NEXT EMBEDDING PREDICTION MAKES WORLD MODELS STRONGER

Anonymous authors
Paper under double-blind review

ABSTRACT

Capturing temporal dependencies is critical for model-based reinforcement learning (MBRL) in partially observable, high-dimensional domains. We introduce NE-Dreamer, a decoder-free MBRL agent that leverages a temporal transformer to predict next-step encoder embeddings from latent state sequences, directly optimizing temporal predictive alignment in representation space. This approach enables NE-Dreamer to learn coherent, predictive state representations without reconstruction losses or auxiliary supervision. On the DeepMind Control Suite, NE-Dreamer matches or exceeds the performance of DreamerV3 and leading decoder-free agents. On a challenging subset of DMLab tasks involving memory and spatial reasoning, NE-Dreamer achieves substantial gains. These results establish next-embedding prediction with temporal transformers as an effective, scalable framework for MBRL in complex, partially observable environments.

1 INTRODUCTION

Model-based reinforcement learning (MBRL) from high-dimensional observations hinges on learning a compact latent state that supports long-horizon prediction and control. This requirement becomes more important under partial observability: the agent must integrate information over time rather than react to a single frame. A dominant approach learns the world model with a pixel decoder, as in Dreamer, where reconstruction produces rich, control-effective features. The cost is modeling burden: reconstruction introduces a heavy generative objective, complicates optimization, and can allocate capacity to visually detailed but task-irrelevant aspects. Decoder-free methods remove the pixel decoder, training representations directly to simplify the pipeline and improve efficiency.

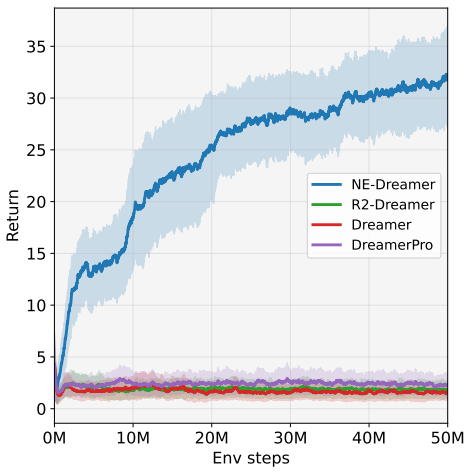

Figure 1: **DMLab Benchmark Summary.** Under matched compute and model capacity (50M environment steps; 5 seeds; 12M parameters), NE-Dreamer outperforms strong decoder-based (DreamerV3) and decoder-free world-model baselines (R2-Dreamer, DreamerPro) on the DMLab Rooms memory/navigation tasks.

However, many decoder-free objectives mainly enforce *instantaneous* (same-timestep) agreement. Under partial observability, instantaneous agreement is not enough: the representation must be *predictive across time*. Without an explicit temporal constraint, training can drift or collapse, leading to weak long-horizon structure—failure modes that surface in memory- and navigation-heavy tasks.

In this paper, we introduce NE-Dreamer, a decoder-free world model that learns by directly optimizing for *temporal predictive alignment* in its latent representations. NE-Dreamer replaces pixel-level reconstruction with a simple yet powerful objective: at each timestep, a temporal transformer predicts the *next* encoder embedding in the sequence, and this prediction is aligned to the actual next-step embedding using a redundancy-reduction metric (specifically, Barlow Twins in our implementation). By

shifting the focus from same-timestep matching to next-step prediction, NE-Dreamer learns temporally coherent latent states without the need for pixel reconstruction, data augmentation, or auxiliary regularization. As illustrated in Figure 1, this design enables NE-Dreamer to achieve substantially higher performance in partially observable DMLab environments compared to prior methods of the same model size.

Our main contributions are as follows:

1. We propose a decoder-free world-model objective based on *next-embedding prediction*, which explicitly enforces temporal predictiveness in the learned representation.
2. We integrate a lightweight causal temporal transformer into a Dreamer-style MBRL pipeline to implement next-step prediction from history within standard RSSM training.
3. We evaluate NE-Dreamer on DeepMind Control Suite and DeepMind Lab, showing strong performance on DMC and substantial gains on memory/navigation-heavy DMLab Rooms under matched compute and model size.
4. Through targeted ablations and representation diagnostics, we isolate the gains to predictive sequence modeling (causal transformer + next-step target shift) rather than reconstruction or auxiliary tricks.

2 Related Work

World models for pixel control. Latent world models aim to learn compact states that support long-horizon prediction and decision-making from high-dimensional observations. Early work demonstrated that learning dynamics in a latent space can enable planning and control by acting "in imagination" from pixels (Ha & Schmidhuber, 2018). PlaNet introduced the recurrent state-space model (RSSM) as a practical latent dynamics backbone for planning from images (Hafner et al., 2019b). Building on RSSMs, the Dreamer family trains an actor–critic on imagined rollouts in latent space via *latent imagination* (Hafner et al., 2019a; 2021; 2025). NE-Dreamer keeps this RSSM-based control backbone and changes how the latent representation is learned.

Reconstruction-based world models. A common way to learn world-model representations is to maximize an observation likelihood (pixel reconstruction), often alongside reward and termination/continuation prediction (Ha & Schmidhuber, 2018; Hafner et al., 2019b;a). Reconstruction provides dense supervision that often stabilizes optimization, but it can also allocate capacity to visually detailed factors (e.g., textures or backgrounds) that are only weakly coupled to reward. This motivates decoder-free objectives that shape the latent space directly for decision-making.

Decoder-free world models. Removing pixel reconstruction shifts the problem from modeling observations to choosing *what anchors* the latent state and *which time index* the learning signal targets. One family is *task-oriented*: latents are optimized to support reward/value prediction and planning, with supervision induced by search or TD learning, as in MuZero and TD-MPC variants (Schrittwieser et al., 2020; Hansen et al., 2022; 2024); related Dreamer-style agents also replace reconstruction with control-centric prediction objectives (e.g., MuDreamer) (Burchi & Timofte, 2024). A second family is *representation-oriented*: models predict or align learned embeddings with self-supervised objectives, sometimes across future steps (e.g., CPC, SPR) (van den Oord et al., 2018; Schwarzer et al., 2021; Paster et al., 2021) and sometimes via per-timestep invariances or clustering (Okada & Taniguchi, 2021; 2022; Deng et al., 2022; Anonymous, 2026).

For partially observable control, even strong *same-step* objectives need not make the state at time t *predictive* of what happens at $t{+}1$. NE-Dreamer belongs to the representation-oriented family but makes this temporal requirement explicit: a causal sequence model predicts the *next* encoder embedding from history and aligns it to a stop-gradient target, turning representation learning into *causal next-step prediction* rather than per-timestep agreement.

Representation prediction and collapse prevention. Predicting future embeddings is an increasingly popular alternative to reconstruction in self-supervised learning. For instance, NEPA applies next-embedding prediction with stop-gradient targets (Xu et al., 2025), while I-JEPA and data2vec focus on masked prediction and context modeling (Assran et al., 2023; Baevski et al., 2022). A

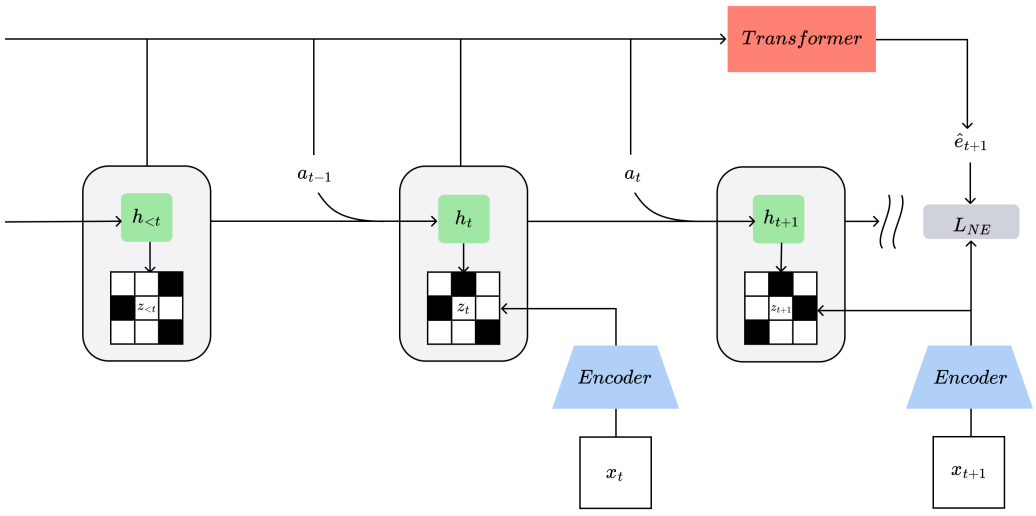

Figure 2: **Method overview.** NE-Dreamer keeps Dreamer's RSSM dynamics and imagination-based actor–critic, but replaces same-step pixel reconstruction with *next-embedding prediction* using a causal temporal transformer, improving long-horizon performance under partial observability.

central issue is preventing representational collapse, where the learned state becomes degenerate. In reinforcement learning, invariance via augmentations is a common stabilizer (Laskin et al., 2020; Kostrikov et al., 2020), and benchmarks such as the Distracting Control Suite (Stone et al., 2021) make this explicit. Bootstrapping and redundancy-reduction regularizers—like those used in BYOL, SimSiam, Barlow Twins, or VICReg (Grill et al., 2020; Chen & He, 2021; Zbontar et al., 2021; Bardes et al., 2021)—can also prevent collapse without negatives, but are usually applied to paired views at the same timestep.

NE-Dreamer generalizes these ideas to a predictive context: its causal sequence model produces a forecasted embedding e_{t+1} from history, which is aligned (with, e.g., a Barlow Twins loss) to a stop-gradient target. This enforces temporal coherence in the latent space, extending redundancy reduction to future prediction rather than just within-frame invariance.

3 Method

3.1 Problem Setup

We study partially observable control from pixels. At time t, the environment emits an image observation x_t. The agent selects an action a_t and receives a reward r_t. We also use a continuation indicator $c_t \in \{0, 1\}$, where $c_t = 1$ if the episode continues from t to $t+1$ and $c_t = 0$ on terminal transitions.

NE-Dreamer follows the standard Dreamer pipeline—(i) learn a latent world model from experience, and (ii) train an actor–critic on imagined rollouts in latent space—but **changes the representation objective** for the world model. Specifically, it removes pixel reconstruction and instead **predicts the next-step encoder embedding**. Using only information available up to time t, the model predicts $\hat{e}_{t+1}$ and aligns it to a stop-gradient target with a self-supervised loss (Barlow Twins in our instantiation).

3.2 LATENT WORLD MODEL (RSSM)

We build on a recurrent state-space model (RSSM) with a deterministic recurrent state h_t and a stochastic latent z_t.

Encoder and latent inference. An encoder maps observations to embeddings:

$$e_t \;=\; f_{\mathrm{enc}}(x_t). \tag{1}$$

Given the previous latent state and the previous action, the RSSM updates its deterministic state:

$$h_t \;=\; f_{\mathrm{rec}}(h_{t-1}, z_{t-1}, a_{t-1}). \tag{2}$$

It then defines a prior and posterior over the stochastic latent:

$$p_\phi(z_t \mid h_t), \qquad q_\phi(z_t \mid h_t, e_t). \tag{3}$$

During world-model training we sample $z_t \sim q_\phi(z_t \mid h_t, e_t)$; during imagination we sample $\hat{z}_t \sim p_\phi(z_t \mid h_t)$.

Reward and continuation heads. As in Dreamer, the world model predicts reward and continuation:

$$p_\phi(r_t \mid h_t, z_t), \qquad p_\phi(c_t \mid h_t, z_t). \tag{4}$$

Standard Dreamer also predicts observations via a pixel decoder $p_\phi(x_t \mid h_t, z_t)$. NE-Dreamer removes this decoder and replaces it with the next-embedding objective in Sec. 3.3.

World-model objective. The world model is trained with reward and continuation likelihoods, a prior–posterior regularizer, and the proposed next-embedding loss:

$$\mathcal{L}_{\mathrm{wm}} \;=\; \mathcal{L}_{\mathrm{rew}} + \mathcal{L}_{\mathrm{cont}} + \beta_{\mathrm{kl}}\,\mathcal{L}_{\mathrm{kl}} + \beta_{\mathrm{ne}}\,\mathcal{L}_{\mathrm{NE}}. \tag{5}$$

The prediction losses are negative log-likelihoods:

$$\begin{aligned} \mathcal{L}_{\mathrm{rew}} &= -\mathbb{E}[\log p_\phi(r_t \mid h_t, z_t)]\,, \\ \mathcal{L}_{\mathrm{cont}} &= -\mathbb{E}[\log p_\phi(c_t \mid h_t, z_t)]\,. \end{aligned} \tag{6}$$

The KL term regularizes the posterior toward the prior:

$$\mathcal{L}_{\mathrm{kl}} \;=\; \mathbb{E}\Big[\mathrm{KL}\big(q_\phi(z_t \mid h_t, e_t) \,\|\, p_\phi(z_t \mid h_t)\big)\Big]. \tag{7}$$

We adopt standard Dreamer stabilizers for $\mathcal{L}_{\mathrm{kl}}$ (e.g., KL balancing / free-nats); details follow prior Dreamer practice.

3.3 NEXT-EMBEDDING PREDICTIVE ALIGNMENT

NE-Dreamer trains the latent dynamics to be predictive in representation space: from history up to time t, it predicts the encoder embedding of the next observation and aligns the prediction to a stop-gradient target.

Causal next-embedding predictor. A causal temporal transformer T_θ (with a causal mask) uses only information available up to time t to produce a next-step embedding prediction:

$$\hat{e}_{t+1} \;=\; T_\theta\big(h_{\le t}, z_{\le t}, a_{\le t}\big). \tag{8}$$

The target is the next-step encoder embedding:

$$e^\star_{t+1} \;=\; \mathrm{sg}(e_{t+1}) \;=\; \mathrm{sg}\big(f_{\mathrm{enc}}(x_{t+1})\big). \tag{9}$$

We write $\mathrm{sg}(\cdot)$ for stop-gradient. Gradients flow through $\hat{e}_{t+1}$ into T_θ and the RSSM, but not through $e^\star_{t+1}$.

Alignment loss (Barlow Twins). We instantiate $\mathcal{L}_{\text{NE}}$ with a Barlow Twins redundancy-reduction objective between predicted and target embeddings. Let $\tilde{\hat{e}}_{t+1}$ and $\tilde{e}^{\star}_{t+1}$ denote embeddings normalized *per dimension* over the set of valid transitions within each minibatch (zero mean, unit variance). Let

$$\mathcal{I} \doteq \{(b,t) \,:\, c_t^{(b)} = 1\}, \qquad N \doteq |\mathcal{I}|. \tag{10}$$

The cross-correlation matrix is

$$C_{ij} \;=\; \frac{1}{N} \sum_{(b,t)\in\mathcal{I}} \tilde{\hat{e}}^{(b)}_{t+1,i}\, \tilde{e}^{\star(b)}_{t+1,j}. \tag{11}$$

The next-embedding loss is

$$\mathcal{L}_{\text{NE}} \;=\; \sum_i \big(1 - C_{ii}\big)^2 \;+\; \lambda_{\text{BT}} \sum_{i\neq j} C_{ij}^2. \tag{12}$$

This objective encourages invariance (large diagonal correlations) while discouraging redundancy (small off-diagonal correlations), here applied to *next-step* prediction rather than same-timestep matching.

3.4 Actor-Critic Learning

Like DreamerV3, NE-Dreamer learns a policy and value function in latent space by generating imagined trajectories with a world model. These imagined trajectories (of horizon $H = 15$ steps) enable efficient batch actor-critic updates. We denote the imagined full latent state as $s_t = (h_t, \hat{z}_t)$, where $\hat{z}_t \sim p_\phi(z_t \mid h_t)$. At each imagination step, actions are sampled from the policy π_θ and their values are estimated by the critic V_ψ:

$$a_t \sim \pi_\theta(a_t \mid s_t), \quad V_\psi(s_t) \approx \mathbb{E}_{p_\phi,\pi_\theta}[R_t^\lambda] \tag{13}$$

Critic: The critic predicts the distribution of λ-returns based on imagined rewards:

$$R_t^\lambda = r_t + \gamma c_t\big((1-\lambda)V_\psi(s_{t+1}) + \lambda R_{t+1}^\lambda\big) \tag{14}$$

$$\mathcal{L}_{\text{critic}}(\psi) = -\mathbb{E}_{p_\phi,\pi_\theta}\left[\sum_{t=1}^{H} \log p_\psi(R_t^\lambda \mid s_t)\right] \tag{15}$$

Actor: The actor maximizes normalized advantages, with S as an EMA-based scale:

$$\begin{aligned}\mathcal{L}_{\text{actor}}(\theta) = \; -\mathbb{E}_{p_\phi,\pi_\theta}\Bigg\{ \sum_{t=1}^{H} \text{sg}\left(\frac{R_t^\lambda - V_\psi(s_t)}{\max(1,S)}\right) \log \pi_\theta(a_t \mid s_t) \\ + \eta\, \mathcal{H}[\pi_\theta(a_t \mid s_t)]\Bigg\}\end{aligned} \tag{16}$$

Here, $\text{sg}(\cdot)$ denotes the stop-gradient operator and η is the entropy regularization coefficient.

Policy gradients are backpropagated through the world model for continuous actions. The learning procedure and all hyperparameters match DreamerV3, ensuring that observed gains stem from the representation learning objective.

4 Experiments

We evaluate whether next-embedding prediction improves long-horizon control under partial observability. We structure the results around three claims: **(C1)** NE-Dreamer improves memory/navigation performance on DMLab Rooms; **(C2)** the gains come from *predictive* sequence modeling (causal transformer + next-step target shift); and **(C3)** removing reconstruction does not degrade standard continuous control on DMC. Figure 3 (C1), Figure 4 (C2), and Figure 5 (C3) provide the headline evidence.

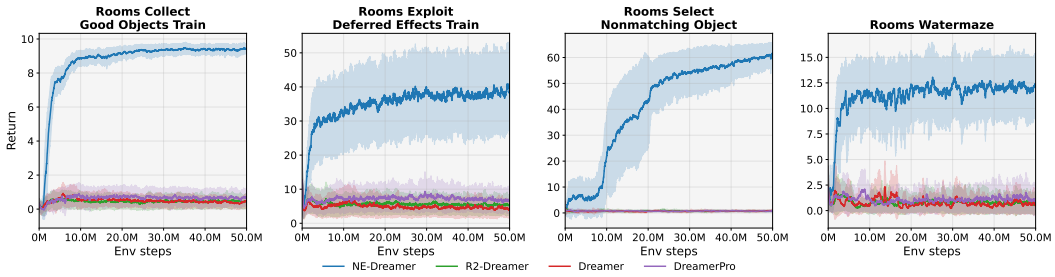

Figure 3: **DMLab Rooms: improved long-horizon memory/navigation.** Under matched compute and model capacity (50M environment steps; 5 seeds; 12M parameters), NE-Dreamer outperforms strong decoder-based (DreamerV3) and decoder-free world-model baselines (R2-Dreamer, DreamerPro) on four Rooms tasks. The largest gains occur when success depends on maintaining state over long horizons rather than reacting to short-lived visual cues.

4.1 Experimental setup

Benchmarks and tasks. We evaluate all methods on two widely used RL benchmarks:

- **DeepMind Lab (DMLab)** (Beattie et al., 2016) is a suite of first-person 3D navigation tasks designed to test partial observability, long-horizon credit assignment, and memory. Our evaluation targets four challenging "Rooms" tasks that require agents to integrate information over time and reason about spatial layouts.
- **DeepMind Control Suite (DMC)** (Tunyasuvunakool et al., 2020) is a standard benchmark for pixel-based continuous control in robotics-inspired environments. It is widely used to compare model-based RL methods, and recent advances have reached near-ceiling performance on many tasks.

Compared methods. We benchmark NE-Dreamer against representative state-of-the-art agents from three families:

- *Decoder-based world models:* **DreamerV3**: trains latent dynamics and policy using pixel-level reconstruction as the main representation objective.
- *Decoder-free world models:*
  - **R2-Dreamer** removes the pixel decoder and replaces reconstruction with a redundancy reduction loss (Barlow Twins) applied at the same timestep, enforcing agreement between encoder and latent via a lightweight projector.
  - **DreamerPro** adopts a decoder-free design but uses strong data augmentations (random image shifts) to avoid representation collapse and enforce invariance.
  - **Dreamer (no reconstruction)**- a special Dreamer variant that omits pixel reconstruction entirely, relying solely on reward, continuation, and KL objectives. This baseline tests the effect of removing explicit representation learning signals on the world model.
- *Model-free reference:* **DrQv2**: a strong pixel-based model-free RL agent that leverages strong data augmentation and direct policy/value learning from observations, providing a competitive non-model-based baseline.

All agents, including NE-Dreamer, baselines, and DrQv2 (which uses its official implementation), are evaluated under identical conditions: world-model methods share a unified PyTorch R2-Dreamer codebase with matched capacity (12M parameters, Dreamer-S architecture), while all agents undergo the same training protocol (50M environment steps on DMLab, 1M on DMC) across five random seeds. Results are reported as mean ± standard deviation; full architectural, hyperparameter, and reproducibility details appear in Appendix A.

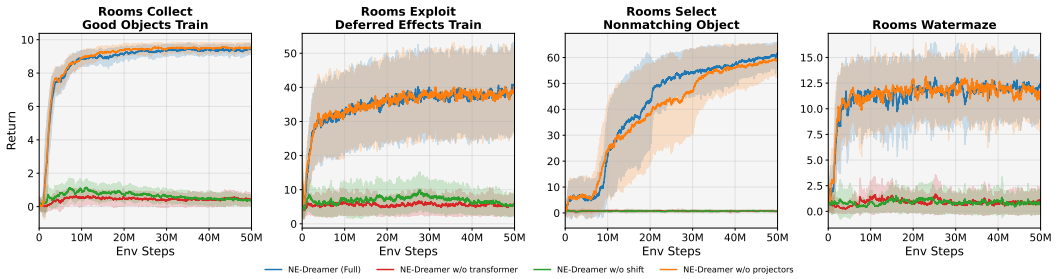

Figure 4: **Mechanism on DMLab Rooms: predictive sequence modeling is the key.** Under matched compute and model capacity (50M environment steps; 5 seeds; mean±std), removing the causal temporal transformer (*w/o transformer*) or removing the next-step target shift (*w/o shift*) substantially reduces performance. Removing the lightweight projector (*w/o projector*) mainly affects optimization speed/stability, with smaller impact on final returns.

4.2 DMLab Rooms: long-horizon memory and navigation (C1)

The DMLab Rooms benchmark directly targets the core challenge for model-based RL agents: reasoning over long temporal horizons in environments with sparse rewards and high partial observability. In these tasks, agents must integrate information across time, remember key scene elements, and plan multi-step behaviors—conditions under which standard per-timestep objectives often fail.

Figure 3 presents the per-task learning curves. Across all four tasks, NE-Dreamer delivers a dramatic improvement in returns—learning reliably and achieving substantially higher final performance than all baseline methods.

These results underscore two main strengths of NE-Dreamer:

- **Superior temporal representation:** The use of next-embedding prediction with a temporal transformer enables the agent to maintain stable, predictive state representations over long horizons, a property directly reflected in its ability to solve complex spatial memory tasks.
- **Efficiency without extra complexity:** NE-Dreamer achieves these gains without pixel-level reconstruction, heavy data augmentation, or additional domain-specific tuning. All methods operate under identical architecture and training budgets, highlighting the effectiveness of our approach rather than differences in model capacity or optimization.

4.3 DMLab Rooms ablations: isolating the mechanism (C2)

To isolate the key contributors to NE-Dreamer's performance, we systematically ablate three architectural and objective choices, keeping the rest of the pipeline strictly unchanged. The results, shown in Figure 4, highlight the critical importance of both the temporal transformer and the next-step prediction target.

No transformer: When the temporal transformer is removed, the model regresses to using a simple feedforward or shallow architecture for sequence modeling. As shown by the red curve, performance collapses on all tasks, highlighting that causal sequence modeling is indispensable for partially observable environments. The agent fails to maintain useful temporal state, suggesting that the transformer's sequence modeling capacity is important in this regime.

No next-step shift: Here, the model is trained to match the current-step embedding (as in most bootstrapped or instantaneous self-supervised objectives), rather than to predict the next-step target, while maintaining temporal transformer. This ablation demonstrates a nearly complete loss of the gains seen in the full method. The result points directly to the need for temporal prediction—not merely matching or reconstructing current observations, but explicitly encouraging the model to anticipate future latent structure.

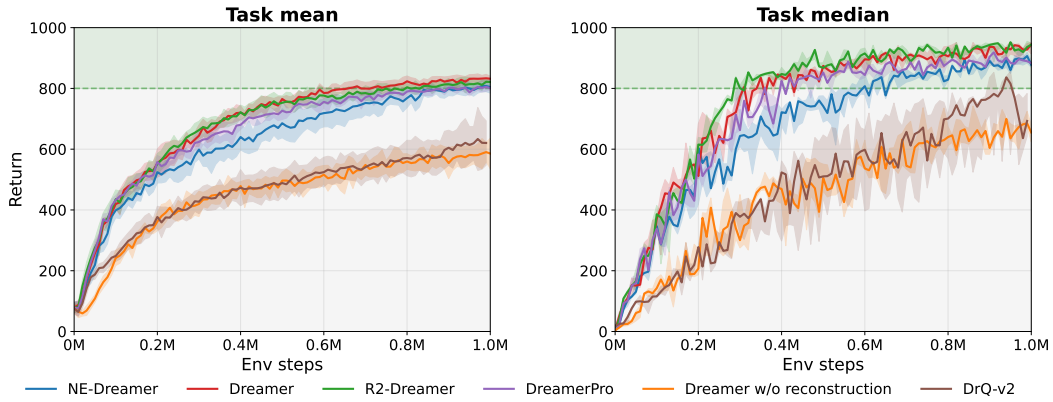

Figure 5: **DMC: removing reconstruction does not hurt standard control.** On near-saturated pixel-based continuous-control benchmarks, NE-Dreamer matches or slightly exceeds strong decoder-based (DreamerV3) and decoder-free world-model baselines (R2-Dreamer, DreamerPro) under a unified protocol (1M environment steps; 5 seeds; 12M parameters). Per-task learning curves can be found in Appendix B

No projector: The lightweight projection head before transformer is removed in this setting. Such a change leads to only a minor reduction in asymptotic performance. This suggests that while the projector may aid optimization, by smoothing the alignment objective or improving conditioning, it is not fundamentally responsible for the observed gains.

Together, these ablations show that NE-Dreamer's core mechanism is the combination of a causal temporal transformer and a next-step prediction objective. The model's success is not due to auxiliary tricks or architectural tweaks, but to its direct enforcement of temporal predictive alignment over latent trajectories.

4.4 DMC: NO REGRESSION WITHOUT RECONSTRUCTION (C3)

We include DMC as a calibration point. Under the unified protocol, NE-Dreamer matches DreamerV3 and competitive decoder-free baselines (Figure 5), supporting the practical takeaway that replacing reconstruction with next-embedding prediction improves the hard regime (DMLab) *without* sacrificing standard continuous-control performance.

4.5 REPRESENTATION DIAGNOSTICS

To interpret what information is encoded in the learned latent state, we perform a lightweight diagnostic: we train a post-hoc pixel decoder to reconstruct observations from frozen latent representations. Importantly, this decoder is *not* used during agent training and serves only as an analysis tool.

As shown in Figure 6, NE-Dreamer's latent representations enable reconstructions that preserve object identity, spatial layout, and task-relevant features consistently across time. In contrast, decoder-based Dreamer and decoder-free R2-Dreamer exhibit a characteristic failure mode: task-specific attributes (e.g., the relevant object in a room) may be present in one timestep but disappear or degrade in subsequent latents, even when the underlying scene has not changed.

NE-Dreamer's next-embedding prediction objective enforces temporal stability by training the world model to predict the *next encoder embedding* from history, which encourages the latent state to retain information that is predictive of what comes next. In contrast, same-timestep reconstruction or alignment objectives can allow latent drift toward transient visual details. Consequently, NE-Dreamer learns representations that prioritize persistent, decision-relevant structure, making it better suited for memory, planning, and long-horizon control.

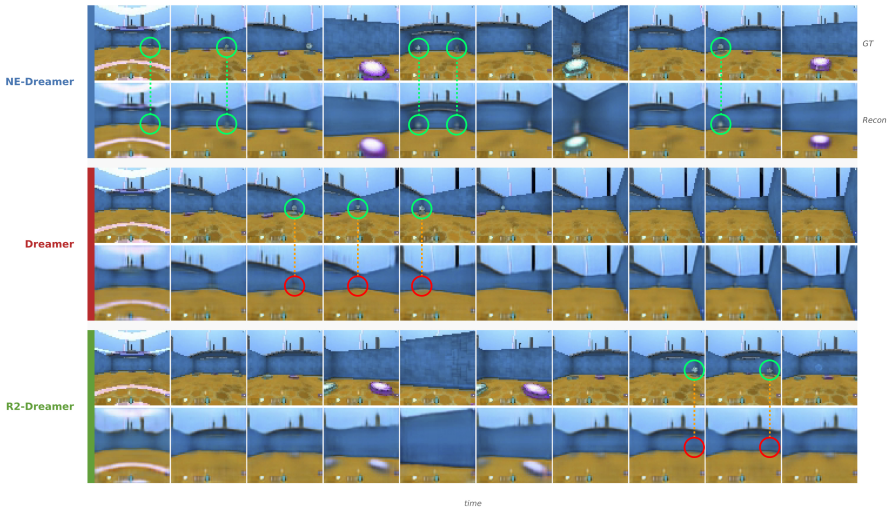

Figure 6: **Post-hoc decoder reconstruction reveals temporal consistency.** Rows show ground-truth observations (GT) and reconstructions from a post-hoc decoder trained on frozen latents. NE-Dreamer preserves task-relevant objects and spatial layout consistently over time (marked green circles), while same-timestep methods (Dreamer, R2-Dreamer) exhibit temporal inconsistency, where task-specific attributes appear transiently and then fade (marked red circles).

5 Discussion

NE-Dreamer abandons pixel reconstruction in favor of direct next-embedding prediction: the model learns to predict the next encoder embedding $\hat{e}_{t+1}$ from history and aligns it to a stop-gradient target $e^{\star}_{t+1}$. We use the Barlow Twins (BT) objective to ensure stability and avoid collapse, but any alignment loss that encourages both expressiveness and non-degenerate solutions could be substituted.

A causal temporal transformer critically enables world models to compress history into only those latent features predictive of future states—yielding robustness to partial observability. Its architecture inherently supports multi-step prediction (latent overshooting), allowing efficient training of long-horizon dependencies without additional rollout cost.

NE-Dreamer delivers consistent, substantial gains on memory- and planning-intensive DMLab Rooms tasks—outperforming both decoder-free and strong decoder-based baselines at equal model size and compute. These improvements arise from temporal predictive alignment with a sequence model, not larger architectures or aggressive tuning. On standard DMC benchmarks, NE-Dreamer matches prior methods, confirming that its advantages in harder domains incur no regression elsewhere.

One limitation is that our experiments focus on environments where long-term structure, rather than fine visual detail, is the primary challenge. Whether decoder-free, prediction-based objectives can match reconstruction in high-fidelity tasks remains open. Future work should explore alternative alignment losses and test NE-Dreamer in visually complex domains.

Overall, our results establish next-embedding prediction with a causal transformer as a practical, scalable foundation for robust representation learning in model-based RL.

6 Conclusion

We presented NE-Dreamer, a decoder-free Dreamer-style agent that learns world-model representations by predicting and aligning the *next* encoder embedding using a causal temporal transformer. NE-Dreamer improves long-horizon memory/navigation in DeepMind Lab Rooms while matching strong baselines on the DeepMind Control Suite, and ablations attribute these gains to predictive sequence modeling (causal transformer and next-step target shift), not reconstruction.

REFERENCES

Anonymous. R2-Dreamer: Redundancy-reduced world models without decoders or augmentation. Manuscript under review, 2026.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael G. Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. *CVPR*, pp. 15619–15629, 2023. doi: 10.1109/cvpr52729.2023.01499. URL https://doi.org/10.1109/cvpr52729.2023.01499.

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. data2vec: A general framework for self-supervised learning in speech, vision and language. *ICML*, pp. 1298–1312, 2022. URL https://proceedings.mlr.press/v162/baevski22a.html.

Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *CoRR*, abs/2105.04906, 2021. URL https://arxiv.org/abs/2105.04906.

Charles Beattie, Joel Z Leibo, Denis Teplyashin, Tom Ward, Marcus Wainwright, Heinrich Küttler, Andrew Lefrancq, Simon Green, Víctor Valdés, Amir Sadik, et al. Deepmind lab. *arXiv preprint arXiv:1612.03801*, 2016.

Maxime Burchi and Radu Timofte. Mudreamer: Learning predictive world models without reconstruction. *CoRR*, abs/2405.15083, 2024. doi: 10.48550/arxiv.2405.15083. URL https://doi.org/10.48550/arxiv.2405.15083.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *CVPR*, pp. 15750–15758, 2021. doi: 10.1109/cvpr46437.2021.01549. URL https://doi.org/10.1109/cvpr46437.2021.01549.

Fei Deng, Ingook Jang, and Sungjin Ahn. Dreamerpro: Reconstruction-free model-based reinforcement learning with prototypical representations. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pp. 4956–4974, 2022. URL https://proceedings.mlr.press/v162/deng22a.html.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *CoRR*, abs/2006.07733, 2020. URL https://arxiv.org/abs/2006.07733.

David Ha and Jürgen Schmidhuber. World models. *CoRR*, abs/1803.10122, 2018. URL http://arxiv.org/abs/1803.10122.

Danijar Hafner, Timothy P. Lillicrap, Jimmy Ba, and Mohammad Norouzi 0002. Dream to control: Learning behaviors by latent imagination. *CoRR*, abs/1912.01603, 2019a. URL http://arxiv.org/abs/1912.01603.

Danijar Hafner, Timothy P. Lillicrap, Mohammad Norouzi, and Jimmy Ba. Learning latent dynamics for planning from pixels. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2555–2565, 2019b. URL https://proceedings.mlr.press/v97/hafner19a.html.

Danijar Hafner, Timothy P. Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://arxiv.org/abs/2010.02193.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy P. Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025. doi: 10.1038/S41586-025-08744-2. URL https://doi.org/10.1038/s41586-025-08744-2.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=Oxh5CstDJU.

Nicklas A. Hansen, Hao Su, and Xiaolong Wang. Temporal difference learning for model predictive control. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pp. 8387–8406, 2022. URL https://proceedings.mlr.press/v162/hansen22a.html.

Ilya Kostrikov, Denis Yarats, and Rob Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. *CoRR*, abs/2004.13649, 2020. URL https://arxiv.org/abs/2004.13649.

Michael Laskin, Kimin Lee, Adam Stooke, Lerrel Pinto, Pieter Abbeel, and Aravind Srinivas. Reinforcement learning with augmented data. *CoRR*, abs/2004.14990, 2020. URL https://arxiv.org/abs/2004.14990.

Masashi Okada and Tadahiro Taniguchi. Dreaming: Model-based reinforcement learning by latent imagination without reconstruction. *ICRA*, pp. 4209–4215, 2021. doi: 10.1109/icra48506.2021.9560734. URL https://doi.org/10.1109/icra48506.2021.9560734.

Masashi Okada and Tadahiro Taniguchi. Dreamingv2: Reinforcement learning with discrete world models without reconstruction. *IROS*, pp. 985–991, 2022. doi: 10.1109/iros47612.2022.9981405. URL https://doi.org/10.1109/iros47612.2022.9981405.

Keiran Paster, Kyle McKinney, Sheila McIlraith, and Jimmy Ba. Blast: Latent dynamics models from bootstrapping. In *NeurIPS 2021 Deep Reinforcement Learning Workshop*, 2021. URL https://openreview.net/forum?id=VwA_hKnX_kR.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588 (7839):604–609, 2020. doi: 10.1038/s41586-020-03051-4. URL https://doi.org/10.1038/s41586-020-03051-4.

Max Schwarzer, Ankesh Anand, Rishab Goel, R Devon Hjelm, Aaron Courville, and Philip Bachman. Data-efficient reinforcement learning with self-predictive representations. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://openreview.net/forum?id=uCQfPZwRaUu.

Austin Stone, Oscar Ramirez, Kurt Konolige, and Rico Jonschkowski. The distracting control suite - a challenging benchmark for reinforcement learning from pixels. *CoRR*, abs/2101.02722, 2021. URL https://arxiv.org/abs/2101.02722.

Saran Tunyasuvunakool, Alistair Muldal, Yotam Doron, Siqi Liu, Steven Bohez, Josh Merel, Tom Erez, Timothy Lillicrap, Nicolas Heess, and Yuval Tassa. dm_control: Software and tasks for continuous control. *Software Impacts*, 6:100022, 2020. ISSN 2665-9638. doi: 10.1016/j.simpa.2020.100022. URL https://www.sciencedirect.com/science/article/pii/S2665963820300099.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. URL https://arxiv.org/abs/1807.03748.

Sihan Xu, Ziqiao Ma, Wenhao Chai, Xuweiyi Chen, Weiyang Jin, Joyce Chai, Saining Xie, and Stella X. Yu. Next-embedding prediction makes strong vision learners. *arXiv preprint arXiv:2512.16922*, 2025. URL https://arxiv.org/abs/2512.16922.

Jure Zbontar, Li Jing, Ishan Misra, et al. Barlow twins: Self-supervised learning via redundancy reduction. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12310–12320, 2021. URL https://proceedings.mlr.press/v139/zbontar21a.html.

A TECHNICAL DETAILS

Table A summarizes the primary hyperparameters used in this study. These settings are primarily based on those of DreamerV3, with minimal modifications related to the proposed representation learning objective.

Table 1: Main hyperparameters. Our settings are identical to DreamerV3 unless otherwise noted. All method-based hyperparameters identical to original implementations too.

Parameter	Symbol	Setting
General		
Replay Buffer Capacity	—	5×10^6
Batch Size	B	16
Batch Length	T	64
Optimizer	—	Adam
Activation	—	RMSNorm + SiLU
Model Size	—	DreamerV3 Small
Input Image Resolution	—	64×64 RGB
Replayed Steps per Policy Step	—	512 (DMC) / 32 (DMLab)
Environment Instances	—	16 (DMC) / 16 (DMLab)
Action Repeat	—	2 (DMC) / 4 (DMLab)
World Model		
Number of Latents	—	32
Classes per Latent	—	32
Prediction Loss Scale	β_{pred}	1.0
Dynamics Loss Scale	β_{dyn}	1.0
Representation Loss Scale	β_{rep}	0.1
Learning Rate	α	4×10^{-5}
Adam Betas	β_1, β_2	0.9, 0.999
Adam Epsilon	ϵ	1×10^{-20}
Gradient Clipping	AGC	0.3
Slow Value Momentum	—	0.02
Model Return Lambda	λ	0.95
Actor Critic		
Imagination Horizon	H	15
Discount	γ	0.85
Return Lambda	λ	0.95
Critic EMA Decay	—	0.98
Critic EMA regularizer	—	1.0
Return Normalization Percentiles	—	5^{th} and 95^{th}
Return Normalization Decay	—	0.99
Actor Entropy Scale	η	3×10^{-4}
Learning Rate	α	4×10^{-5}
Adam Betas	β_1, β_2	0.9, 0.999
Adam Epsilon	ϵ	1×10^{-20}
Gradient Clipping	AGC	0.3
Per method loss weight		
Dreamerv3 reconstruction loss	—	1.0
NE-Dreamer next-embedding loss	—	1.0
R2-Dreamer barlow-twins loss	—	0.05
DreamerPro SwAV loss	—	1.0
NE-Dreamer transformer configuration		
hidden dim	—	256
num layers	—	2
num heads	—	4
Additional hyperparameters	—	
BT redundancy λ	—	5×10^{-4}

B DMC DETAILED RESULTS

Figure B the individual learning curves for all 20 tasks in the DMC benchmark.

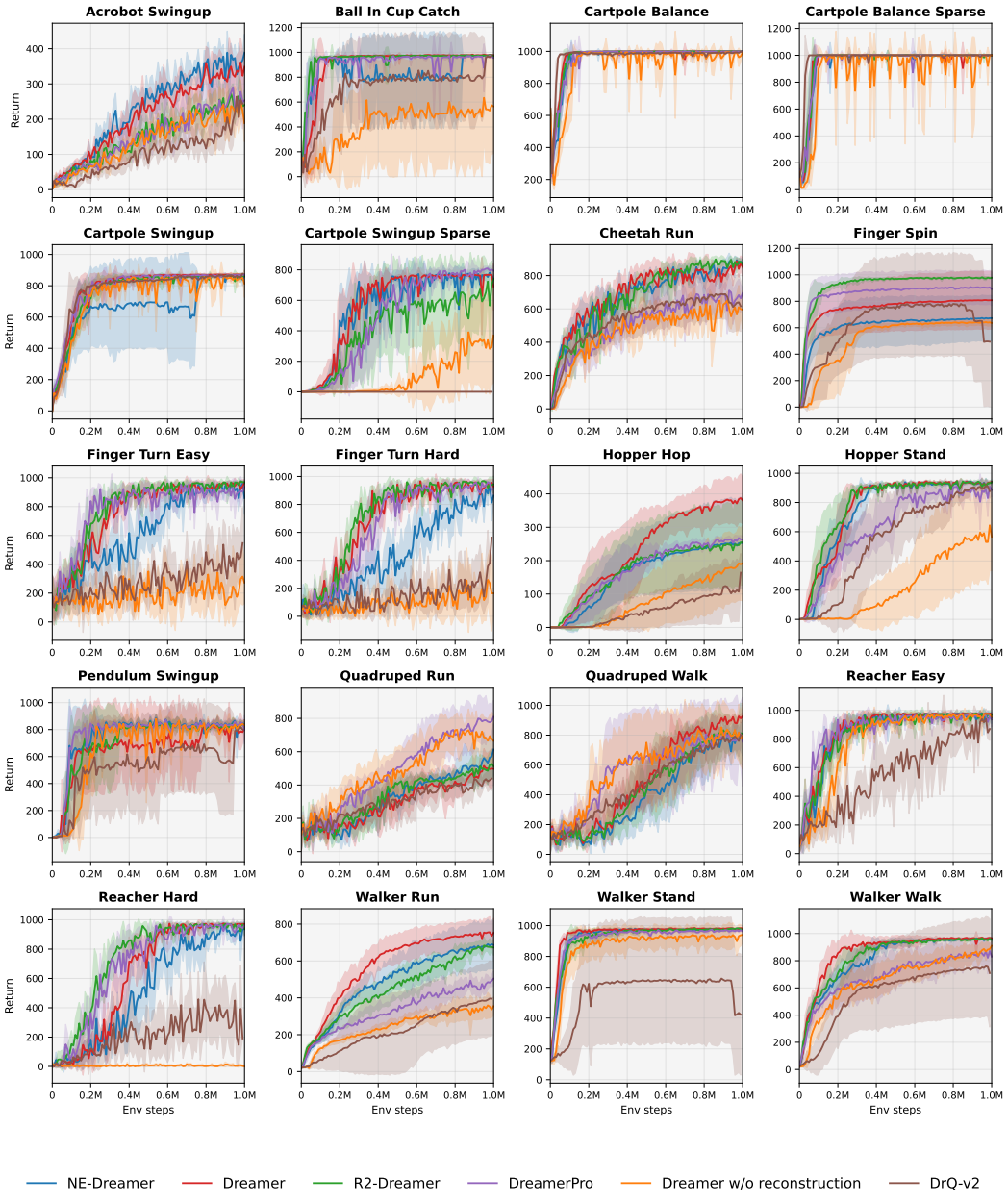

Figure 7: Per-task learning curves for all 20 DMC tasks