Exploring Attention Mechanisms in Transformers for Data-Efficient Model-Based Reinforcement Learning

Daniel De Dios Allegue[1]

Supervisor(s): Dr. Frans Oliehoek[1]**, Dr. Jinke He**[1]

[1]**EEMCS, Delft University of Technology, The Netherlands**

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 22, 2025

Name of the student: Daniel De Dios Allegue
Final project course: CSE3000 Research Project
Thesis committee: Dr. Frans Oliehoek, Dr. Jinke He, Dr. Michael Weinmann

Abstract

A key advancement in model-based Reinforcement Learning (RL) stems from Transformer-based world models, which allow agents to plan effectively by learning an internal representation of the environment. However, causal self-attention in Transformers can be computationally redundant when most relevant information lies in only a few recent steps, making it inefficient for environments with predominantly short-term memory dependencies. This paper investigates integrating alternative attention mechanisms into world models to address these limitations. We embed inductive biases via local attention and Gaussian adaptive attention, aiming to guide the model's focus towards more relevant elements of the observation history. We evaluate these modified architectures in four environments on the Atari 100k benchmark under partially observable conditions. Our results show that, in environments where relevant information is contained within a specific recent window of observations (i.e. a short-term memory dependency), tuning local or Gaussian adaptive attention to that window lets them significantly outperform causal attention within a limited number of interactions. In Pong, the best performing Gaussian attention model raised the mean return from –14.53 to –6.86, representing roughly a 53% improvement over the baseline. The effectiveness of these mechanisms varies with the complexity and dynamism of the influential variables within an environment, highlighting the importance of appropriate prior selection and flexibility. This work highlights that leveraging influence-based principles through inductive biases can lead to more data-efficient attention mechanisms for world models, particularly when agents must learn from limited environment interactions in diverse RL settings.

1 Introduction

Reinforcement Learning (RL) algorithms have become one of the dominant approaches for tackling tasks in complex environments. However, traditional model-based RL methods suffer when handling domains with heterogeneous scenarios [1]. Recent research has focused on enhancing planning algorithms [2] with effective predictive world models to address this issue. Notably, the MuZero-style algorithms [3, 4] have achieved exceptional performance in board games and Atari games by planning on learned latent spaces that emulate the real-world environments. These predictive world models are often limited by their architectural choices and perform inadequately in environments requiring long-term memory and diverse action spaces [5].

UniZero [5] introduced self-attention mechanisms [6, 7] as part of MuZero's backbone to leverage the diverse *backward* memory capabilities of Transformers [8] along with efficient *forward* planning. Despite the state-of-the-art results in multi-task domains with long-term dependencies, the new architecture raises two significant concerns: (1) Whether standard Transformer-based world models, which inherently use causal attention mechanisms, can effectively learn tasks characterized predominantly by short-term dependencies, and (2) Whether the computational complexity of causal attention mechanisms is justified in settings with shorter sequence dependencies.

We first review the limitations of UniZero's architecture in the Atari 100k benchmark. UniZero's Transformer backbone underperforms in tasks where relevant information is confined to a brief observation window (i.e., environments with narrow memory dependencies), potentially due to inefficiencies from applying causal attention across longer observation sequences. The global nature of the Transformer's causal attention mechanism increases exposure to irrelevant or noisy input tokens, potentially causing distraction and reducing decision-making precision. This sensitivity to irrelevant information becomes more pronounced as we increase the size of observation sequences, incrementing the complexity and computational overhead.

To address these limitations, we augment UniZero's Transformer backbone by replacing its causal attention with both local and Gaussian adaptive attention mechanisms [9, 10]. Both approaches aim to reduce computational overhead while preserving the Transformer's ability to model short and long-term dependencies crucial for decision-making tasks. By introducing inductive biases about the domain through the attention masks, the model can learn more effectively within a reduced number of environment interactions.

Our evaluations on selected Atari 100k benchmark environments, under partially observable conditions, demonstrate that these alternative attention mechanisms can improve performance and learning efficiency over causal attention when the embedded inductive biases align well with an environment's temporal dependency structure, particularly in data-limited settings. However, the findings also reveal that the effectiveness varies with complexity and the dynamic nature of each environment. With these results, we seek to answer the following research questions:

- **RQ1:** What is the performance of local and Gaussian attention mechanisms compared to causal attention in environments dominated by short-memory and long-memory dependencies?
- **RQ2:** How does varying the initial local window size (for local attention) and the initial distribution (for Gaussian adaptive attention) impact the performance of these attention mechanisms?

The remainder of this paper is structured as follows. We begin by providing background on key concepts in Section 2 and reviewing related work in Section 3. In Section 4, we analyse the temporal dependencies of two selected environments from an influence perspective. Section 5 details the local and Gaussian adaptive attention mechanisms. Section 6 describes our experimental setup and presents the results of our evaluations. Finally, in Section 7, we conclude with a discussion of our findings, their limitations, and promising directions for future work.

2 Background

Reinforcement Learning [11] is the standard for tackling sequential decision-making problems. Essentially, RL involves an agent interacting with an environment to learn a policy π that maximises the expected cumulative discounted rewards. The main assumption is the Markov Property, which asserts that future states s_{t+1} and rewards r_{t+1} depend solely on the current state and action, expressed by $P(s_{t+1}, r_{t+1}|s_t, a_t)$. However, in real-world environments, including Atari games, the Markovian assumption is often violated due to partial observability. Such problems can be modeled as Partially Observable Markov Decision Processes (POMDPs) [12], defined by the tuple $(S, A, T, R, \Omega, O, \gamma, b_0)$, where S is the set of states, A is the set of actions, $T(s'|s, a)$ denotes the state transition probabilities, $R(s, a)$ specifies the reward function, Ω is the set of observations, $O(o|s', a)$ is the observation probability distribution, $\gamma \in [0, 1)$ is the discount factor and b_0 the initial belief distribution over S. In environments characterised by long-term dependencies, efficient *forward* planning requires focusing on the observation history. However, this history is often truncated to a length H, yielding observation sequences $\tau_{t-H+1:t} = (o_{t-H+1}, a_{t-H+1}, \ldots, o_t, a_t)$, where $o_t \in \Omega$ and $a_t \in A$.

Transformers [6] have emerged as a powerful alternative to Recurrent Neural Networks (RNNs) to process long sequences. By leveraging *(self-) attention* mechanisms, Transformers can model long-term dependencies without relying on recurrence. The core idea of attention is to weigh each element in a sequence according to its relevance to others, mimicking where the model focuses. Formally, attention computes weights using queries Q, keys K and values V by applying a softmax function to the scaled dot product between keys and queries:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

Crucially, this mechanism is permutation invariant; it treats ordered sequences as sets. The properties of these architectures mitigate the limitations of using truncated observation sequences by simultaneously attending to all elements from the past. This offers a compelling opportunity for Transformers to serve as effective world models, as they can accurately capture environmental dynamics. Results show that incorporating Transformer-based architectures into RL frameworks substantially enhances performance in environments with long-term dependencies [8, 1].

MuZero [3] improves upon traditional model-based RL approaches by combining Monte Carlo Tree Search (MCTS) [2] with a RNN learned model of the environment. MuZero does not explicitly receive the environment's rules; instead, it infers them through interactions with the environment. Despite its performance, MuZero has three main limitations: (1) during training, only the initial observation is explicitly used to predict the next state, (2) at inference time, predictions rely exclusively on learned latent states as opposed to the observations in the context, causing the *incomplete context problem*; and (3) capturing long-term dependencies can become difficult as the recursion depth increases [13].

UniZero [5] replaces MuZero's recurrent latent dynamics with a Transformer backbone that uses masked causal attention to model sequences autoregressively [7]. Specifically, at each time step t, the sequence of observation–action pairs $\big[(o_1, a_1), (o_2, a_2), \ldots, (o_t, a_t)\big],\ o_{1:t} \in \Omega$, is fed through stacked Transformer layers with causal (triangular) masks, so that each token may attend only to itself and preceding tokens, never to future ones. The attention mechanisms allow every latent token at step t to attend to preceding tokens $(o_{1:t}, a_{1:t})$, capturing long-range dependencies beyond the fixed horizon in MuZero's RNN. UniZero's learned model is comprised by:

1. Encoder: $z_t = h_\theta(o_t)$ which maps observations to latent states.
2. Dynamics Head: $z_{t+1}, r_t = g_\theta(z_{1:t}, a_{1:t})$ which models the environment's latent dynamics and rewards.
3. Decision Head: $\pi_t, v_t = f_\theta(z_{1:t}, a_{1:t-1})$ which predicts the policy and the value.

Both models are characterised by the joint optimisation of the model and the policy, maintaining a soft target world model $\hat{W} = (\hat{h}_\theta, \hat{g}_\theta, \hat{f}_\theta)$ [14]. The new architectural changes allow the model to use the full observation sequence at training and access complete contexts at inference, mitigating the limitations exposed in MuZero. UniZero exhibits markedly improved memory performance on Atari games that require long-term planning by utilising its global context. However, causal attention often suffers from *attention dilution*, where the attention mechanism spreads its focus so broadly that it can underemphasise the most important tokens [15]. Hence, the new attention leads to slight performance degradation in very short, tightly-coupled games like Pong or Boxing.

3 Related Work

Sparse Attention Mechanisms have been introduced to focus computation on the most relevant tokens while reducing computational overhead. In local sparse attention, each token only attends to a fixed window of its nearest tokens, capturing short-term dependencies very efficiently [16]. Similarly, adaptive attention dynamically learns which tokens deserve higher weight based on importance, adapting to long- or short-range contexts when necessary [9]. Such adaptive mechanisms have been applied in sequential decision-making settings successfully [17]. Striking a balance between modelling temporal dependencies at different scales and keeping computational cost low is crucial for strong performance across tasks with both short- and long-horizon requirements.

Transformer World Models. The concept of an agent learning an internal representation of its environment (a world model) to guide decision-making [18] has been powerfully advanced by the subsequent integration of Transformer architectures [8]. These architectures offer significant advantages in this context, particularly their ability to model complex dynamics and capture long-range dependencies within an agent's observation history compared to recurrent world models. These capabilities have led to Transformer-based world models achieving state-of-the-art performance in various challenging sequential decision-making tasks, primarily by enabling more accurate and extended *forward* planning in learned latent spaces. One prominent direction is the integration of sparse attention mechanisms, aiming to reduce computational overhead without sacrificing critical information for dynamics modelling [19].

Influence-Aware Memory (IAM) architectures [20] apply concepts from Influence-Based Abstractions (IBA) to address partial observability challenges in Deep Reinforcement Learning (DRL). IBA argues that not all past observations are equally relevant for predicting future states [21]. Therefore, IAM focuses on identifying and using only the most influential observations to simplify decision making.

To mitigate partial observability, DRL has employed several memory mechanisms: (1) **Frame stacking**, which feeds the agent a moving window of n observations and (2) **Recurrent Neural Networks (RNNs)**, which offer a more scalable solution by mapping the action-observation history into an internal hidden state [22, 23]. However, standard RNNs might struggle to separate critical information, so IAM decouples the learning process: a feedforward network (FFN) handles immediate decision making based on o_t, while the RNN component constructs an internal memory by filtering only the most influential spatial features of an observation. Extending this principle, an influence-aware framework could similarly inform temporal patterns through attention mechanisms in Transformer-based world models to aid the learning process.

4 Environments from an Influence Perspective

This section analyses two classical RL environments from an IBA perspective. Firstly, we introduce the approaches to model the influences in partially observable environments in Section 4.1. The following sections consider the influence dependencies in Atari Pong and Boxing.

4.1 Influence in Partial Observability Environments

In a POMDP setting, the agent cannot directly observe the true state $s \in S$ but instead receives an observation $o \in \Omega$ which is related to s through a probability function. The agent typically maintains a history of these interactions, $H_t = (o_1, a_1, ..., o_{t-1}, a_{t-1})$, which is used to shape a belief state $b_t = P(s_t|H_t)$ which is a probability distribution over the true states. If the belief state is the smallest possible representation summarising the relevant information from H_t, then it qualifies as a minimal sufficient state representation [24].

In graph theory, d-separation (directional separation) provides a formal way to determine if information can flow between different parts of a network, often a Directed Acyclic Graph (DAG): two distinct sets of nodes, X and Y, are d-separated by a third set Z if every path between any node in X and any node in Y is "blocked" by the set Z [25].

A d-separating set, Z_t, in decision-making, would represent a subset of the information available to the agent at time t that is sufficient to make a target variable Target_{t+k} conditionally independent of the rest of the history $H_t \setminus Z_t$, given the current action A_t [26, Sec. 8.2.2]. This can be expressed as:

$$P(\text{Target}_{t+k}|H_t, A_t) = P(\text{Target}_{t+k}|Z_t, A_t). \tag{2}$$

IBA offers a framework to identify such concise and influential representations from the agent's history. By operating on the principle that not all observations or actions are equally important, it seeks to pinpoint the smaller subset of past observations that are most influential for predicting future outcomes, a d-separating set [21]. Leveraging the concept of d-separation, we adapt our world models to approximate a compact subset Z_t of the agent's history that captures the key properties of the underlying true state S_t.

4.2 Pong

In Pong, the agent moves a paddle vertically to intercept a ball bouncing between two paddles. The environment exhibits partial observability, primarily from temporal dependencies, as the ball's current velocity and direction must be inferred from past observations for successful predictions.

Influence dependencies in Pong are relatively simple and mostly static, as the state transitions remain consistent over time. Hence, we could define a simple approximation to the optimal d-separating set in Pong as:

$$Z_t^{\text{Pong}} \approx (y_{p,t}, y_{o,t}, x_{b,t}, y_{b,t}, v_{by,t}), \tag{3}$$

where $(y_{p,t}, y_{o,t})$ are the y-coordinates of the agent and opponent's paddle, $(x_{b,t}, y_{b,t})$ are the coordinates of the ball and $(v_{by,t})$ is the y-velocity component of the ball. Note that in this environment, the x-component of the ball's velocity remains constant while the y-component is directly affected by where it hits a paddle. Hence, only observing a few past frames is enough to estimate all the variables in Z_t^{Pong}.

4.3 Boxing

Boxing presents a more complex scenario. In the game, the agent controls a boxer to land punches on an opponent and score points, while also dodging the opponent's attacks. Partial observability is significant due to unobservable state variables, such as player velocities and the action momentum, which must be inferred from temporal patterns in the observation history to anticipate actions.

The presence of an opponent whose hidden intentions and strategy directly influence the game state introduces further complexity. To act optimally in this dynamic environment, an agent must not only track basic coordinates $(x_{p,t}, y_{p,t}, x_{o,t}, y_{o,t})$ but also continuously infer opponent tendencies and predict their actions from past observations. This requires summarising the agent's history of observations and actions into a compact representation that captures all information relevant for predicting future states and rewards. We propose that such a sufficient set of historical information, crucial for decision-making in Boxing, could be approximated by the following:

$$Z_t^{\text{Boxing}} \approx (x_{p,t}, y_{p,t}, \text{st}_{p,t}, x_{o,t}, y_{o,t}, \text{st}_{o,t}, \text{score}_{p,t}, \text{score}_{o,t}, \Psi(h_t^{\text{local}})) \tag{4}$$

The current stance $(\text{st}_{p,t}, \text{st}_{o,t})$ (e.g. whether the agent is punching) as well as the current scores $(\text{score}_{p,t}, \text{score}_{o,t})$ need to be integrated into the decision making for a more strategic planning. Finally, we also assume there is a function $\Psi(h_t^{\text{local}})$ which extracts from the recent local history a compact representation of how the opponent's evolving strategy is influencing the game. The approximated d-separating set must be flexible, continuously updating based on observed patterns in the opponent's behaviour. Estimating many of these variables, particularly the opponent's evolving strategy captured demands integrating information from more distant frames than would typically be required in a simpler game like Pong.

5 Influence-Aware Attention

This section details our study into attention mechanisms from an influence perspective, designed to refine how Transformer-based world models learn. In Section 5.1, we analyse the limitations of using causal attention mechanisms for world models. Most of these limitations involve handling short-term tasks. Section 5.2 introduces local attention patterns and their implications. Finally, we consider adaptive attention mechanisms for dynamic world models in Section 5.3.

5.1 Limitations of Causal Attention World Models

Introducing the attention mechanism has numerous implications for the agent's training and inference process. First, the complete observation sequence $o_{1:t}$ can now be used during training, enabling the world model to learn from full contextual histories rather than truncated windows. UniZero's backbone also explicitly separates the latent state z_t from the learned implicit history h_t. In training, the Transformer processes the context $M_{\text{enc}} = (z_{t-H}, a_{t-H}, \ldots, z_{t-1}, a_{t-1})$, where a history of length H is used. The attention mechanism operates temporally across this sequence in a single forward pass, treating each latent state z_t and action a_t as a distinct token rather than attending to features within each vector. This yields an attention matrix of size $|M_{\text{enc}} \times M_{\text{enc}}|$. The model injects token positions via learned absolute positional embeddings [27]. During inference, the encoder uses the entire observed history in a single pass to generate z_t, enabling the MCTS root to leverage the richer context $M_{\text{infer}} = (z_{t-H_{\text{infer}}}, a_{t-H_{\text{infer}}}, ..., z_t, a_t)$. Figure 1 shows the architecture details during training and inference.

All these properties combined yield outstanding results in memory-intensive tasks. For instance, UniZero achieves a higher human-normalised median score than MuZero in 15 out of 26 Atari games [5]. However, in several games like Boxing and Pong, the new architecture degrades compared to standard MuZero. These properties that allow UniZero to excel in memory domains are potentially responsible for its subtle underperformance in short-dependency domains.

In games with short-memory dependencies, the d-separating set is likely very small. For any given prediction of the next state z_{t+1} or reward r_t, only a small subset of the history is typically relevant. UniZero's attention

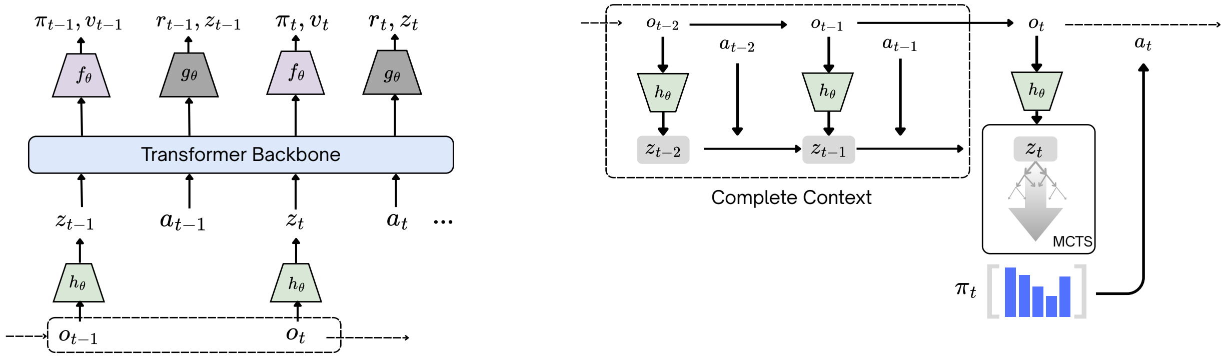

Figure 1: Transformer-Based World Model: The training architecture (left) , comprised of an encoder h_θ, a Transformer backbone, a decision head f_θ, and a dynamics head g_θ. The encoder maps an observation o_t to a latent state z_t, and the Transformer backbone autoregressively processes the history of states and actions to predict future outcomes and policies. At inference (right), the model conditions on the complete observed context to start a Monte-Carlo Tree Search (MCTS) planning process, which computes an improved policy π_t. For learning, optimisation is guided by a soft-target model $\hat{W} = (\hat{h}_\theta, \hat{g}_\theta, \hat{f}_\theta)$, which is maintained as an Exponential Moving Average (EMA) of the online world model's weights.

mechanism lacks an efficient method to initially isolate this subset, so it must learn to downweight nearly all of the remaining history M. However, in Atari environments, without strong influence priors, learning to focus on the few relevant frames can become particularly challenging when data is scarce.

5.2 Local Attention

Local attention mechanisms have been introduced in Transformer architectures as a solution to mitigate the computational overhead inherent in causal attention. Local attention constraints each token to attend only to a fixed window of neighbouring tokens of size w by masking out every other token, reducing computational costs to $\sim \mathcal{O}(nw)$, as opposed to $\mathcal{O}(n^2)$. The inductive bias we embed by integrating is predominantly local: the most critical information for understanding the present and predicting the immediate future is presumed to lie in the very recent past. This contrasts sharply with causal attention, which initially assigns equal importance to all past tokens, irrespective of their temporal distance. This draws links to Convolutional Neural Networks (CNNs), which enforce locality through their kernel operations [28].

Formally, the local attention for token i with a local window size w is given by:

$$\text{Attention}(Q_i, K, V) = \text{softmax}(\frac{Q_i K_{i-w:i+w}^T}{\sqrt{d_k}})V_{i-w:i+w} \tag{5}$$

where $K_{i-w:i+w}$ and $V_{i-w:i+w}$ are subsets restricted to tokens within the local window around the i-th token. In this particular setting, sequence-modelling tasks require us to model the attention autoregressively, so we prevent the model from attending to future observations. In practice, we apply a mask W_{ij} (with entries $-\infty$ for $|i-j| > w$) on top of the causal mask to the attention logits in Eq. (1) *before* applying the softmax, effectively filtering attention scores without directly calculating them.

Introducing local attention into UniZero's backbone offers a way to approximate the environment's d-separating set by focusing on the most recent w tokens, regardless of incoming observations. This is especially valuable with limited data, since a causal mechanism can struggle to extract short-term dependencies from noisy long histories, and causal attention risks diluting crucial recent information in favour of distant, often irrelevant context [15]. However, the model cannot capture dependencies beyond the window w.

5.3 Gaussian Adaptive Attention

The main limitation with local, hard attention spans comes from its poor flexibility. Adaptive attention overcomes the rigidity of hard, fixed windows by learning a soft mask for each attention head [9]. The concept of adaptivity introduces a learnable soft mask per head: given a distance matrix $\Delta_{ij} = |i - j|$, the soft mask is given by:

$$W_{ij}^{(h)} = \min(\max(\frac{R + z_h - \Delta_{ij}}{R}, 0), 1) \tag{6}$$

where z_h is a learnable span for head h and R is a fixed hyperparameter called the *ramp width*, which controls how sharply the mask falls from 1 to 0. In effect, $W_{ij}^{(h)}$ equals 1 for $\Delta_{ij} \leq z_h$, decays linearly to 0 over the next R tokens, and is 0 beyond $\Delta_{ij} \geq z_h + R$. The mask W_{ij} is applied to the attention logits in Eq. (1).

Because we parametrise $z_h = \text{softplus}(s_h)$, for any token pair whose distance falls into the ramp region,

$$\frac{\partial W_{ij}^{(h)}}{\partial s_h} = \frac{\partial W_{ij}^{(h)}}{\partial z_h} \frac{\partial z_h}{\partial s_h} = \frac{1}{R} \frac{\partial \ \text{softplus}(s_h)}{\partial s_h} > 0, \tag{7}$$

ensuring non-zero gradient flow into the span parameter.

We extend this concept to implement a related variant, the Gaussian Adaptive Attention Mechanism (GAAM) [10], which models influence with a bell-shaped mask:

$$W_{ij}^{(h)} = \exp(-\frac{(\Delta_{ij} - \mu_h)^2}{2\sigma_h^2}) \tag{8}$$

where μ_h and σ_h are head-specific, softplus-parametrized parameters ($\mu_h = \text{softplus}(m_h) \in [0, L_{\max}], \sigma_h = \text{softplus}(v_h) > 0$). Here, tokens with $\Delta_{ij} = \mu_h$ receive a mask weight of 1, and beyond $\mu_h \pm 3\sigma_h$ the attention decays almost to zero, similar to a Gaussian distribution $\sim \mathcal{N}(\mu_h, \sigma_h^2)$. The inductive bias being introduced assumes that tokens whose relative distance Δ_{ij} lies near the learned centre μ_h are assumed to be the most influential for token i. This pattern offers a powerful alternative to the previous attention mechanisms by allowing the model to shift focus towards specific observations or actions in H_t instead of shaping a window adjacent to the current token. Each head learns to approximate the d-separating set $\hat{Z}_t$ dynamically, based on each token. See Figure 2 for an illustration of the different attention mechanisms.

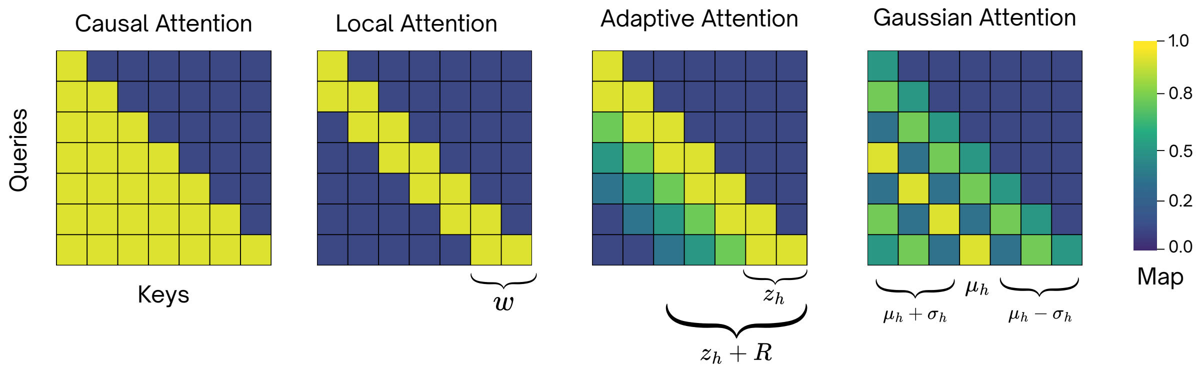

Figure 2: Comparison of Attention Masks: Each map shows queries (rows) attending to keys (columns); yellow indicates high weight, dark blue zero weight. All attention types enforce autoregressiveness via a causal (upper-triangular) mask, preventing access to future keys. (a) Causal attention attends to all past tokens. (b) Local attention restricts attention to a fixed recent window. (c) Adaptive attention learns a dynamic window defined by span z_h and ramp R. (d) Gaussian adaptive attention softly weights keys via a learned Gaussian curve with centre μ_h and width σ_h.

6 Experiments

To evaluate the ability of the proposed attention mechanisms to model environments, we conduct evaluations across different short- and long-memory domains. Specifically, we evaluate UniZero in the Single-Task (ST) setting on three environments from the Atari 100k benchmark [29]. To measure whether the difference between models is statistically significant, we employ Welch's t-test at the $\alpha = 0.05$ level [30].

6.1 Setup

Codebase: All experiments are implemented on top of the LightZero codebase[1]. All the implemented attention mechanisms can be found under `/lzero/model/unizero_world_models`. Additional analytics scripts to aggregate and show results can be found in `/analysis`.

Environments: The models are evaluated on 4 different Atari environments: Pong, Boxing, BankHeist and MsPacman. The first 2 represent short-memory, tightly-coupled environments, while the last 2 are environments where long-memory capabilities are advantageous. The agents interact with each environment for 100k environment steps in training, and an evaluation is initialised every 10k environment steps. The shared hyperparameters can be found in Appendix A.

Computational Cost: All experiments were conducted on the DelftBlue cluster[2] configured with a single NVIDIA Tesla A100 / V100 GPU, 15-20 CPU cores, and 60-80 GB of RAM. Training an Atari agent for 100k environment steps requires approximately 4-5 hours.

Baseline: The new attention mechanisms are compared against vanilla UniZero: the original Transformer-based world model with causal attention [5].

6.2 Attention Mechanisms Benchmark on Atari Games

We evaluate causal (baseline), local, and Gaussian adaptive attention mechanisms across four Atari games to assess their performance on tasks requiring different memory dependencies. Table 1 summarises the mean scores and standard errors for each model across all environments. We align the attention bias to each environment's memory needs: in **Pong** we use local $w = 2$ and Gaussian $\sim \mathcal{N}(2, 1)$; in **Boxing** local $w = 6$ and Gaussian $\sim \mathcal{N}(6, 1)$; and both **BankHeist**, **MsPacman** with local $w = 10$ and Gaussian $\sim \mathcal{N}(10, 2)$. These configurations are the best performing in each environment. More in-depth analysis of the configurations can be found in Section 6.4.

Table 1: Atari Benchmark Score: Mean episode returns (± standard error) for UniZero variants using causal (baseline), local, and Gaussian adaptive attention across four Atari games. Best performing models are marked in bold.

Environment	UniZero (Baseline)	Local UniZero (Ours)	Gaussian UniZero (Ours)
Pong	-14.53 ± 0.66	-9.35 ± 0.76	**-6.86 ± 1.82**
Boxing	0.14 ± 0.63	0.62 ± 0.91	0.83 ± 1.10
MsPacman	643.93 ± 35.66	624.68 ± 59.89	**928.12 ± 128.43**
BankHeist	91.34 ± 28.03	**191.30 ± 33.00**	94.08 ± 35.80

Short-Term Memory Tasks: In Pong (Fig. 3, top), both Gaussian (p-value = 0.0093) and local attention (p-value = 0.0008) significantly outperform vanilla causal attention within 100 k steps by focusing on the most recent frames; moreover, Gaussian's learned μ_h settles to a value well below its initial value. By contrast, Boxing has no significant sample-efficiency gap, and the learned Gaussian μ_h exhibits high variability (Fig. 4, top) with large standard deviations, evidence that dynamic action patterns and more diverse attention space demands more flexible attention. Overall, when an attention bias aligns with an environment's true short-term dependencies, the model learns more efficiently.

[1] Code can be found in: github.com/daniallegue/UniZero.

[2] Delftblue Documentation: doc.dhpc.tudelft.nl/delftblue

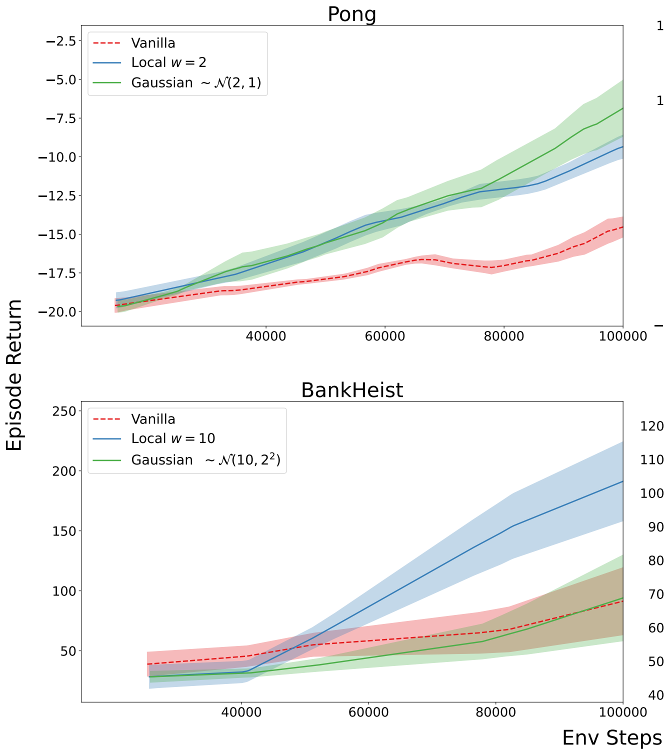

Figure 3: Learning curves for the UniZero model variants across four Atari games: Pong, Boxing, BankHeist, and MsPacman. The curves depict mean episode returns (± standard error) versus environment steps. Local and Gaussian adaptive attention mechanisms generally outperform or match the performance of vanilla causal attention, highlighting task-specific benefits of tailored attention windows and offsets. All experiments used 7 seeds.

Memory-Intensive Tasks: In BankHeist (Fig. 3, bottom), local attention with $w = 10$ remains significantly more sample-efficient than vanilla UniZero (p-value=0.0439), while Gaussian heads learn $\mu_h \approx 10$ across all heads, anchoring on their initial span and yielding no significant performance gain. Conversely, in MsPacman the learned μ_h values fan out from short (~ 4) to longer (~ 11) offsets (Fig. 4, bottom), allowing Gaussian attention to capture both immediate and distant dependencies and significantly outperform the baseline (p-value=0.0487), whereas fixed-window attention only matches the baseline. These results suggest that simple windows can suffice to model longer-term dependencies, but Gaussian adaptivity pays off when it can flexibly allocate attention across varying dependencies.

6.3 Comparative Analysis of Attention Mechanisms

In Pong (Fig. 5, top), causal attention distributes weight broadly over many past frames (broad diagonal); local attention ($w = 2$) confines all weight to the two most recent tokens (sharp, narrow diagonal); and Gaussian attention ($\mu_h = 2, \sigma_h = 1$) peaks at a two-step offset and smoothly decays, blending strict locality with slight flexibility. These patterns confirm that a tight locality bias best matches Pong's short-term dynamics. In Boxing (Fig. 5, bottom), causal attention shows variable, sometimes long-range dependencies; local attention remains narrowly focused; and Gaussian attends immediate and distant frames, better capturing the environment's dynamic dependencies. Full attention maps for Pong are in Appendix C.

We further verify that these localised attention patterns do not compromise predictive accuracy: Appendix D contains full example trajectories for both local and Gaussian world models (Figures 11 and 12), showing reward predictions remaining exactly zero and correct paddle actions under MCTS, on par with the causal attention baseline. Thus, constraining attention to the nearest timesteps improves efficiency without sacrificing the fidelity of state, reward, or policy predictions in Pong.

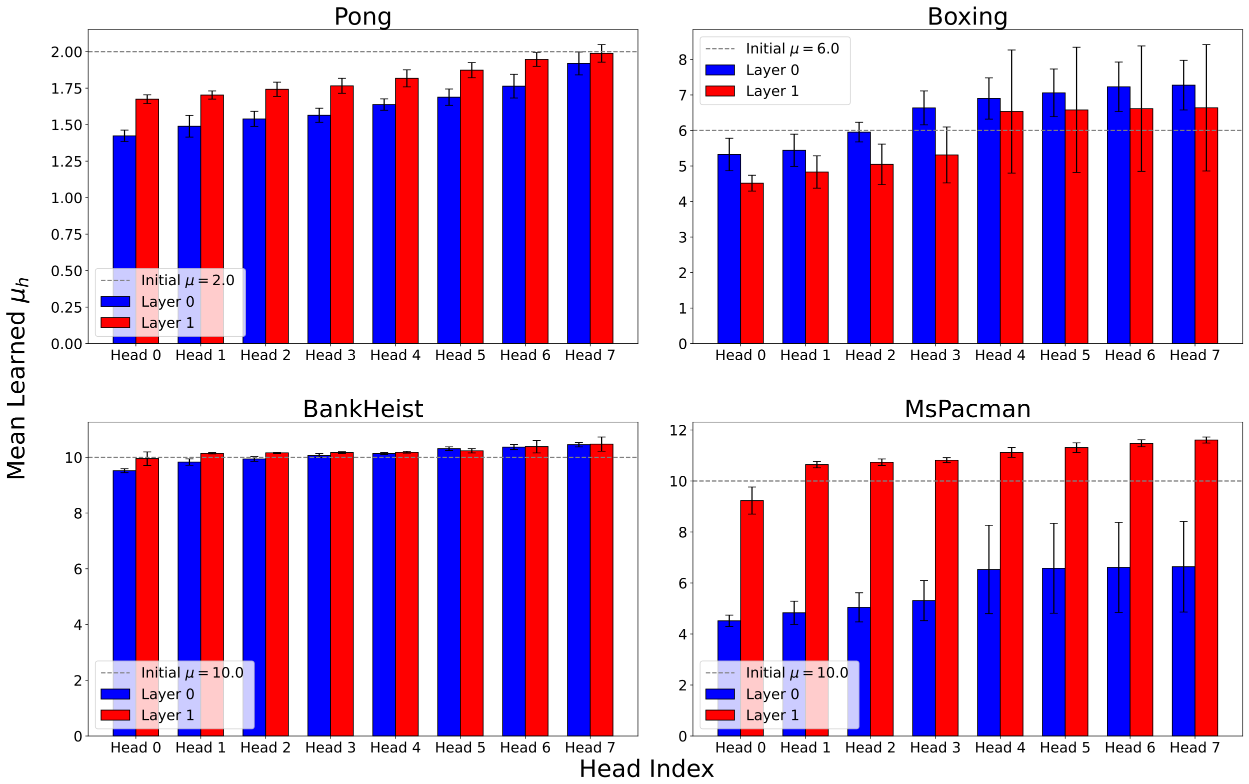

Figure 4: Mean learned mean-offsets per attention head for Layer 0 (blue) and Layer 1 (red) in Pong and Boxing (top row), BankHeist and MsPacman (bottom row), with standard-deviation error bars and the initial offset shown as a dashed line. MsPacman and Boxing environments benefit from head divergence to capture both short- and long-range dependencies, while Pong and BankHeist learn offsets close to the initial value.

In Figure 6, you can see that the learned Gaussian delay perfectly matches the Boxing environment's action-effect latency. When the agent presses "punch" at a_7 (just after Frame 7), the visual effect appears in Frame 8, and the glove only reaches full extension by Frame 10. By placing its Gaussian $\mu_h = 6$ at a fixed 6 token shift (i.e. three states/actions), the head automatically aligns a_{10} (the moment the punch arc is at its peak) with a_7 (the button press that initiated it), as evidenced by the bright green square at (19, 13). This fixed 3-frame look-back in Gaussian attention mitigates the temporal credit-assignment problem [31]. So the model knows which punch press produced which impact and gives the agent the exact timing it needs to execute and learn effective actions under delayed feedback.

6.4 Hyperparameter Sensitivity in Attention Mechanisms

We analyse hyperparameter sensitivity for local and Gaussian attention mechanisms in Pong, Boxing and MsPacman (Table 3). Narrowly focused attention yields substantial gains, especially in Pong, aligning closely with its short-term dynamics. Deviating from these optimal settings, by broadening the local window or increasing the Gaussian width, leads to clear performance declines. Boxing benefits most from moderate attention spans, while MsPacman requires wider Gaussian windows to capture longer-range dependencies. Specifically, fixing $\mu_h = 2$ and increasing σ_h from 1 to 4 significantly worsens performance in Pong and Boxing; similarly, in MsPacman, increasing σ_h from 2 to 4 at $\mu_h = 10$ decreases performance by over 26%. This highlights the crucial role of carefully matching attention parameters to environment dynamics. This ablation study offers an initial exploration of the sensitivity of the attention parameters; however, a more comprehensive hyperparameter optimisation is reserved for future work to establish more generalisable principles. A summary of the results and full learning curves for Pong are provided in Appendix B.

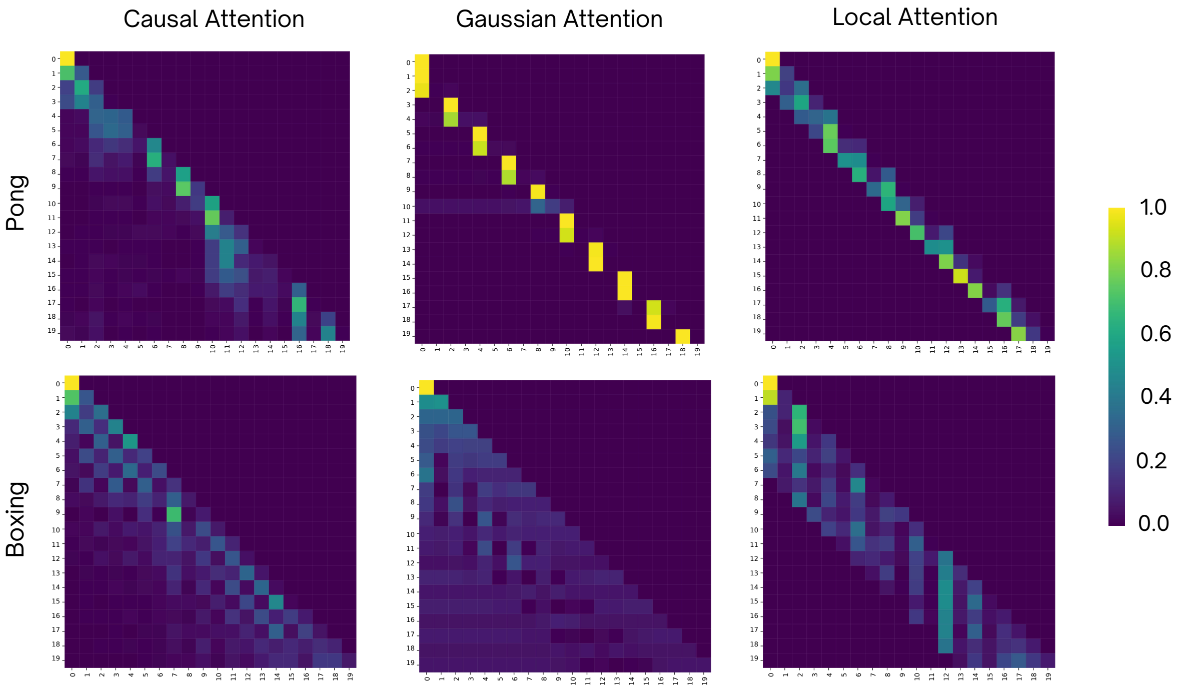

Figure 5: Example attention maps in Pong (top row) and Boxing (bottom row). Left: Causal attention attends broadly across past tokens, forming a wide diagonal band. Centre: Gaussian attention peaks at a two-step lag with smooth decay, blending tight focus and flexibility. Right: Local attention strictly confines weights to the w immediately preceding tokens, yielding a sharp, narrow diagonal. In all maps, yellow denotes higher attention from query (y-axis) to key (x-axis), and dark blue denotes zero weight.

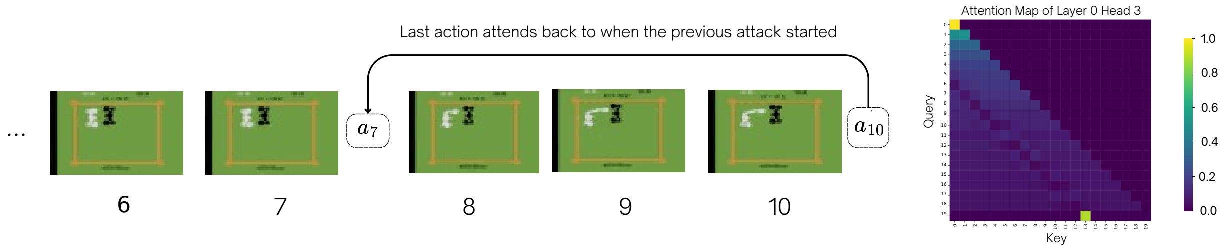
Last action attends back to when the previous attack started
...
a_7
a_{10}
6
7
8
9
10
Attention Map of Layer 0 Head 3
Query
Key
1.0
0.8
0.6
0.4
0.2
0.0

Figure 6: Example Attention Head (right) in the Boxing environment learns a fixed 3-frame Gaussian delay: when processing the action at time step 10 (a_{10}, the query), the head attends most strongly to the punch initiation at time 7 (a_7, the key), as illustrated by the arrow and the bright green square in the bottom of the attention map. This alignment perfectly captures the action-to-impact delay.

7 Conclusions and Future Work

This paper investigates integrating alternative attention mechanisms into Transformer-based world models to enhance data efficiency in model-based RL. Causal attention can be sample inefficient when environments have short-term memory dependencies, since it must learn to ignore irrelevant history within a limited number of interactions. To address this, we replaced it in UniZero's Transformer with local and adaptive Gaussian attention as strong inductive biases that focus the model on temporally relevant information. On the Atari 100k benchmark, when these biases align with an environment's dynamics, such as Pong's strictly short-term memory, sample efficiency improves significantly over the baseline. In BankHeist, local attention's fixed window suffices to capture the game's longer-range dependencies, yielding gains without any need for Gaussian adaptivity. In environments with dynamic memory demands like Boxing, where local attention fails to capture long-range dependencies, and MsPacman, which requires both short and long memory dependencies, Gaussian adaptive attention excels. These findings show that simple influence-based priors can make world models far more resource-efficient in data-limited regimes, pointing toward more targeted world model designs.

7.1 Discussion

Our evaluation reveals that the effectiveness of influence-aware attention mechanisms is determined by the alignment between the embedded inductive bias and the environment's temporal dynamics.

In environments with highly localised dependencies like Pong, providing a strong, correctly matched inductive bias via narrow local or Gaussian attention spans led to significant performance gains over the baseline. This demonstrates that when the model's focus is correctly constrained, it can learn critical short-term patterns with substantially greater data efficiency. Conversely, in Boxing, a game with more dynamic and a more diverse action space, the rigidity of these fixed priors offered no significant advantage over causal attention. This finding points to a crucial trade-off: the same strong inductive biases that make the model highly efficient when correctly matched with the environment can become a rigid constraint that hinders performance when faced with more dynamic and unpredictable memory dependencies.

Interestingly, in long-memory tasks such as BankHeist and MsPacman, both local and Gaussian attention with moderately sized spans proved competitive with the original attention model. In MsPacman, Gaussian attention's learned combination of short and long spans enables effective modelling of both immediate cues and extended dependencies. This suggests that attending to the full context history is not always necessary and that these mechanisms can effectively approximate longer-memory dependencies, often at a reduced computational cost.

This aligns with the theoretical principles of Influence-Based Abstraction (IBA), where identifying and focusing on subsets of relevant information can lead to more efficient learning. Local attention provides a fixed approximation of the influential variables, while adaptive mechanisms like Gaussian attention learn a flexible approximation based on each observation. Our work suggests that applying these principles to find temporal dependencies can make Transformer-based world models more data-efficient.

7.2 Limitations

This study has several limitations. The hyperparameter settings for the attention mechanisms (window sizes, initial Gaussian parameters) were chosen based on an initial analysis of the environments but were not exhaustively tuned. A more thorough ablation study, as indicated in Section 6.4, is needed to understand the sensitivity to these parameters. Furthermore, future work should incorporate a greater number of repetitions for each experiment to support the statistical robustness of our findings. Evaluating the proposed attention mechanisms across a more diverse suite of environments, encompassing a broader spectrum of short-term and long-term memory demands, would also be crucial for assessing their generalisability. Specifically, to robustly evaluate performance in memory-intensive domains, future studies should also employ continuous-control environments such as those available in the DeepMind Control Suite [32]. Additionally, conducting comparative analyses with state-of-the-art models like DreamerV3 [33] would provide deeper insights into the relative effectiveness and limitations of our approach.

7.3 Future Work

Building on our findings, we identify several promising opportunities for future research. A primary direction is to conduct a large-scale evaluation of the proposed attention mechanisms across the entire Atari 100k benchmark [29]. This will rigorously test their generalisability and performance in environments with more diverse and complex memory dependencies.

Furthermore, we plan to develop more sophisticated hybrid architectures. This includes designing models that dynamically combine local and adaptive attention patterns through learned gating mechanisms, enabling automatic adaptation to environment-specific temporal influences [34]. A natural extension of this is to make the local attention window itself learnable and adaptive, rather than a fixed hyperparameter [9]. We also aim to integrate principles of memory-augmented attention systems [35] to more efficiently store and retrieve important temporal information.

Finally, we will investigate methods to improve the data efficiency and rapid adaptation of these models through meta-learning [36]. This involves developing targeted regularisation techniques to better initialise the Gaussian adaptive attention parameters (μ_h, σ_h), allowing the model to quickly adapt to new environments from minimal interaction data.

8 Responsible Research

This section addresses considerations related to the reproducibility of this study and the ethical implications associated with the use of transformer-based models in RL tasks.

Reproducibility. The main repository extends from the LightZero [37] codebase. The codebase can be found in: github.com/daniallegue/UniZero. Detailed information regarding implementations, training procedures, hyperparameters, and data generation is thoroughly documented within the ReadMe file of the codebase and this paper. When performing n runs, we used random seeds 1–n. LLM tools (Gemini 2.5 Pro) were used exclusively for several tasks: summarising and synthesising text, searching for related literature work, generating plotting scripts and assisting with data processing from Wandb.

Ethical Considerations. Our work primarily explores theoretical enhancements in the data efficiency of transformer-based reinforcement learning world models, specifically focusing on attention mechanisms. While these improvements do not inherently entail direct ethical risks, Transformer models can generalise widely, and their deployment in sensitive environments warrants cautious ethical consideration. We highlight the importance of monitoring for unintended biases. Moreover, computationally intensive training of these models has environmental implications due to energy consumption; thus, all experiments were conducted on efficient computing clusters, and resources were utilised responsibly. All steps taken have been motivated using referenced literature, and the limitations of the results have been discussed in Section 7.2 to ensure maximal integrity of the results arrived at in this paper.

Acknowledgments. I would like to thank my professor, supervisor, and fellow research peers for their valuable feedback, ongoing support and enjoyable collaboration throughout this project.

References

[1] Tianwei Ni, Michel Ma, Benjamin Eysenbach, and Pierre-Luc Bacon. When do transformers shine in rl? decoupling memory from credit assignment. *Advances in Neural Information Processing Systems*, 36:50429–50452, 2023.

[2] Maciej Świechowski, Konrad Godlewski, Bartosz Sawicki, and Jacek Mańdziuk. Monte carlo tree search: A review of recent modifications and applications. *Artificial Intelligence Review*, 56(3):2497–2562, 2023.

[3] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

[4] Weirui Ye, Shaohuai Liu, Thanard Kurutach, Pieter Abbeel, and Yang Gao. Mastering atari games with limited data. *Advances in neural information processing systems*, 34:25476–25488, 2021.

[5] Yuan Pu, Yazhe Niu, Zhenjie Yang, Jiyuan Ren, Hongsheng Li, and Yu Liu. Unizero: Generalized and efficient planning with scalable latent world models. *arXiv preprint arXiv:2406.10667*, 2024.

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[7] Nayoung Lee, Kartik Sreenivasan, Jason D Lee, Kangwook Lee, and Dimitris Papailiopoulos. Teaching arithmetic to small transformers. *arXiv preprint arXiv:2307.03381*, 2023.

[8] Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. *arXiv preprint arXiv:2209.00588*, 2022.

[9] Sainbayar Sukhbaatar, Edouard Grave, Piotr Bojanowski, and Armand Joulin. Adaptive attention span in transformers. *arXiv preprint arXiv:1905.07799*, 2019.

[10] Georgios Ioannides, Aman Chadha, and Aaron Elkins. Gaussian adaptive attention is all you need: Robust contextual representations across multiple modalities. *CoRR*, 2024.

[11] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction.* MIT Press, Cambridge, MA, second edition, 2018.

[12] Edward Jay Sondik. *The Optimal Control of Partially Observable Markov Processes.* PhD thesis, Stanford University, Stanford, CA, 1971.

[13] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

[14] Benjamin Eysenbach, Alexander Khazatsky, Sergey Levine, and Russ R Salakhutdinov. Mismatched no more: Joint model-policy optimization for model-based rl. *Advances in Neural Information Processing Systems*, 35:23230–23243, 2022.

[15] Zhen Qin, Xiaodong Han, Weixuan Sun, Dongxu Li, Lingpeng Kong, Nick Barnes, and Yiran Zhong. The devil in linear transformer. *arXiv preprint arXiv:2210.10340*, 2022.

[16] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

[17] Shakti Kumar, Jerrod Parker, and Panteha Naderian. Adaptive transformers in rl. *arXiv preprint arXiv:2004.03761*, 2020.

[18] David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

[19] Anson Lei, Bernhard Schölkopf, and Ingmar Posner. Spartan: A sparse transformer learning local causation. *arXiv preprint arXiv:2411.06890*, 2024.

[20] Miguel Suau, Jinke He, Elena Congeduti, Rolf AN Starre, Aleksander Czechowski, and Frans A Oliehoek. Influence-aware memory architectures for deep reinforcement learning. *arXiv preprint arXiv:1911.07643*, 2019.

[21] Frans Oliehoek, Stefan Witwicki, and Leslie Kaelbling. Influence-based abstraction for multiagent systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pages 1422–1428, 2012.

[22] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[23] Matthew J Hausknecht and Peter Stone. Deep recurrent q-learning for partially observable mdps. In *AAAI fall symposia*, volume 45, page 141, 2015.

[24] Robert Givan, Thomas Dean, and Matthew Greig. Equivalence notions and model minimization in markov decision processes. *Artificial intelligence*, 147(1-2):163–223, 2003.

[25] Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Elsevier, 2014.

[26] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, 2006.

[27] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pages 4171–4186, 2019.

[28] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 2002.

[29] Lukasz Kaiser, Mohammad Babaeizadeh, Piotr Milos, Blazej Osinski, Roy H Campbell, Konrad Czechowski, Dumitru Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine, et al. Model-based reinforcement learning for atari. *arXiv preprint arXiv:1903.00374*, 2019.

[30] Bernard L Welch. The generalization of 'student's'problem when several different population varlances are involved. *Biometrika*, 34(1-2):28–35, 1947.

[31] Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3: 9–44, 1988.

[32] Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

[33] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

[34] Kaitao Song, Xu Tan, Furong Peng, and Jianfeng Lu. Hybrid self-attention network for machine translation. *arXiv preprint arXiv:1811.00253*, 2018.

[35] Emilio Parisotto, Francis Song, Jack Rae, Razvan Pascanu, Caglar Gulcehre, Siddhant Jayakumar, Max Jaderberg, Raphael Lopez Kaufman, Aidan Clark, Seb Noury, et al. Stabilizing transformers for reinforcement learning. In *International conference on machine learning*, pages 7487–7498. PMLR, 2020.

[36] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[37] Yazhe Niu, Yuan Pu, Zhenjie Yang, Xueyan Li, Tong Zhou, Jiyuan Ren, Shuai Hu, Hongsheng Li, and Yu Liu. Lightzero: A unified benchmark for monte carlo tree search in general sequential decision scenarios. *Advances in Neural Information Processing Systems*, 36:37594–37635, 2023.

[38] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

A Hyperparameters

Table 2: Key Hyperparameters. The values are aligned with those in [5] for Atari environments. The section on **Attention** refers to the newly added parameters.

Hyperparameter	**Value**
Planning	
Number of MCTS Simulations (sim)	50
Inference Context Length (H_{infer})	4
Temperature	0.25
Dirichlet Noise (α)	0.3
Dirichlet Noise Weight	0.25
Coefficient c_1	1.25
Coefficient c_2	19652
Environment and Replay Buffer	
Replay Buffer Capacity	1,000,000
Sampling Strategy	Uniform
Observation Shape (Atari)	(3, 64, 64) (stack1)
Reward Clipping	True
Number of Frames Stacked	1 (stack1)
Frame Skip	4
Game Segment Length	400
Data Augmentation	False
Architecture	
Latent State Dimension (D)	768
Number of Transformer Heads	8
Number of Transformer Layers (N)	2
Dropout Rate (p)	0.1
Activation Function	LeakyReLU (encoder); GELU (others)
Reward/Value Bins	101
SimNorm Dimension (V)	8
SimNorm Temperature (τ)	1
Optimization	
Training Context Length (H)	10
Replay Ratio	0.25
Buffer Reanalyze Frequency	1/50
Batch Size	64
Optimizer	AdamW [38]
Learning Rate	1×10^{-4}
Next Latent State Loss Coefficient	10
Reward Loss Coefficient	1
Policy Loss Coefficient	1
Value Loss Coefficient	0.5
Policy Entropy Coefficient	1×10^{-4}
Weight Decay	10^{-4}
Max Gradient Norm	5
Discount Factor	0.997
Soft Target Update Momentum	0.05
Hard Target Network Update Frequency	100
Temporal Difference (TD) Steps	5

(continued)

Hyperparameter	**Value**
Evaluation Frequency	10k Collector Steps
Attention	
Rotary Positional Embeddings (`rotary_emb`)	False
Local Window Size (`local_window_size`)	Varied across models
Initial Gaussian Mean Offset μ (`init_adaptive_mu`)	Varied across models
Initial Gaussian Standard Deviation σ (`init_adaptive_sigma`)	Varied across models
Diversity Regularization (`gaam_span_diversity_coeff`)	0.0

B Attention Hyperparameter Ablation

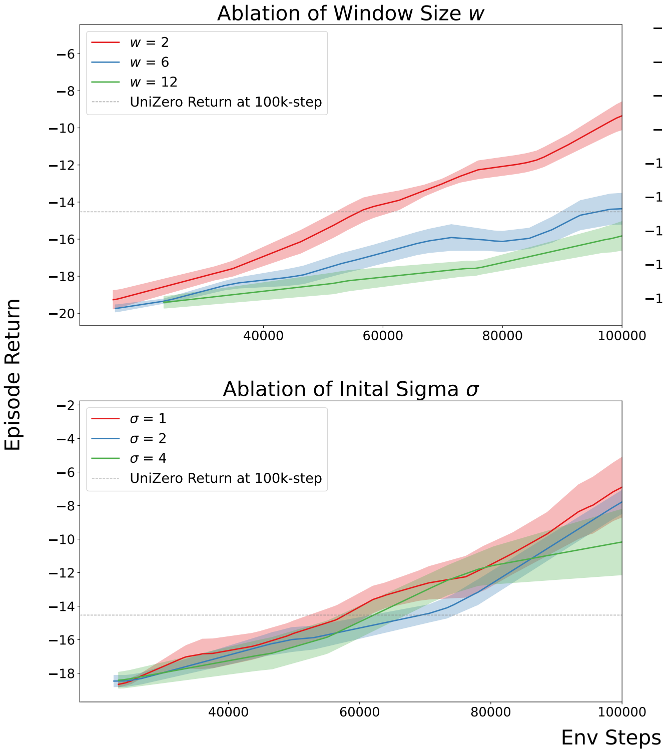

Figure 7: Learning curves illustrating hyperparameter sensitivity for UniZero's attention mechanisms in Pong. Top-left: Ablation of Local attention window size (w). Top-right: Ablation of Gaussian attention mean offset (μ_h). Bottom: Ablation of Gaussian attention standard deviation (σ_h). Optimal hyperparameter choices consistently yield faster convergence and higher returns, confirming the significance of precise parameter tuning aligned with environment-specific temporal dynamics. The horizontal line represents the mean return of vanilla UniZero at the 100k environment step. 5 seeds were used for each configuration. Note: For $w = 12$ in the first ablation (top-left), we plot evaluations after the 30k step only for computational ease.

Table 3: Hyperparameter Ablation (100k steps). Mean episode returns ($\pm$ standard error) for UniZero variants on Pong, Boxing and MsPacman, evaluated over 5 seeds. We compare local attention with window sizes $w \in \{2, 6, 10, 12\}$ and Gaussian adaptive attention with $(\mu_h, \sigma_h) \in \{2, 6, 10, 12\} \times \{1, 2, 4\}$. The best result per game is underlined; "–" indicates a configuration that was not evaluated. No configuration yields a statistically significant improvement in Boxing.

Model	**Pong**	**Boxing**	**MsPacman**
Vanilla UniZero	-14.53 ± 0.65	0.14 ± 0.63	643.93 ± 35.66
Local ($w = 2$)	-9.35 ± 0.76	-1.32 ± 1.00	764.44 ± 73.27
Local ($w = 6$)	-14.36 ± 0.84	0.62 ± 0.91	851.7 ± 253.2
Local ($w = 10$)	-	-	624.68 ± 59.89
Local ($w = 12$)	-15.83 ± 0.78	0.45 ± 1.37	-
Gaussian ($\mu_h = 2, \sigma_h = 1$)	-6.86 ± 1.79	-0.13 ± 1.69	-
Gaussian ($\mu_h = 6, \sigma_h = 1$)	-8.78 ± 1.04	0.83 ± 1.10	-
Gaussian ($\mu_h = 12, \sigma_h = 1$)	-9.25 ± 1.26	1.00 ± 1.03	-
Gaussian ($\mu_h = 2, \sigma_h = 2$)	-	-	778.7 ± 79.8
Gaussian ($\mu_h = 6, \sigma_h = 2$)	-	-	707.78 ± 152.14
Gaussian ($\mu_h = 10, \sigma_h = 2$)	-	-	928.12 ± 128.43
Gaussian ($\mu_h = 2, \sigma_h = 1$)	-6.86 ± 1.79	0.22 ± 0.78	-
Gaussian ($\mu_h = 2, \sigma_h = 2$)	-7.78 ± 0.69	-0.67 ± 2.11	-
Gaussian ($\mu_h = 2, \sigma_h = 4$)	-10.17 ± 1.96	-0.78 ± 1.85	-
Gaussian ($\mu_h = 10, \sigma_h = 1$)	-	-	898.9 ± 239.9
Gaussian ($\mu_h = 10, \sigma_h = 2$)	-	-	928.12 ± 128.43
Gaussian ($\mu_h = 10, \sigma_h = 4$)	-	-	706.67 ± 58.34

C Attention Map Analysis

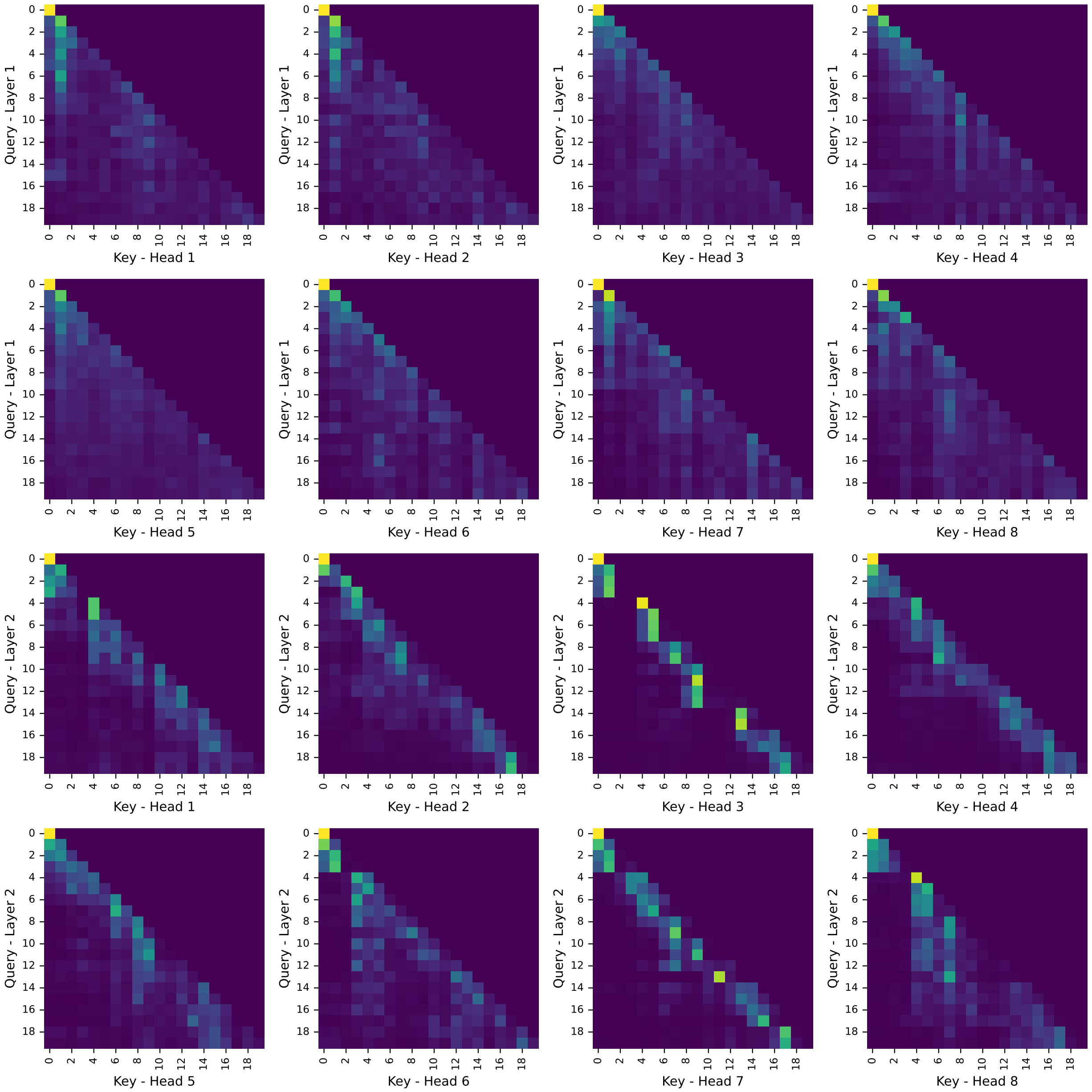

Figure 8: Causal attention maps for a two-layer, eight-head Transformer world model on a single Pong trajectory. The first eight heatmaps show layer 1 heads, the next eight show layer 2. Yellow indicates high attention, while dark blue indicates no attention weight. In every head, queries (y-axis) overwhelmingly attend to keys (x-axis) from the most recent frames; especially the immediately preceding latent state/action pair, reflecting Pong's inherently short-term dynamics and demonstrating that information from the last few timesteps suffices for accurate state prediction and policy/value learning. However, most heads still suffer from attention dilution by attending to distant frames considered irrelevant in this environment.

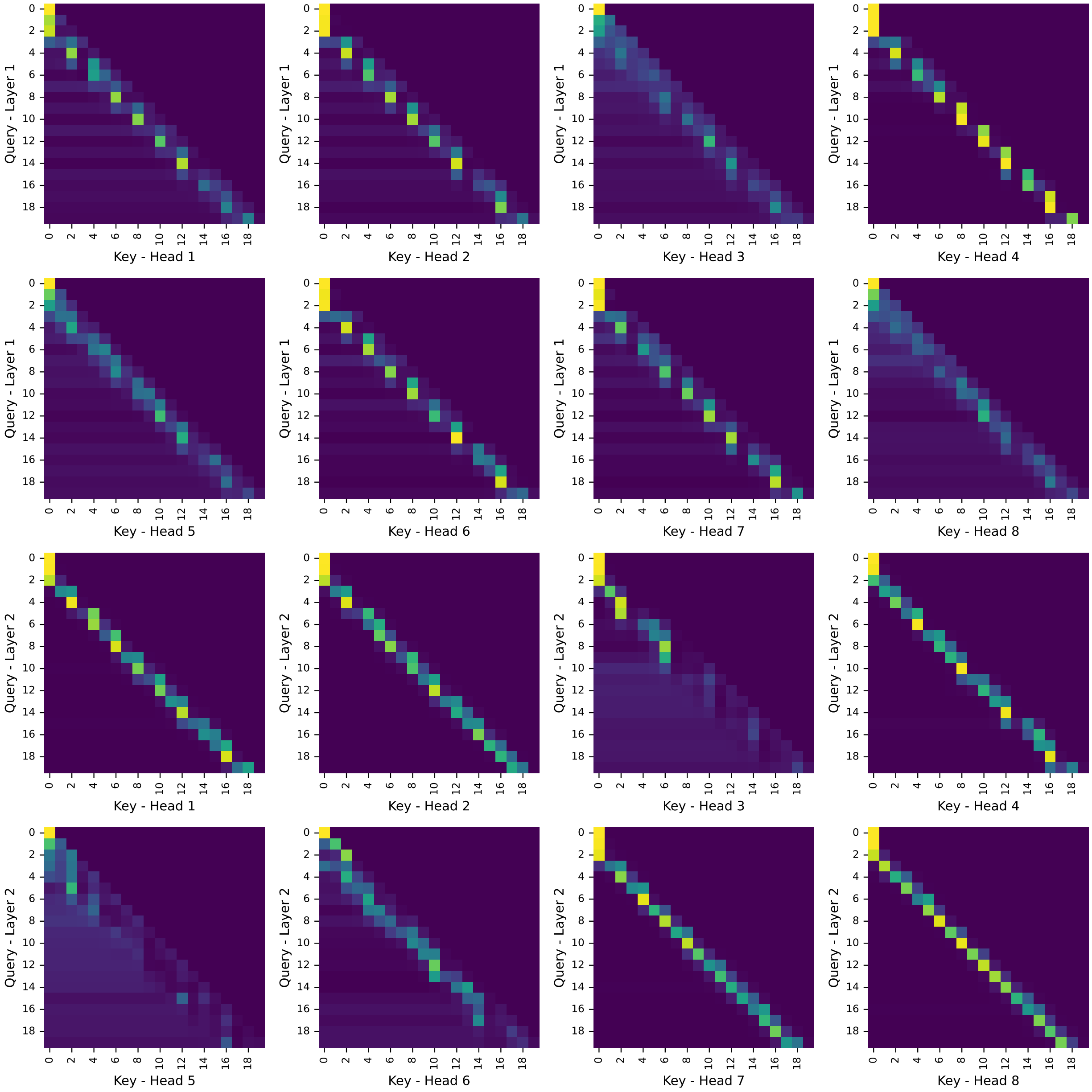

Figure 9: Gaussian adaptive attention maps for a two-layer, eight-head Transformer world model on a single Pong trajectory ($\mu = 2, \sigma = 1$). Yellow indicates high attention, while dark blue indicates no attention weight. As before, the top eight heatmaps are layer 1 and the bottom eight are layer 2. Here, each head's attention is sharply concentrated in a narrow band around a two-step look-back (the Gaussian mean), with minimal weight on distant frames. This localised focus, especially on the immediately preceding latent-state/action tokens, demonstrates how Gaussian initialisation enforces short-range dependencies, reducing attention dilution and further emphasising the sufficiency of recent timesteps for accurate state prediction and policy/value learning.

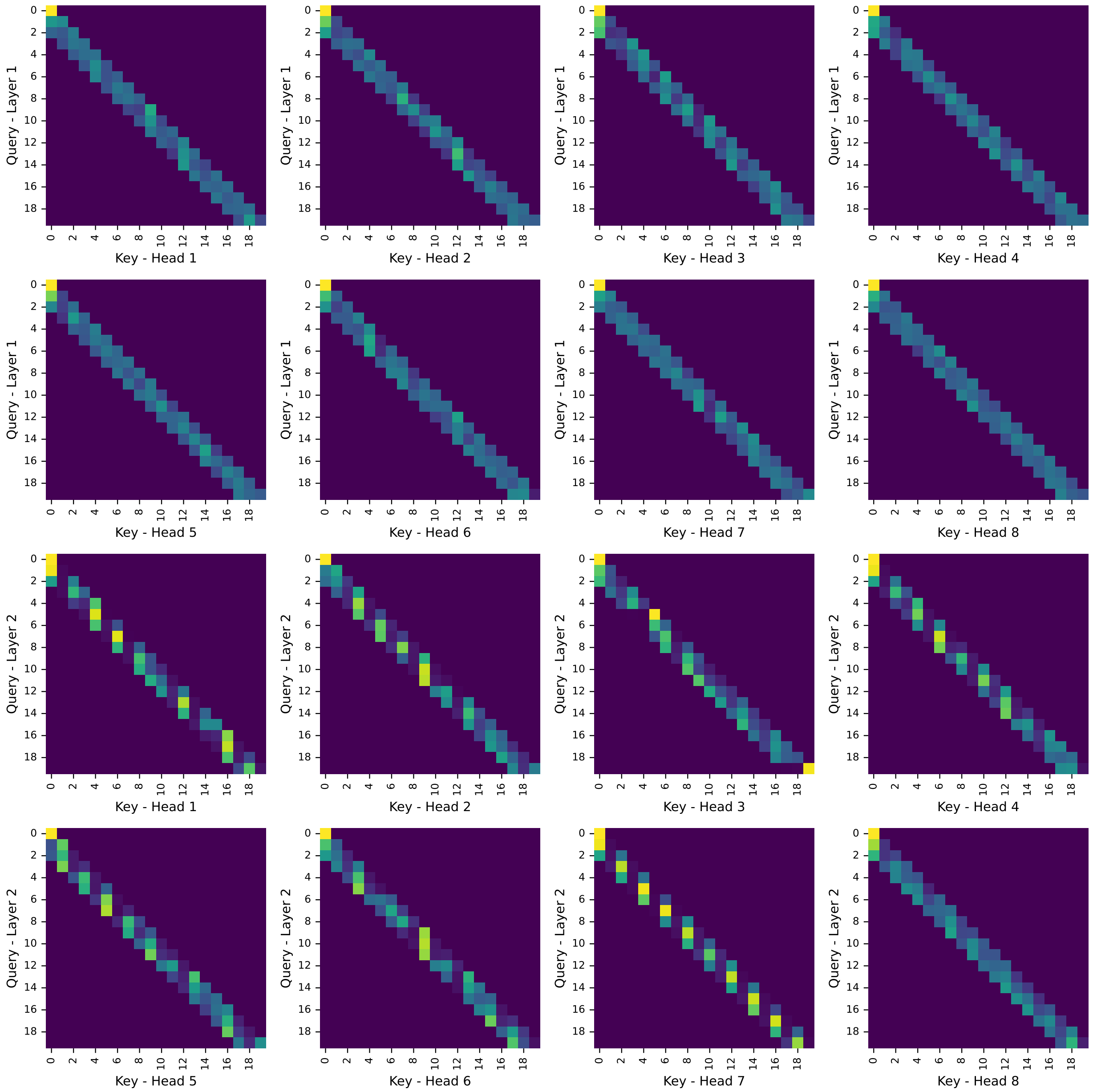

Figure 10: Local-window attention maps for a two-layer, eight-head Transformer world model on a single Pong trajectory (window size $w = 2$). Yellow indicates high attention, while dark blue indicates no attention weight. The top eight heatmaps are layer 1 heads, and the bottom eight are layer 2. Here, each query (y-axis) is allowed to attend only to keys (x-axis) within two tokens behind, producing a narrow diagonal band of width two in every head. This strict locality further enforces Pong's short-term dynamics by entirely ignoring distant frames and confirms that just the two most recent state/action pairs suffice for accurate state prediction and policy/value learning.

D World Model Analysis

In Figure 11, we examine a single Pong trajectory under the Gaussian attention world model. The top panel plots true versus predicted rewards over ten timesteps: both remain identically zero, since no scoring events occur, and it correctly maintains this constant prediction. The second panel shows the original game frames, each resized to 64×64 from the raw Atari input (note slight visual distortion). In the third panel, the predicted prior policy gradually shifts probability toward the upward “action 2” around timestep 7, but remains relatively spread across actions. The bottom panel displays the MCTS-refined policy: here, the world model sharply concentrates its probability mass on action 2, peaking near 0.6 at t=7, and the agent indeed executes this action to bounce the ball back at t=8, illustrating strong policy improvement via MCTS.

In Figure 12, we present the same Pong trajectory analysed with the local attention world model. Again, predicted and true rewards are perfectly aligned at zero throughout, accurately reflecting the absence of points. The resized image frames in the second row confirm consistent visual input. The prior policy in the third row shows a more modest buildup toward action 2, only reaching about 0.25 probability at t=7, while the MCTS policy in the fourth row amplifies this to roughly 0.35. Although this refinement is less pronounced than with Gaussian attention, the agent still selects action 2 at the critical moment (t=8) and successfully returns the ball, demonstrating that even with lower prior confidence. Local’s model supports effective short-term decision making.

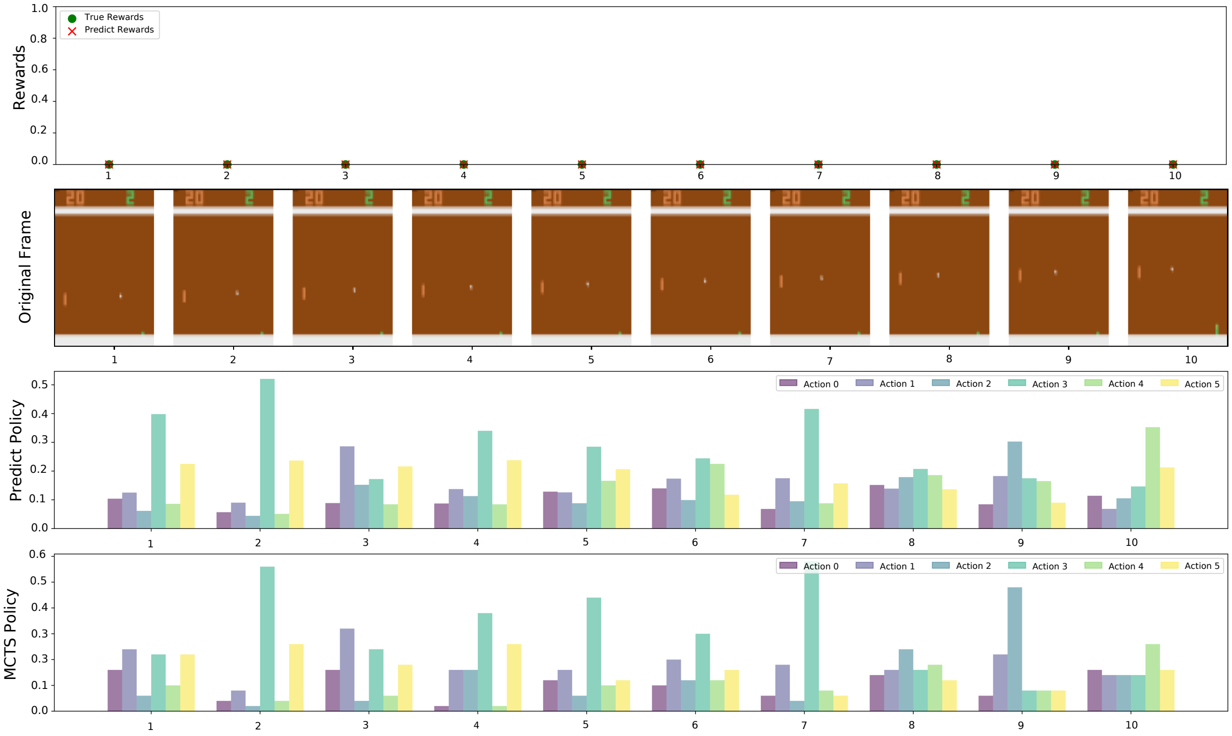

Figure 11: Predictions of the Gaussian attention world model in a single Pong trajectory are shown: in the first row, predicted and true rewards; in the second, the reconstructed image frames; in the third, the model’s prior action probabilities; and in the fourth, the policy refined by MCTS from the priors in the third row.

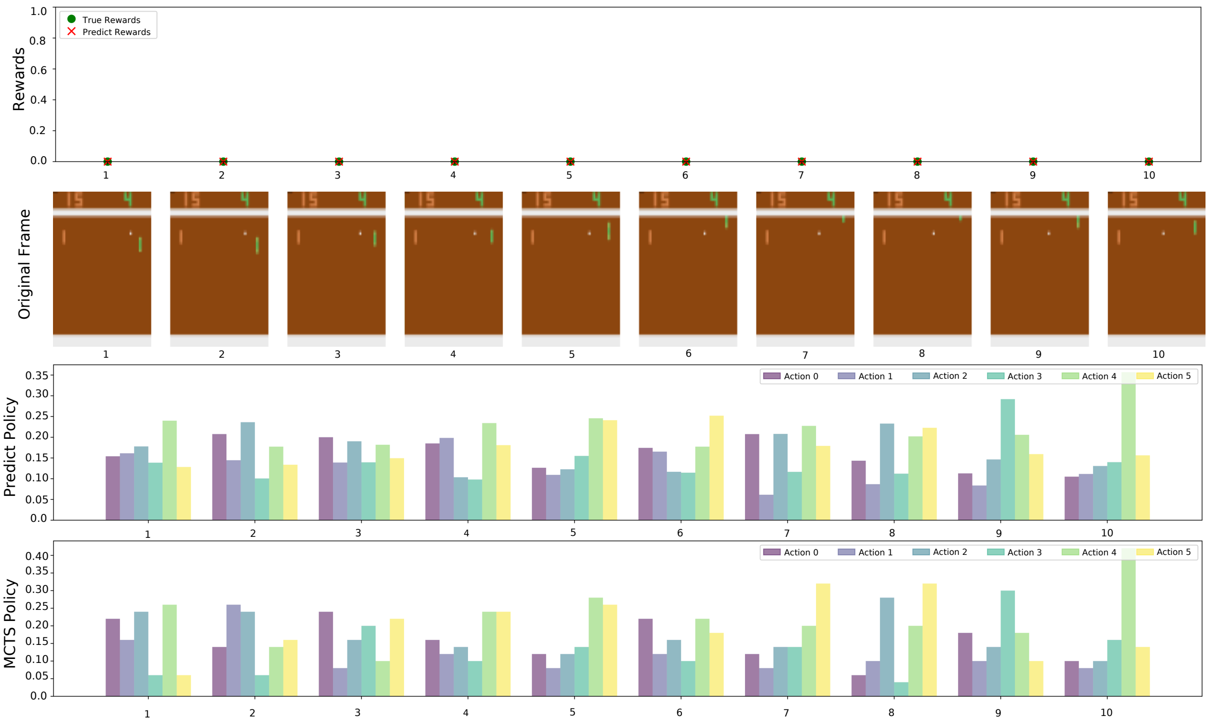

Figure 12: Predictions of the local attention world model in a single Pong trajectory are shown: in the first row, predicted and true rewards; in the second, the reconstructed image frames; in the third, the model's prior action probabilities; and in the fourth, the policy refined by MCTS from the priors in the third row.