	DOR	FR	KC	RBD	Unlock
Mean	0.94	0.31	0.94	0.09	0	0	0.64
Std	0	0.29	0	0.02	0	0	0.3

Table 1: Results for DTQN (hyperparameters kept the same as in the original paper) on seven Minigrid tasks (task names are given as abbreviations)

Figure 6 and Table 1 demonstrates the training dynamics

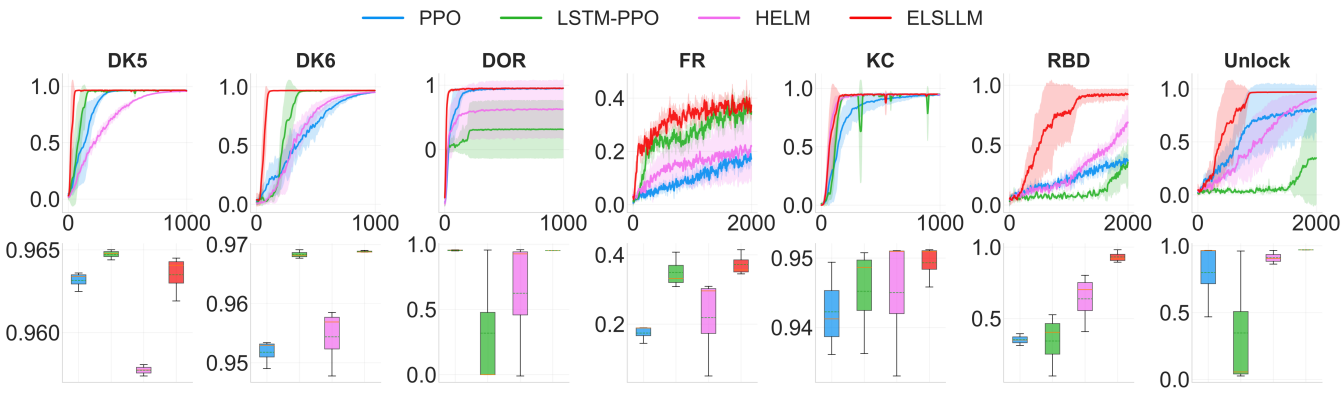

Figure 6: Performance analysis across seven Minigrid tasks. The top row shows training curves of episode reward over training epochs. The bottom row reports the average episode reward over the final 20,480 interaction steps. To account for varying task complexity, FourRooms, RedBlueDoors-6x6, and Unlock tasks are trained for 2,000 epochs, while other Minigrid tasks are trained for 1,000 epochs. Task names are given as abbreviations.

and final performance of all four methods across the Minigrid environment suite. Our proposed ELSLLM method consistently achieves either the best results or on par with the strongest baseline, in terms of both convergence speed and final policy quality. This highlights ELSLLM's superior capability on moderately challenging partially observable tasks. LSTM-PPO exhibits mixed performance compared to memory-free PPO: outperforming on certain tasks while underperforming on others. This inconsistency likely stems from the trade-off between the expressive power of recurrent representations and the increased difficulty of sequence modelling in these chanllenging tasks. Specifically, while LSTMs can, in principle, retain informative temporal features, they also introduce training instability and optimization difficulties, especially under limited training budgets. Given a constraint of a maximum training budget of 2,000 epochs, DTQN and LSTM-PPO do not fully converge on some most challenging tasks, i.e., FourRooms, RedBlueDoors-6x6, and Unlock, leaving its asymptotic performance potential on these tasks inconclusive. One likely reason is that Transformer and LSTM trained from scratch struggle to capture long-term dependencies with limited data, which negatively affects final performance. HELM's experimental results are similar to those reported in the original paper (Paischer et al. 2022a) but remain significantly inferior to ELSLLM, indicating that pre-trained models that compress history using only visual observations are insufficient for capturing long-term dependencies in the Minigrid environment.

Method	PPO	LSTM-PPO	HELM	ELSLLM
MSE	0.0207	0.1686	0.0098	**0.0004**

Table 2: MSE of estimated state-values on sampled Unlock trajectories.

To better assess the impact of our method on belief-state estimation, We sampled trajectories in the Unlock environment using ELSLLM, HELM, LSTM-PPO, and PPO, all trained for 2 million steps. We evaluated each trajectory's value with the value function learned by the corresponding method, and computed the MSE against the true returns computed from the reward function. As shown in Table 2, ELSLLM's estimated V-values are the most accurate, indicating that its belief-state estimates are sufficiently precise.

Procgen Procgen (Cobbe et al. 2020) is a suite of generated environment designed to encourage agents to learn generalizable skills rather than overfitting to specific trajectories. In this paper, we conduct extensive experiments on four representative tasks: maze, miner, caveflyer and dodgeball.

All these tasks demand extended memory capabilities, requiring agents to explore and acquire specific skills under constrained visibility. Furthermore, they feature sparse reward structures that necessitate sophisticated credit assignment mechanisms for effective learning.

Figure 7 demonstrates the training dynamics and final performance of all three methods across the Procgen environment suite. On the Dodgeball and Miner tasks, ELSLLM exhibits clear advantages in both convergence speed and final performance, indicating that demonstrates that frozen pretrained large language models provide superior capabilities for POMDP latent state estimation in long-sequence tasks compared to LSTMs trained from scratch. By leveraging rich prior knowledge through our KEBSE module to align both modality and knowledge, ELSLLM can estimate latent states in partially observable Procgen environments more effectively than learnable CNN or LSTM based architectures, specifically in settings with larger observation and action spaces. ELSLLM shows consistent advantages across RandomMaze, Minigrid, and Procgen Maze tasks, confirming its scalability and robustness across similar task domains. All four methods fail to learn useful information and skills in the Caveflyer task, unable to complete the challenging navigation requirements. A likely reason is that the tasks require too many steps, making it difficult for existing methods to model long-term dependencies and perform effective credit

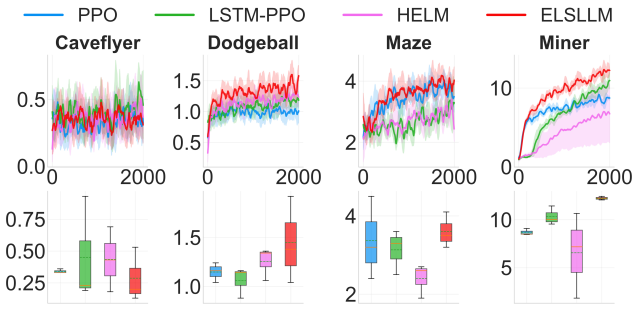

Figure 7: Performance analysis across four Procgen tasks. The top row shows training curves of episode reward over training epochs. The bottom row presents the average episode reward over the final 20,480 interaction steps. All results show the performance of each method over 2,000 training epochs.

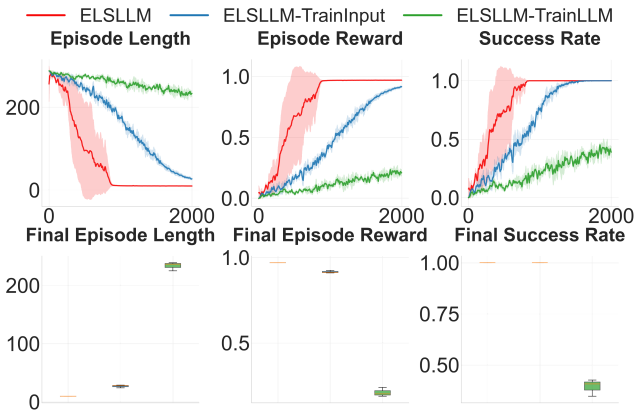

Figure 8: Ablation study results on the Minigrid Unlock task. The top row shows training curves of episode reward over training epochs for three variants: ELSLLM (full model), ELSLLM-TrainInput (replacing JLP and MHN with learnable MLPs), and ELSLLM-TrainLLM (using randomly initialized TransformerXL without pre-trained weights). The bottom row reports the average episode reward over the final 20,480 interaction steps. All methods are trained for 2,000 epochs, demonstrating the individual contributions of our JLP modality alignment and pre-trained LLM knowledge component.

assignment; future work should explore more effective long-term memory mechanisms and reward design.

Ablation Study

We conduct comprehensive ablation studies on the Unlock task in the Minigrid environment to investigate two critical research questions: (1) Does the Johnson-Lindenstrauss Projection (JLP) enable effective modality alignment? (2) Do pre-trained large language models and their embedded prior knowledge contribute to processing historical action-observation sequences?

To address the first question, we introduce the ELSLLM-TrainInput variant, which replaces both the JLP and MHN components in ELSLLM with learnable MLP layers. This variant aims to learn input representations for the pre-trained LLM through environmental exploration rather than leveraging our theoretically-grounded projection mechanism.

To address the second question, we design the ELSLLM-TrainLLM variant, which retains the JLP layer for dimensional mapping but removes the MHN component containing pre-trained LLM prior knowledge. Instead of utilizing pre-trained TransformerXL weights, this variant employs randomly initialized weights and learns representations of historical action-state pairs from scratch through agent exploration.

The ablation study yields several key insights into the contributions of our framework's components. The full ELSLLM model achieves the fastest convergence and highest final performance, confirming the effectiveness of our integrated design. The ELSLLM-TrainInput variant, which removes the JLP component, achieves the second-best performance, indicating that action-observation pair representations are relatively learnable through environmental exploration. Importantly, this variant still benefits from pre-trained LLM knowledge, achieving sub-optimal but competitive performance, which demonstrates the crucial importance of the pre-trained knowledge embedded in large language models.

The ELSLLM-TrainLLM variant, which removes all pre-trained knowledge and learns the language model from scratch, exhibits the poorest performance, on par with LSTM-PPO. This results highlights the inherent difficulty of learning memory mechanisms solely through exploration in POMDPs, regardless of whether a Transformer or LSTM is used. The poor performance of ELSLLM-TrainLLM directly validates the substantial benefit of pre-trained LLMs in latent state inference, emphasizing the core contribution of leveraging frozen LLMs for belief representation in partially observable environments.

Conclusion

We introduce ELSLLM, a novel framework that leverages pre-trained large language models for latent state estimation in partially observable reinforcement learning. Drawing inspiration from the analogy between LLM token generation and POMDP belief state updates, our approach employs Johnson-Lindenstrauss projection and modern Hopfield networks to bridge the modality gap between visual observations and discrete actions with the natural language input space of pre-trained LLMs. Comprehensive experiments across RandomMaze, Minigrid, and Procgen environments demonstrate that ELSLLM achieves superior performance compared to both memory-free and LSTM-based approaches. Our ablation studies confirm the importance of each component, particularly highlighting the significant benefits of leveraging frozen pre-trained LLM knowledge for POMDP tasks. This work opens promising new directions for integrating foundation models with reinforcement learning in partially observable environments.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable feedback. This work was partially supported by the NSFC under Grants 92270125 and 62276024; by the Fundamental Research Funds for the Central Universities, JLU, under Grant 93K172025K01; and by the Fundamental Research Funds for the Central Universities under Grant 2025CX01010.

References

Adhikari, A.; Yuan, X.; Côté, M.-A.; Zelinka, M.; Rondeau, M.-A.; Laroche, R.; Poupart, P.; Tang, J.; Trischler, A.; and Hamilton, W. 2020. Learning dynamic belief graphs to generalize on text-based games. *Advances in Neural Information Processing Systems*, 33: 3045–3057.

Åström, K. J. 1965. Optimal control of Markov processes with incomplete state information I. *Journal of mathematical analysis and applications*, 10: 174–205.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Chen, C.; Wu, Y.-F.; Yoon, J.; and Ahn, S. 2022. Transdreamer: Reinforcement learning with transformer world models. *arXiv preprint arXiv:2202.09481*.

Chen, L.; Lu, K.; Rajeswaran, A.; Lee, K.; Grover, A.; Laskin, M.; Abbeel, P.; Srinivas, A.; and Mordatch, I. 2021. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34: 15084–15097.

Chen, P.; Guo, S.; and Gao, Y. 2021. Deep reinforcement learning with bidirectional recurrent neural networks for dynamic spectrum access. In *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*, 1–5. IEEE.

Chevalier-Boisvert, M.; Dai, B.; Towers, M.; Perez-Vicente, R.; Willems, L.; Lahlou, S.; Pal, S.; Castro, P. S.; and Terry, J. 2023. Minigrid & Miniworld: Modular & Customizable Reinforcement Learning Environments for Goal-Oriented Tasks. In *Advances in Neural Information Processing Systems 36, New Orleans, LA, USA*.

Cobbe, K.; Hesse, C.; Hilton, J.; and Schulman, J. 2020. Leveraging procedural generation to benchmark reinforcement learning. In *International conference on machine learning*, 2048–2056. PMLR.

Dai, Z.; Yang, Z.; Yang, Y.; Carbonell, J.; Le, Q. V.; and Salakhutdinov, R. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

Dasgupta, S.; and Gupta, A. 2003. An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Structures & Algorithms*, 22(1): 60–65.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 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)*, 4171–4186.

Ding, S.; Hu, K.; Zhang, Z.; Ren, K.; Zhang, W.; Yu, J.; Wang, J.; and Shi, Y. 2024. Diffusion-based reinforcement learning via q-weighted variational policy optimization. *arXiv preprint arXiv:2405.16173*.

Dosovitskiy, A. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Du, Y.; Watkins, O.; Wang, Z.; Colas, C.; Darrell, T.; Abbeel, P.; Gupta, A.; and Andreas, J. 2023. Guiding pretraining in reinforcement learning with large language models. In *International Conference on Machine Learning*, 8657–8677. PMLR.

Esslinger, K.; Platt, R.; and Amato, C. 2022. Deep transformer q-networks for partially observable reinforcement learning. *arXiv preprint arXiv:2206.01078*.

Foerster, J. N.; Assael, Y. M.; de Freitas, N.; and Whiteson, S. 2016. Learning to communicate to solve riddles with deep distributed recurrent q-networks. *arXiv preprint arXiv:1602.02672*.

Graves, A.; and Graves, A. 2012. Long short-term memory. *Supervised sequence labelling with recurrent neural networks*, 37–45.

Hafner, D.; Lillicrap, T.; Ba, J.; and Norouzi, M. 2019a. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*.

Hafner, D.; Lillicrap, T.; Fischer, I.; Villegas, R.; Ha, D.; Lee, H.; and Davidson, J. 2019b. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, 2555–2565. PMLR.

Hafner, D.; Lillicrap, T.; Norouzi, M.; and Ba, J. 2020. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*.

Han, S.; Mao, H.; and Dally, W. J. 2015. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*.

Hausknecht, M. J.; and Stone, P. 2015. Deep Recurrent Q-Learning for Partially Observable MDPs. In *AAAI fall symposia*, volume 45, 141.

Igl, M.; Zintgraf, L.; Le, T. A.; Wood, F.; and Whiteson, S. 2018. Deep variational reinforcement learning for POMDPs. In *International conference on machine learning*, 2117–2126. PMLR.

Janner, M.; Li, Q.; and Levine, S. 2021. Offline reinforcement learning as one big sequence modeling problem. *Advances in neural information processing systems*, 34: 1273–1286.

Jin, M.; Wang, S.; Ma, L.; Chu, Z.; Zhang, J. Y.; Shi, X.; Chen, P.-Y.; Liang, Y.; Li, Y.-F.; Pan, S.; et al. 2023. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*.

Johnson, W. B.; Lindenstrauss, J.; et al. 1984. Extensions of Lipschitz mappings into a Hilbert space. *Contemporary mathematics*, 26(189-206): 1.

Kaelbling, L. P.; Littman, M. L.; and Cassandra, A. R. 1998. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2): 99–134.

Krotov, D.; and Hopfield, J. J. 2016. Dense associative memory for pattern recognition. *Advances in neural information processing systems*, 29.

LeCun, Y.; Boser, B.; Denker, J. S.; Henderson, D.; Howard, R. E.; Hubbard, W.; and Jackel, L. D. 1989. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4): 541–551.

Mott, A.; Zoran, D.; Chrzanowski, M.; Wierstra, D.; and Jimenez Rezende, D. 2019. Towards interpretable reinforcement learning using attention augmented agents. *Advances in neural information processing systems*, 32.

Paischer, F.; Adler, T.; Hofmarcher, M.; and Hochreiter, S. 2023. Semantic HELM: an interpretable memory for reinforcement learning. *CoRR, abs/2306.09312*.

Paischer, F.; Adler, T.; Patil, V.; Bitto-Nemling, A.; Holzleitner, M.; Lehner, S.; Eghbal-Zadeh, H.; and Hochreiter, S. 2022a. History compression via language models in reinforcement learning. In *International Conference on Machine Learning*, 17156–17185. PMLR.

Paischer, F.; Adler, T.; Radler, A.; Hofmarcher, M.; and Hochreiter, S. 2022b. Toward semantic history compression for reinforcement learning. In *Second Workshop on Language and Reinforcement Learning*.

Ramsauer, H.; Schäfl, B.; Lehner, J.; Seidl, P.; Widrich, M.; Adler, T.; Gruber, L.; Holzleitner, M.; Pavlović, M.; Sandve, G. K.; et al. 2020. Hopfield networks is all you need. *arXiv preprint arXiv:2008.02217*.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Sorokin, I.; Seleznev, A.; Pavlov, M.; Fedorov, A.; and Ignateva, A. 2015. Deep attention recurrent Q-network. *arXiv preprint arXiv:1512.01693*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Yuan, H.; Zhang, C.; Wang, H.; Xie, F.; Cai, P.; Dong, H.; and Lu, Z. 2023. Skill reinforcement learning and planning for open-world long-horizon tasks. *arXiv preprint arXiv:2303.16563*.

Zaremba, W.; Sutskever, I.; and Vinyals, O. 2014. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.

Zheng, Q.; Zhang, A.; and Grover, A. 2022. Online decision transformer. In *international conference on machine learning*, 27042–27059. PMLR.

Zhu, P.; Li, X.; Poupart, P.; and Miao, G. 2017. On improving deep reinforcement learning for pomdps. *arXiv preprint arXiv:1704.07978*.

Zuo, X. 2018. mazelab: A customizable framework to create maze and gridworld environments. https://github.com/zuoxingdong/mazelab.