2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

# $E^2DT$: Efficient and Effective Decision Transformer with Experience-Aware Sampling for Robotic Manipulation

Kaiyan Zhao[1], Borong Zhang[2], Yiming Wang[2], Xingyu Liu[1], Xuetao Li[1], Yuyang Chen[2], Xiaoguang Niu[1*]

***Abstract*— In reinforcement learning (RL) for robotic manipulation, the Decision Transformer (DT) has emerged as an effective framework for addressing long-horizon tasks. However, DT's performance depends heavily on the coverage of collected experiences. Without an active exploration mechanism, standard DT relies on uniform replay, which leads to poor sample efficiency, limited exploration, and reduced overall effectiveness. At the same time, while excessive exploration can help avoid local optima, it often delays policy convergence and leads to degraded efficiency. To address these limitations, we propose E$^2$DT, a DT-guided k-Determinantal Point Process sampling framework that enables the model to actively shape its own experience selection. Our framework is experience-aware, allowing E$^2$DT to be both efficient, by prioritizing sampling quality (e.g., high-return, high-uncertainty, and underrepresented trajectories), and effective, by ensuring diversity across trajectory windows to preserve policy optimality. Specifically, DT's internal latent embeddings measure diversity across trajectory windows, while quality is quantified through a composite metric that integrates return-to-go (RTG) quantiles, predictive uncertainty, and stage coverage (inverse frequency). These two dimensions are integrated into a novel quality–diversity joint kernel that prioritizes the most informative experiences, thereby enabling learning that is both efficient and effective. We evaluate E$^2$DT on challenging robotic manipulation benchmarks in both simulation and real-robot settings. Results show that it consistently outperforms prior methods. These findings demonstrate that coupling policy learning with experience-aware sampling provides a principled path toward robust long-horizon robotic learning.**

## I. Introduction

Autonomous decision-making and control in robotics [1], [2] have long been recognized as a fundamental challenge. The central goal is to enable robots to autonomously plan critical task states and execute complex tasks with high efficiency, approaching human-level capabilities. Achieving such intelligence requires robots not only to interpret the semantics of task instructions but also to infer key states from feedback signals and make informed decisions accordingly. Among various learning paradigms, Reinforcement Learning (RL) [3], [4] has shown remarkable promise for robotic task planning and control, allowing agents to master tasks of diverse complexity. Despite this progress, existing RL approaches still encounter major obstacles, particularly in long-term dependency modeling [5], [6], exploration efficiency [7]–[10], and sample utilization [11]–[13], which collectively hinder scalability to real-world robotic applications.

[1] School of Computer Science, Wuhan University, Wuhan 430072, China
[2] State Key Lab of Internet of Things for Smart City and Department of Computer Information Science, University of Macau, Macao 999078, China
[*] indicates corresponding authors: XG.Niu (xgniu@whu.edu.cn)

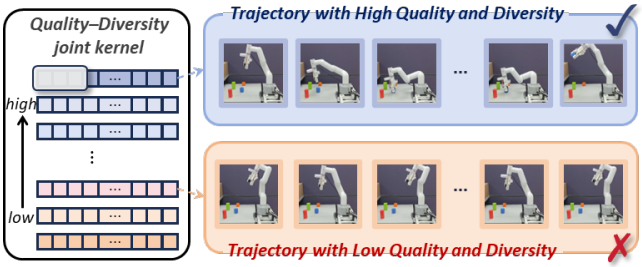


Fig. 1: Our method prioritizes trajectories that jointly achieve high quality and high diversity, moving beyond simple heuristics. By scoring trajectories with a quality–diversity measure, the sampler ensures that training focuses on the most informative experiences.

To address long-horizon dependencies [14], [15], sequence modeling approaches, most notably the Decision Transformer (DT) [16], have emerged as a powerful paradigm. By conditioning action selection on the expected return-to-go, DT effectively captures long-horizon task structures. However, as a passive learner, its effectiveness is limited by the coverage and quality of the training data. When paired with uniform experience replay, DT suffers from redundancy, fails to prioritize high-value trajectories, and ultimately delivers poor performance.

This motivates a key insight: *maximizing DT's learning efficiency requires breaking its passive data-receiving mode and equipping it with an active, model-cooperative experience selection mechanism*. Such a mechanism must extend beyond short-sighted heuristics driven by instantaneous errors and instead evaluate the usefulness of experience from a long-term perspective. An ideal sampler should balance two orthogonal objectives: quality, meaning prioritizing experiences that directly refine the policy and carry high return potential; and diversity, meaning ensuring broad coverage of the state–action space to prevent distribution collapse and premature convergence to suboptimal solutions.

To remedy this critical deficiency, we introduce an *experience-aware* framework in which the Decision Transformer actively guides its own experience sampling, jointly optimizing trajectory quality and diversity and thereby making training both *efficient* and *effective*. We design a principled sampler based on a k-Determinantal Point Process (k-DPP) [17]–[19], driven entirely by signals derived from the DT itself. Concretely, DT's internal latent representations provide a rich metric space for assessing diversity, while quality is quantified by a composite of *return-to-go*

*(RTG) quantiles*, *predictive uncertainty*, and *stage coverage* (inverse-frequency). These two dimensions are unified into a novel quality–diversity joint kernel, which encourages the agent to learn preferentially from the most informative and policy-relevant experiences *(driving efficiency via quality and effectiveness via diversity)* (see Fig. 1). Because the k-DPP selector up-weights high-quality/rare windows, the mini-batch sampling distribution $p_{\text{sel}}(w)$ departs from the dataset/behavior distribution $p_D(w)$ (i.e., $p_{\text{sel}}(w) \neq p_D(w)$), inducing selection bias in the gradient estimates. To mitigate selection bias during training, we further adopt *debiased mixed replay* with normalized importance weighting.

We term this framework $\mathbf{E}^2\mathbf{DT}$ and integrate the intelligent sampling mechanism into the policy learning process to maximize data efficiency. Our contributions are as follows:

- We propose a learning paradigm that couples a sequence-based policy model (DT) with *experience-aware* data selection, enabling the model to shape its training distribution for *efficient and effective* learning.
- We instantiate this paradigm with a DT-guided k-DPP sampler that unifies quality and diversity through a novel joint kernel, thereby balancing exploration and exploitation in the data space.
- We conduct extensive experiments on challenging robotic manipulation benchmarks [20] and realistic settings, demonstrating substantial improvements in sample efficiency, convergence speed, and final task success rate compared with state-of-the-art methods.

## II. Preliminaries

### A. Reinforcement Learning in Robotics

Reinforcement Learning (RL) [21]–[23] is a machine learning paradigm where agents learn to make optimal decisions through interactions with the environment. In robotic control [24]–[26], RL is used for task planning and control, especially in complex robotic environments. The standard formulation of RL is a Markov Decision Process (MDP) [27], represented as $\langle \mathcal{S}, \mathcal{A}, P, R, \gamma \rangle$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P$ defines the state transition probabilities, $R$ is the reward function, and $\gamma$ is the discount factor. In robotic control tasks, the state typically includes joint states, velocities, target positions, and other relevant information, while the action represents control signals, such as joint positions or velocities. Although RL has achieved success in simpler tasks, traditional RL methods, such as DQN [28], DDPG [29], and A3C [30], face challenges in complex robotic environments, such as inefficient exploration and poor modeling of long-term dependencies. This motivates the development of new methods to improve exploration strategies and long-term decision-making.

### B. Decision Transformer for Long-Horizon Tasks

Decision Transformer (DT) [16], [31] is a reinforcement learning method based on the Transformer architecture, specifically designed to handle long-term reward dependencies. Unlike traditional RL methods, DT models long-term return optimization through return-to-go (RTG) and predict actions based on historical states and rewards. The goal of DT is to predict actions that maximize long-term return, conditioned on the RTG, as follows:

$$a_t = \text{DT}(s_t, r_t, \hat{R}_t) \tag{1}$$

where $\hat{R}_t$ represents the return-to-go starting from time step $t$, and $a_t$ is the action predicted by the DT model. DT excels in capturing dependencies across long time horizons, which is useful in complex robotic manipulation tasks, optimize long-term strategies and improving task success rates.

### C. Determinantal Point Processes (DPPs)

A Determinantal Point Process (DPP) [18] is a probabilistic model that favors selecting *diverse* subsets. In the L-ensemble form with a positive semidefinite kernel $L \in \mathbb{R}^{N\times N}$, the probability of selecting a subset $Y \subseteq \{1, \ldots, N\}$ is

$$\mathbb{P}(Y) = \frac{\det(L_Y)}{\det(L+I)} \tag{2}$$

where $L_Y$ is the principal submatrix of $L$ indexed by $Y$, and $I$ is the identity matrix of compatible size. Intuitively, when $L_{ij}$ encodes item similarity, $\det(L_Y)$ equals the (squared) volume spanned by the selected items' feature vectors, larger volumes arise when items are individually salient yet mutually dissimilar, inducing negative correlations among similar items. In our setting, using a DPP to sample trajectory windows increases replay diversity, reduces redundancy, and improves coverage in complex environments. In Sec. IV, we instantiate a *k*-DPP with a quality–diversity joint kernel to select informative training subsets.

## III. Problem Formulation

We begin with a typical long-horizon robotic manipulation task, such as Target Grasping, where success depends on a coherent sequence of sub-tasks like reaching, grasping, and rotating, as shown in Fig. 2. The challenge is not only long-horizon credit assignment but also efficiently learning key operations from trajectories that contain substantial redundancy (e.g., free-space arm motions).

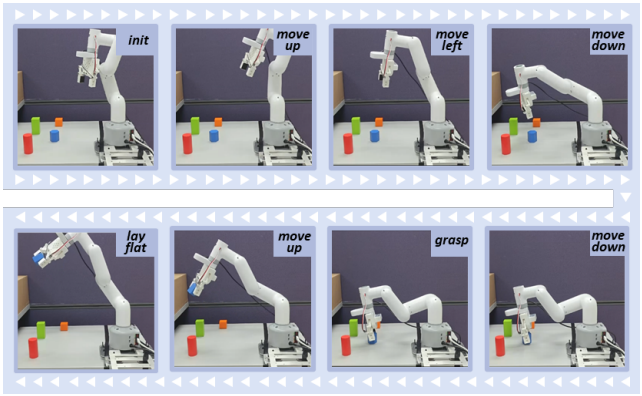


Fig. 2: Trajectory of a long-horizon manipulation task (Target Grasping). Success depends on sub-tasks like reaching, grasping, and rotating.

Our goal is to learn an efficient control policy for such tasks, modeling it as a Markov Decision Process (MDP). The policy is instantiated as a Decision Transformer (DT) $\pi_\theta$ with parameters $\theta$. In standard online learning, the agent collects new trajectories stored in a replay buffer $\mathcal{D}$, and mini-batches are drawn uniformly for training. However, this passive scheme is limited: DT is highly sensitive to the data distribution, and uniform sampling cannot distinguish the *quality* or *novelty* of experience, leading to low learning efficiency.

**Core Issue: Active Experience Selection.** We argue that the challenge is not just optimizing $\theta$, but *designing an active experience selection mechanism* that cooperates with the policy $\pi_\theta$. Instead of uniform sampling from $\mathcal{D}$, we introduce a selection function

$$g(\cdot;\,\psi):\ \mathcal{C}\ \mapsto\ \mathcal{Y} \tag{3}$$

which selects an informative subset $\mathcal{Y} \subset \mathcal{C}$ from a candidate pool $\mathcal{C} \subseteq \mathcal{D}$ for policy updates (e.g., trajectory windows of length $H$). This leads to the joint objective:

$$\min_{\theta,\ \psi}\ \ \mathbb{E}_{\mathcal{Y}\sim g(\mathcal{C};\,\psi)}[\,\mathcal{L}(\theta;\,\mathcal{Y})\,] \tag{4}$$

where $\mathcal{L}(\theta;\,\mathcal{Y})$ is the training loss of the policy on the selected subset. In practice (Sec. IV), we realize $g$ via k-DPP on a quality-diversity kernel and perform *debiased mixed replay*, mixing samples from $\mathcal{Y}$ and the global buffer $\mathcal{D}$ with importance weighting. Equation (4) *couples* policy learning (optimizing $\theta$) with data selection (optimizing $g$ via $\psi$).

**Key Objectives.** To achieve the joint optimization in Eq. (4), we address three key objectives: 1. Defining "High-Quality" Experience: For DT, high-quality experience balances two dimensions: (i) Quality, the utility of an experience for policy improvement (e.g., high-return trajectories or regions of high uncertainty); and (ii) Diversity, ensuring coverage of the state–action space to avoid suboptimal convergence due to distribution collapse. 2. Reliable Quality Signals: The active selector must leverage the policy's internal signals, avoiding ad hoc heuristics, allowing DT to identify valuable experiences based on its current knowledge. 3. Balancing Optimality and Efficiency: The selection algorithm must be both theoretically principled and computationally efficient, as finding the optimal subset from a large replay buffer is intractable.

**Summary.** Our goal is to design an active experience selection framework for DT, guided by the model's own signals, that efficiently balances quality and diversity while remaining computationally feasible, maximizing learning efficiency in long-horizon tasks.

## IV. Methodology

To address the sample inefficiency of the Decision Transformer (DT) under passive learning, we design an active experience selection framework. The core idea is to let the DT model itself guide the sampling process, thereby intelligently balancing the *quality* and *diversity* of training data. The framework consists of three stages: (i) signal extraction based on DT's internal states to define what constitutes high-value samples; (ii) construction of a quality–diversity joint kernel; and (iii) subset selection via k-DPP and debiased training.

### A. Diversity and Quality Evaluation

The central question of this stage is: *From the perspective of the current policy, what experiences are most valuable?* We posit that ideal experiences should possess both high quality (directly effective for policy improvement) and high diversity (covering a broader state–action space). To this end, we use DT's own representation capacity and predictive signals to define these two types of metrics.

**Diversity Evaluation.** To achieve effective diversity-aware sampling, we must first solve a fundamental problem: how to define and quantify the dissimilarity between trajectories. Directly computing distances in the raw physical state space (e.g., joint angles) leads to a serious pitfall: spurious diversity. For example, a robot executing the same grasp from slightly different initial poses may produce substantially different state sequences, yet remain highly redundant at the behavioral level. *Our key viewpoint is that diversity should be measured in a behavioral latent representation space rather than in the raw physical space.* This latent space is induced by the DT, where distances reflect behavioral intent rather than low-level kinematics. To realize this, we leverage the DT model itself. The encoder $f_\theta$ of DT, trained to understand long-horizon context for decision making, has learned to abstract a trajectory window

$$w_{t:H} \triangleq (s_{t:t+H},\, a_{t:t+H-1},\, \hat{R}_{t:t+H-1}) \tag{5}$$

into a latent embedding vector $z(w)$ that captures high-level behavioral intent:

$$z(w) = f_\theta(w) \in \mathbb{R}^d \tag{6}$$

This approach directly overcomes the deficiency of "spurious diversity" since $z(w)$ filters out irrelevant physical details while preserving core behavioral patterns, behaviorally similar trajectories naturally cluster in the latent representation space. Based on this, we compute the similarity $S_{ij}$ between any two windows $i$ and $j$ in the latent space using an RBF (Gaussian) kernel:

$$S_{ij} = \exp\left(-\frac{\|z_i - z_j\|_2^2}{\sigma^2}\right) \tag{7}$$

where the bandwidth $\sigma$ can be set via the median-of-pairwise-distances heuristic. The resulting similarity matrix $S$ provides a reliable basis for the subsequent sampling algorithm to assess the true behavioral diversity within the candidate pool.

**Quality Evaluation.** To focus sampling on experiences that most improve the model, we must define what constitutes "high quality." A simple criterion is high return, but this is limited because it ignores samples that, while not achieving high returns, are crucial for exploration and for remedying

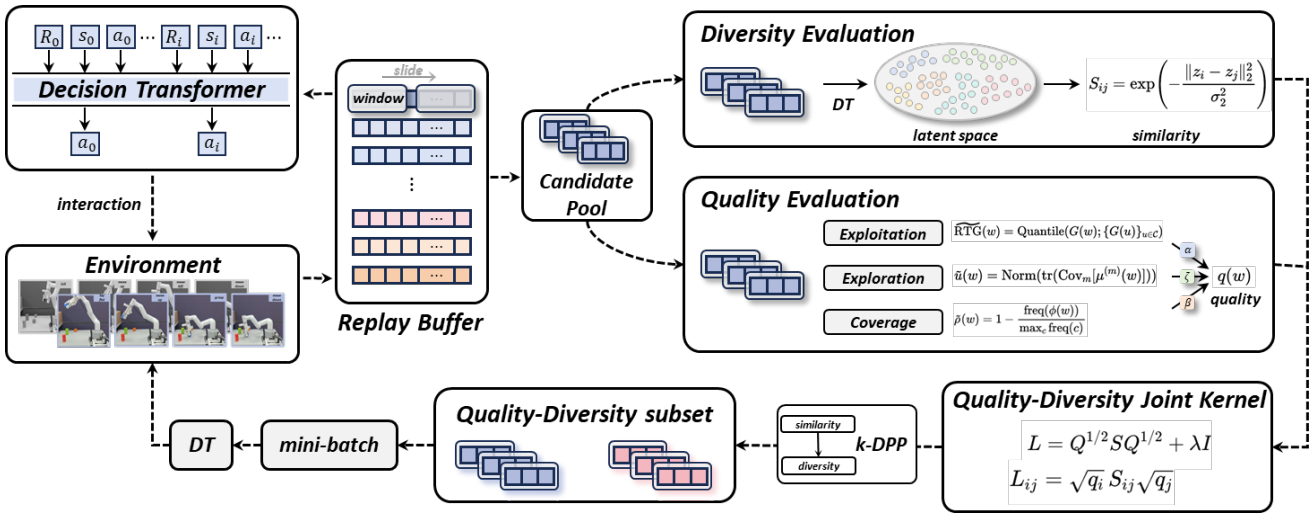


Fig. 3: The E²DT framework diagram illustrates a closed-loop system that deeply integrates the decision model with active data selection. The DT model autonomously extracts internal signals from buffer samples, which are used to evaluate experiences based on two dimensions: quality (defined by return, uncertainty, and coverage) and diversity (assessed by quantifying similarity in the latent representation space between samples). These dimensions are unified into a joint kernel, which guides the k-DPP sampler to select a high-quality and diverse training subset. Finally, the Debiased Mixed Replay mechanism ensures training stability while enabling efficient learning.

the model's cognitive blind spots. *Our key viewpoint is: sample quality should not be defined solely by final return, but by its marginal contribution to the current stage of learning.* To this end, we design a multi-dimensional composite quality score $q(w)$ that balances three core pillars of learning: exploitation, exploration, and coverage:

$$q(w) = \alpha\,\widetilde{\mathrm{RTG}}(w) + \beta\,\widetilde{u}(w) + \zeta\,\widetilde{\rho}(w),\ \alpha+\beta+\zeta = 1 \quad (8)$$

**(1) Exploitation: Identifying High-Value Samples via Return Quantiles.** This component aims to help the model quickly learn successful behavioral patterns. We first compute the discounted return of a window

$$G(w) = \sum_{k=0}^{H-1} \gamma^k r_{t+k} \quad (9)$$

However, using raw returns is sensitive to outliers and scale. For more robust identification of high-value experiences, we take the **quantile** of $G(w)$ within the candidate pool $\mathcal{C}$ as a normalized score:

$$\widetilde{\mathrm{RTG}}(w) = \mathrm{Quantile}\Big(G(w); \{G(u)\}_{u\in\mathcal{C}}\Big) \quad (10)$$

The advantage is that quantiles are insensitive to extreme outliers and can stably highlight samples that are "relatively strong" within the current candidate pool, thereby guiding the policy to converge toward validated successful directions.

**(2) Exploration: Locating Cognitive Boundaries via Uncertainty.** This component aims to guide the model to explore its "blind spots." We estimate predictive uncertainty for a window by performing $M$ stochastic forward passes (e.g., MC-Dropout). Specifically, we compute the trace of the covariance of the predictive means to quantify uncertainty:

$$u(w) = \mathrm{tr}\Big(\mathrm{Cov}_m[\mu^{(m)}(w)]\Big),\ \widetilde{u}(w) = \mathrm{Norm}\left(u(w)\right) \quad (11)$$

The advantage is that large $u(w)$ precisely points to regions where the model is most uncertain and needs learning most. Prioritizing such regions is an efficient way to fill cognitive gaps and improve generalization.

**(3) Coverage: Balancing Rare Stages via Inverse Frequency.** In long-horizon tasks, data distributions are often imbalanced: common and simple behaviors (e.g., moving in free space) may overshadow rare but critical sub-tasks (e.g., final precise alignment). To address this coverage-skew problem, we introduce stage coverage. Each window is assigned a stage label $\phi(w)$, and an inverse-frequency score is computed based on its frequency in the candidate pool:

$$\widetilde{\rho}(w) \;=\; 1 - \frac{\mathrm{freq}(\phi(w))}{\max_c \mathrm{freq}(c)} \quad (12)$$

Intuitively, $\widetilde{\rho}(w)$ converts stage frequency into a *rarity weight*: the fewer windows that share the label $\phi(w)$ in the current pool, the larger the score. The mapping is bounded in $[0,1]$, monotonically decreasing in frequency, and normalized by the maximal count, so it compares stages on a common scale and is insensitive to the absolute pool size. When incorporated into the composite quality score $q(w)$, larger $\widetilde{\rho}(w)$ increases the chance that underrepresented yet pivotal sub-tasks are sampled, broadening coverage across the full task pipeline and reducing the tendency to overfit frequent trivial behaviors. In practice, stage labels can come from environment annotations or from unsupervised clustering (e.g., $K$-means) of latent embeddings $z(w)$; light count smoothing (e.g., add–$\alpha$) may be applied to avoid

overemphasizing extremely rare outliers.

By fusing these three components with weights, our quality evaluation surpasses a single return-based metric and achieves a more comprehensive assessment of learning contribution that dynamically balances exploitation, exploration, and coverage.

### B. Constructing the Quality–Diversity Joint Kernel

After separately quantifying the *quality* of each sample ($q_i$) and the *diversity* between each pair of samples (via the similarity matrix $S$), we face a core challenge: how to fuse these two distinct sources of information, one acting on individual samples (scalar scores), the other on sample pairs (similarities), into a unified mathematical structure that can be utilized by the subsequent sampling algorithm.

To this end, we construct an L-ensemble joint kernel $L$ with the following specific form:

$$L \;=\; Q^{1/2} S Q^{1/2} + \lambda I, \text{ with } Q = \mathrm{diag}(q_1, \ldots, q_{|\mathcal{C}|}) \quad (13)$$

This formula is not a mere sum or product; it has deep geometric and probabilistic meaning. $Q$ is a diagonal matrix whose diagonal entries are the sample-wise quality scores $q_i$. Hence $Q^{1/2}$ is diagonal with entries $\sqrt{q_i}$. $Q^{1/2} S Q^{1/2}$: when this multiplication is performed, the $(i, j)$-th entry of the resulting L-ensemble kernel $L$ becomes

$$L_{ij} = \sqrt{q_i} \cdot S_{ij} \cdot \sqrt{q_j} \quad (14)$$

**Connecting Diversity and Quality: Why and How.** This result clearly reveals the core design. The "association strength" $L_{ij}$ between any two samples $i$ and $j$ in the new kernel is no longer their original latent-space similarity $S_{ij}$ alone, but is modulated by the geometric mean of their qualities ($\sqrt{q_i q_j}$). If the quality of any sample is low (e.g., $q_i \to 0$), then regardless of its relation $S_{ij}$ with any other sample $j$, the association strength $L_{ij}$ will approach zero. This means low-quality samples have systematically diminished influence on the global diversity structure. If both samples have high quality, then their association strength $L_{ij}$ will be amplified.

Ultimately, this construction is crucial for the subsequent DPP (Determinantal Point Process) sampling. DPP aims to select a subset that maximizes the determinant of its kernel submatrix. Geometrically, the determinant represents the "volume" spanned by the vectors in the subset. In our L-ensemble kernel, this "volume" depends on both quality and diversity: *Effect of quality*: the sample quality $q_i$ determines the "length" (norm) of its corresponding vector. High-quality samples correspond to longer vectors and contribute more to expanding the volume. *Effect of diversity*: the pairwise similarity $S_{ij}$ determines the "angle" between vectors. The more similar two vectors are ($S_{ij} \to 1$), the smaller the angle and the smaller their contribution to the volume (as they are nearly collinear).

Therefore, when DPP maximizes the determinant on $L$, it must choose a set of samples whose vectors are long (high quality) and mutually wide-angled (high diversity). This perfectly realizes our goal of selecting a "high-quality and complementary" subset and provides a principled (non-heuristic) solution with solid theoretical underpinnings.

**Algorithm 1** E$^2$DT

**Initialize:** DT policy $\pi_\theta$; replay buffer $\mathcal{D}$; window length $H$; pool size $N$; subset size $k$; refresh period $K$; mix ratio $\eta$; weights $(\alpha, \beta, \zeta)$; RBF bandwidth $\sigma$; kernel regularizer $\lambda$; set $\mathcal{Y} \leftarrow \emptyset$.
**while** not converged **do**
  **Collect** transitions with $\pi_\theta$ and append to $\mathcal{D}$.
  **if** $\mathcal{Y} = \emptyset$ **or** $(\text{step} \bmod K) = 0$ **then**
    **Candidate pool:** $\mathcal{C} \leftarrow$ sample $N$ windows $\{w_i\}_{i=1}^{N}$ of length $H$ from $\mathcal{D}$.
    **Latent representation diversity:** encode $z_i \leftarrow f_\theta(w_i)$; set $S_{ij} \leftarrow \exp\big(-\|z_i - z_j\|_2^2/\sigma^2\big)$.
    **Per-window quality:** for each $w_i$:
      $\widetilde{\mathrm{RTG}}_i \leftarrow$ RTG *quantile* of $w_i$ within $\mathcal{C}$;
      $\tilde{u}_i \leftarrow$ MC-dropout uncertainty
      $\tilde{\rho}_i \leftarrow$ inverse stage frequency;
      $q_i \leftarrow \alpha\, \widetilde{\mathrm{RTG}}_i + \beta\, \tilde{u}_i + \zeta\, \tilde{\rho}_i$.
    **Joint kernel:** $Q \leftarrow \mathrm{diag}(q_1, \ldots, q_N)$; $L \leftarrow Q^{1/2} S Q^{1/2} + \lambda I$.
    **k-DPP selection:** select $\mathcal{Y}$ via MAP inference from $L$ of size $k$.
  **end if**
  **Debiased mixed replay:** sample $\mathcal{B}_Y \sim \mathrm{Unif}(\mathcal{Y})$ of size $\lfloor \eta B \rfloor$ and $\mathcal{B}_D \sim \mathrm{Unif}(\mathcal{D})$ of size $B - |\mathcal{B}_Y|$; set $\mathcal{B} \leftarrow \mathcal{B}_Y \cup \mathcal{B}_D$.
  **Importance weights:** for each $i \in \mathcal{B}$, set $p_i \leftarrow \eta \frac{\mathbb{1}\{i \in \mathcal{Y}\}}{|\mathcal{Y}|} + (1-\eta)\frac{1}{|\mathcal{D}|}$, $\omega_i \propto 1/p_i$.
  **DT update:** minimize the weighted DT loss on $\mathcal{B}$ and update $\theta$.
**end while**

### C. k-DPP Based Sample Selection

The goal of this stage is to select a subset from the replay buffer that is simultaneously high in quality and non-redundant in the latent representation space.

**Subset Selection.** Given the joint kernel $L = Q^{1/2} S Q^{1/2} + \lambda I$, k-DPP [17] assigns to each subset $Y \subseteq \mathcal{C}$ a probability proportional to $\det(L_Y)$. The determinant equals the squared volume spanned by the vectors associated with $Y$, so larger values prefer samples that are individually strong (large norms from high $q_i$) and mutually different (large angles, low redundancy from $S$). We use maximum a posteriori selection with a fixed size $k$:

$$\mathcal{Y}^\star \;=\; \arg \max_{\mathcal{Y} \subseteq \mathcal{C},\, |\mathcal{Y}| = k} \log\det\big(L_{\mathcal{Y}}\big). \quad (15)$$

*How it works in practice:* A simple greedy MAP procedure builds $\mathcal{Y}$ one item at a time by adding the candidate with the largest marginal gain. If $Y$ is the current set, the gain of adding $i \notin Y$ is $\Delta(i \mid Y) = \log(L_{ii} - L_{iY}\, L_{YY}^{-1}\, L_{Yi})$ which is the conditional variance of $i$ after accounting for $Y$ (the Schur complement). This value is large when $i$ is high quality

and contributes information not already explained by $Y$, and it is small for near-duplicates. With incremental Cholesky or Sherman–Morrison updates, this runs in $O(kN^2)$ time and works well because the objective is submodular.

**Debiasing via Importance Sampling.** Training only on the selected subset $\mathcal{Y}$ introduces sampling bias. We address this with Debiased Mixed Replay: draw a fraction $\eta$ from $\mathcal{Y}$ and the remaining $1-\eta$ from the global buffer $\mathcal{D}$, then apply importance weights so that the gradient matches the true data distribution. For each sampled index $i$,

$$p_i \;=\; \eta\,\frac{\mathbb{1}\{i\in\mathcal{Y}\}}{|\mathcal{Y}|} \;+\; (1-\eta)\,\frac{1}{|\mathcal{D}|}, \omega_i \;=\; \frac{\frac{1}{|\mathcal{D}|}}{p_i} \quad (\;\omega_i \propto p_i^{-1}\;) \tag{16}$$

Finally, we update the DT by minimizing the importance-weighted loss on the mini-batch $\mathcal{B}$:

$$\mathcal{L}_{\mathrm{DT}}(\theta) \;=\; \sum_{i\in\mathcal{B}} \omega_i \sum_{j=0}^{H-1} \Big[ -\log \pi_\theta\big(a_{t+j} \mid s_{t:t+j}, \hat{R}_{t:t+j}\big)\Big] \tag{17}$$

These weights make the gradient an unbiased estimator under the full data distribution, while the mix with $\mathcal{D}$ preserves a global view and stabilizes training.

## V. Experiments

We evaluate the proposed E$^2$DT framework across RoboSuite [20] and ManiSkill2 [32] simulation suites and a real-world 6-DoF Elephant Robotics myCobot arm, focusing on whether DT-guided k-DPP sampling improves *learning efficiency*, *coverage/diversity*, and *final task success*. All methods share the same offline logs for pretraining and the same online finetuning budget and reset protocol. Results are averaged over 5 random seeds with confidence intervals.

**Baselines.** We compare against the following representative methods, each using identical observation/action spaces. **DT** [16]: A sequence model conditioned on return-to-go (RTG) for long-horizon credit assignment; trained with uniform replay. **HER** [33]: Hindsight experience relabeling method with off-policy training. **SynthER** [34]: Synthetic Experience Replay to generate synthetic experiences for data augmentation. **MAPLE** [35]: Augments RL with a library of predefined manipulation primitives for long-horizon tasks. **Relo** [36]: Reducible Loss (ReLo) is a sample prioritization method that ranks samples by their learnability, measured by the consistent reduction in loss over time. **SkillTree** [37]: A hierarchical framework that distills actions into a discrete skill space for long-horizon control tasks.

### A. RoboSuite

**Environment.** RoboSuite [20] is a physics-based benchmark for robotic manipulation tasks, using the MuJoCo simulator. We evaluate four long-horizon tasks: *Block Stacking*, *Nut Assembly*, *Door Opening*, and *Pick-and-Place*.Observations include robot proprioception (joint angles, velocities, gripper state) and task-specific object states. Actions are low-level joint commands or deltas of the end effector in the operational space, using the default RoboSuite controllers. Episodes terminate on success or a fixed horizon, with Success as the primary metric.

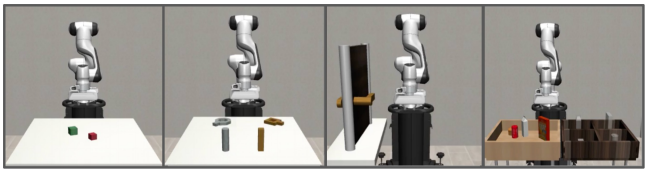

Fig. 4: Manipulation Tasks:[Block Stacking, Nut Assembly, Door Opening, Pick-and-Place]

We test on *Block Stacking*, *Nut Assembly*, *Door Opening*, and *Pick-and-Place*. E$^2$DT achieves higher final success and faster early improvement across all tasks. Gains are most pronounced in tasks with *rare but critical phases* (e.g., precise alignment in *Nut Assembly*, latch release in *Door Opening*). Diversity in the DT latent space prevents "spurious diversity" in raw state space, while the quality branch (RTG quantiles + uncertainty + stage coverage) focuses training on *high-value, complementary* windows, improving long-horizon credit assignment and reducing wasted exploration.

| Method | BlockStack | NutAsm | DoorOpen | PickPlace |
|---|---|---|---|---|
| DT | 60.1 ± 3.3 | 31.7 ± 3.2 | 46.8 ± 3.0 | 58.0 ± 3.4 |
| HER | 57.0 ± 3.5 | 26.4 ± 2.9 | 41.2 ± 2.8 | 51.1 ± 3.1 |
| SynthER | 67.2 ± 3.2 | 35.0 ± 3.0 | 49.4 ± 3.3 | 60.5 ± 2.0 |
| MAPLE | 72.3 ± 3.0 | 42.1 ± 2.8 | 55.5 ± 3.1 | 65.6 ± 3.2 |
| Relo | 70.5 ± 3.1 | 38.7 ± 2.9 | 53.1 ± 3.2 | 62.4 ± 2.8 |
| SkillTree | 71.8 ± 2.9 | 40.3 ± 2.8 | 54.0 ± 2.6 | 63.5 ± 3.1 |
| **E$^2$DT** | **79.8 ± 2.4** | **55.6 ± 2.3** | **65.2 ± 2.4** | **73.7 ± 2.6** |

TABLE I: Mean success rates (%) comparison in RoboSuite.

### B. ManiSkill2

**Environment.** ManiSkill2 [32] is a high-fidelity benchmark for robotic manipulation with photorealistic rendering and physics, domain randomization for generalization.

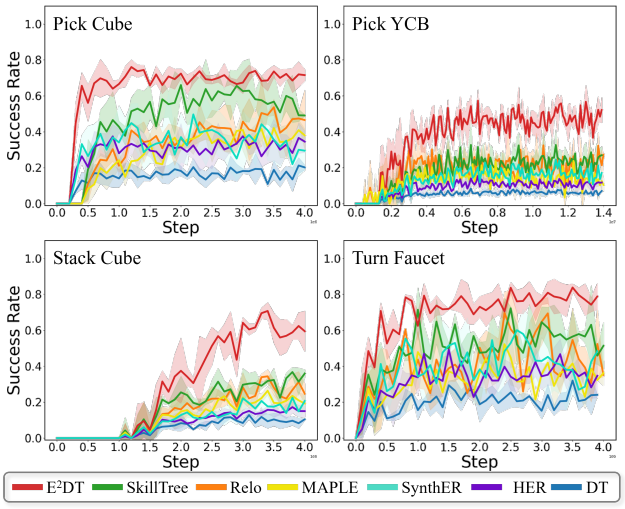


Fig. 5: Mean success rates (%) comparison in ManiSkill2.

We evaluate four vision-based, long-horizon tasks: *Pick Cube)*, *Pick YCB*, *Stack Cube*, and *Turn Faucet*. Observations

consist of $128 \times 128$ RGB-D images, and actions follow the benchmark's default low-level controller for the end-effector. Episodes terminate on success or a fixed horizon, with Success as the primary metric.

E$^2$DT achieves higher final success and faster early improvement across all four tasks (Fig. 5). Measuring diversity in the DT *latent* embedding space avoids spurious diversity from raw pixels, while the quality composite (RTG *quantiles* + predictive uncertainty + stage coverage) steers selection toward *high-value, complementary* windows. Gains are most pronounced on *Stack* and *Turn/Open*, where rare but critical sub-stages (e.g., precise alignment, initiating rotation) dominate success; k-DPP with the joint kernel increases coverage of these sub-stages without sacrificing sample efficiency.

### C. Real-World Robotic Manipulation (Elephant Robotics)

**Platform.** We deploy E$^2$DT on an Elephant Robotics 280 desktop manipulator (6-DoF) with a two-finger parallel gripper. The arm is base-mounted over a flat tabletop workspace. Perception uses an *eye-in-hand* RGB camera on the wrist; proprioception (joint positions/velocities, gripper state) is always available. Actions are operational–space end-effector deltas (translation/rotation + gripper open/close) tracked by the vendor controller. Episodes terminate on success or a fixed horizon; standard safety interlocks (velocity limits, E-stop) are enabled.

(a) High-Shelf (b) Stacking (c) Target Grasping

Fig. 6: Real-world tasks on Elephant Robotics 280: (a) High-Shelf Placement, (b) Stacking, (c) Target Grasping.

**Tasks.** We evaluate three long-horizon tasks: (i) *High-Shelf Placement*: grasp an object and place it onto an elevated shelf with tight final pose tolerance; (ii) *Stacking*: sequentially grasp and stack blocks to a target configuration; (iii) *Target Grasping*: selectively grasp a specified item among distractors and hold it stably for a dwell time.

| Method | High-Shelf | Stacking | Target Grasp |
|---|---|---|---|
| DT | 52.4 ± 3.8 | 56.1 ± 3.2 | 54.7 ± 3.6 |
| HER | 54.9 ± 3.6 | 53.3 ± 3.4 | 50.1 ± 3.1 |
| SynthER | 63.2 ± 3.1 | 64.8 ± 2.9 | 61.7 ± 3.0 |
| MAPLE | 66.5 ± 2.8 | 68.7 ± 2.7 | 66.0 ± 2.9 |
| Relo | 61.5 ± 3.1 | 64.7 ± 3.3 | 63.0 ± 2.8 |
| SkillTree | 69.8 ± 3.3 | 71.5 ± 2.5 | 68.9 ± 2.6 |
| **E$^2$DT** | **82.1 ± 2.7** | **85.6 ± 2.4** | **83.4 ± 2.5** |

TABLE II: Mean success rate comparison (%, mean ± CI over 5 seeds) in real-world manipulation tasks.

**Results.** E$^2$DT achieves target success with *fewer episodes* and demonstrates *faster early learning* compared to baselines across all tasks. The quality branch (RTG *quantiles*, predictive uncertainty, and stage coverage) focuses updates on *key phases* (e.g., grasp onset, lift-to-transport transition, final alignment for shelf placement), while DT's latent space diversity prevents "spurious diversity" from raw images. The k-DPP applied to the joint kernel $L = Q^{1/2} S Q^{1/2} + \lambda I$ enhances coverage of *rare but critical* sub-stages without increasing redundancy, resulting in quicker convergence and higher final success (Tab. II).

### D. Ablation Study and Sensitivity

We conduct ablations on *Nut Assembly*, a long-horizon task with a rare but critical alignment phase, where data selection strongly affects learning. We evaluate four variants: **(1) E$^2$DT (Full)**, k-DPP with the quality-diversity joint kernel and debiased mixed replay; **(2) Quality-Only**, remove diversity (no k-DPP), greedily select top-$q(w)$ windows; **(3) Diversity-Only**, remove quality (set $q_i$ constant), select purely by DT-latent space diversity via k-DPP; **(4) Uniform Replay**, no active selection. Removing k-DPP increases redundancy and slows learning; quality-only collapses coverage and over-focuses on local high-return regions; diversity-only preserves breadth but under-utilizes high-value samples, hurting efficiency and convergence. Only when *quality and diversity are jointly enforced* via k-DPP on $L{=}Q^{1/2} S Q^{1/2}{+}\lambda I$ do we obtain the best efficiency and robustness. Performance remains stable for $k/N \in [0.08, 0.20]$ and $\eta \in [0.6, 0.8]$. Increasing $H$ improves temporal consistency but may reduce diversity in the candidate pool. The pairwise-median heuristic for $\sigma$ is robust. The choice of $K \in [5\text{k}, 15\text{k}]$ strikes a balance between computation and buffer-drift tracking. Similar trends are observed for the real-robot *High-Shelf Placement* task.

| Variant | Success % ↑ | Redund.% ↓ | Diversity ↑ |
|---|---|---|---|
| E$^2$DT (Full) | **55.6** ± 2.3 | **10.1** ± 1.2 | 0.87 ± 0.03 |
| Quality-Only | 49.6 ± 2.8 | 32.9 ± 2.1 | 0.46 ± 0.05 |
| Diversity-Only | 43.8 ± 3.0 | 13.4 ± 1.5 | **0.89** ± 0.04 |
| Uniform Replay | 31.2 ± 3.4 | 42.3 ± 2.7 | 0.40 ± 0.06 |

TABLE III: Ablation on *Nut Assembly* (mean ± CI over 5 seeds). *Diversity* is normalized $\log\det(S_{\mathcal{Y}})$; *Redundancy* is the near-neighbor rate in the DT embedding space.

## VI. Conclusion

We presented E$^2$DT, an experience-aware approach that pairs a Decision Transformer with active data selection to make training both efficient and effective. By scoring each training window for quality (RTG quantiles, uncertainty, stage coverage) and diversity (DT latent embeddings), then selecting with k-DPP and training with debiased mixed replay, E$^2$DT focuses on the most informative experiences. Experiments in RoboSuite, ManiSkill2, and on a real arm confirm consistent gains over strong baselines. Future work aims to automate quality–diversity weighting.

## ACKNOWLEDGMENTS

This work was partially supported by the Key Research and Development Project of Hubei Province (2025BAB023). The calculations were performed on the supercomputing system at Wuhan University's Center for Supercomputing.

## REFERENCES

[1] H. Nguyen and H. La, "Review of deep reinforcement learning for robot manipulation," in *2019 Third IEEE international conference on robotic computing (IRC)*. IEEE, 2019, pp. 590–595.

[2] A. Billard and D. Kragic, "Trends and challenges in robot manipulation," *Science*, vol. 364, no. 6446, p. eaat8414, 2019.

[3] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, 2017.

[4] C. Tang, B. Abbatematteo, J. Hu, R. Chandra, R. Martín-Martín, and P. Stone, "Deep reinforcement learning for robotics: A survey of real-world successes," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 27, 2025, pp. 28 694–28 698.

[5] N. R. Ke, A. Singh, A. Touati, A. Goyal, Y. Bengio, D. Parikh, and D. Batra, "Modeling the long term future in model-based reinforcement learning," in *International Conference on Learning Representations*, 2019.

[6] Y. Cao, R. Zhao, Y. Wang, B. Xiang, and G. Sartoretti, "Deep reinforcement learning-based large-scale robot exploration," *IEEE Robotics and Automation Letters*, vol. 9, no. 5, pp. 4631–4638, 2024.

[7] P. Ladosz, L. Weng, M. Kim, and H. Oh, "Exploration in deep reinforcement learning: A survey," *Information Fusion*, vol. 85, pp. 1–22, 2022.

[8] Y. Wang, M. Yang, R. Dong, B. Sun, F. Liu, *et al.*, "Efficient potential-based exploration in reinforcement learning using inverse dynamic bisimulation metric," *Advances in Neural Information Processing Systems*, vol. 36, pp. 38 786–38 797, 2023.

[9] Y. Wang, K. Zhao, F. Liu, *et al.*, "Rethinking exploration in reinforcement learning with effective metric-based exploration bonus," *Advances in Neural Information Processing Systems*, vol. 37, pp. 57 765–57 792, 2024.

[10] T. Huang, K. Chen, B. Li, Y. Liu, and Q. Dou, "Demonstration-guided reinforcement learning with efficient exploration for task automation of surgical robot," in *IEEE International Conference on Robotics and Automation, ICRA 2023, London, UK, May 29 - June 2, 2023*. IEEE, 2023, pp. 4640–4647. [Online]. Available: https://doi.org/10.1109/ICRA48891.2023.10160327

[11] J. Zhang, J. Kim, B. O'Donoghue, and S. Boyd, "Sample efficient reinforcement learning with reinforce," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 12, 2021, pp. 10 887–10 895.

[12] H. Yuan, Z. Mu, F. Xie, and Z. Lu, "Pre-training goal-based models for sample-efficient reinforcement learning," in *The Twelfth International Conference on Learning Representations*, 2024.

[13] T. Bi and R. D'Andrea, "Sample-efficient learning to solve a real-world labyrinth game using data-augmented model-based reinforcement learning," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 7455–7460.

[14] S. Lee, J. Kim, I. Jang, and H. J. Kim, "Dhrl: A graph-based approach for long-horizon and sparse hierarchical reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 13 668–13 678, 2022.

[15] M. Sivertsvik, K. Sumskiy, and E. Misimi, "Learning active manipulation to target shapes with model-free, long-horizon deep reinforcement learning," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 5411–5418.

[16] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, "Decision transformer: Reinforcement learning via sequence modeling," *Advances in neural information processing systems*, vol. 34, pp. 15 084–15 097, 2021.

[17] A. Kulesza and B. Taskar, "k-dpps: Fixed-size determinantal point processes," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 1193–1200.

[18] A. Kulesza, B. Taskar, *et al.*, "Determinantal point processes for machine learning," *Foundations and Trends® in Machine Learning*, 2012.

[19] C. Li, S. Jegelka, and S. Sra, "Efficient sampling for k-determinantal point processes," 2016. [Online]. Available: https://arxiv.org/abs/1509.01618

[20] Y. Zhu, J. Wong, A. Mandlekar, R. Martín-Martín, A. Joshi, S. Nasiriany, Y. Zhu, and K. Lin, "robosuite: A modular simulation framework and benchmark for robot learning," in *arXiv preprint arXiv:2009.12293*, 2020.

[21] D. Han, B. Mulyana, V. Stankovic, and S. Cheng, "A survey on deep reinforcement learning algorithms for robotic manipulation," *Sensors*, vol. 23, no. 7, 2023.

[22] Y. Wang, K. Zhao, Y. Li, and L. H. U, "Bile: an effective behavior-based latent exploration scheme for deep reinforcement learning," in *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, 2025, pp. 6497–6505.

[23] K. Zhao, Y. Wang, Y. Chen, Y. Li, X. Niu, *et al.*, "Efficient diversity-based experience replay for deep reinforcement learning," *arXiv preprint arXiv:2410.20487*, 2024.

[24] A. Ororbia and A. A. Mali, "Active predictive coding: Brain-inspired reinforcement learning for sparse reward robotic control problems," in *IEEE International Conference on Robotics and Automation, ICRA 2023, London, UK, May 29 - June 2, 2023*. IEEE, 2023, pp. 3015–3021. [Online]. Available: https://doi.org/10.1109/ICRA48891.2023.10160530

[25] J. Chen, T. Lan, and V. Aggarwal, "Option-aware adversarial inverse reinforcement learning for robotic control," in *IEEE International Conference on Robotics and Automation, ICRA 2023, London, UK, May 29 - June 2, 2023*. IEEE, 2023, pp. 5902–5908. [Online]. Available: https://doi.org/10.1109/ICRA48891.2023.10160374

[26] S. Hegde, Z. Huang, and G. S. Sukhatme, "Hyperppo: A scalable method for finding small policies for robotic control," in *IEEE International Conference on Robotics and Automation, ICRA 2024, Yokohama, Japan, May 13-17, 2024*. IEEE, 2024, pp. 10 821–10 828. [Online]. Available: https://doi.org/10.1109/ICRA57147.2024.10610861

[27] S. Gu, E. Holly, T. P. Lillicrap, and S. Levine, "Deep reinforcement learning for robotic manipulation," *arXiv preprint arXiv:1610.00633*, vol. 1, no. 1, 2016.

[28] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," 2013. [Online]. Available: https://arxiv.org/abs/1312.5602

[29] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2019. [Online]. Available: https://arxiv.org/abs/1509.02971

[30] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*. PmLR, 2016, pp. 1928–1937.

[31] W. Yuan, J. Chen, S. Chen, D. Feng, Z. Hu, P. Li, and W. Zhao, "Transformer in reinforcement learning for decision-making: a survey," *Frontiers of Information Technology & Electronic Engineering*, vol. 25, no. 6, pp. 763–790, 2024.

[32] J. Gu, F. Xiang, X. Li, Z. Ling, X. Liu, T. Mu, Y. Tang, S. Tao, X. Wei, Y. Yao, X. Yuan, P. Xie, Z. Huang, R. Chen, and H. Su, "Maniskill2: A unified benchmark for generalizable manipulation skills," 2023. [Online]. Available: https://arxiv.org/abs/2302.04659

[33] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, P. Abbeel, and W. Zaremba, "Hindsight experience replay," in *Neural Information Processing Systems*, 2017.

[34] C. Lu, P. Ball, Y. W. Teh, and J. Parker-Holder, "Synthetic experience replay," *Advances in Neural Information Processing Systems*, vol. 36, pp. 46 323–46 344, 2023.

[35] S. Nasiriany, H. Liu, and Y. Zhu, "Augmenting reinforcement learning with behavior primitives for diverse manipulation tasks," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 7477–7484.

[36] S. Sujit, S. Nath, P. Braga, and S. Ebrahimi Kahou, "Prioritizing samples in reinforcement learning with reducible loss," *Advances in Neural Information Processing Systems*, vol. 36, pp. 23 237–23 258, 2023.

[37] Y. Wen, S. Li, R. Zuo, L. Yuan, H. Mao, and P. Liu, "Skilltree: Explainable skill-based deep reinforcement learning for long-horizon control tasks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 20, 2025, pp. 21 491–21 500.