applied sciences

MDPI

Article

Pairwise Classification as a Unified Framework for Offline Reinforcement Learning and Large-Language-Model Alignment

Chayoung Kim

Bright College (College of Liberal Arts and Interdisciplinary Studies), Hankyong National University, 327, Jungang-ro, Anseong-si 17579, Gyeonggi-do, Republic of Korea; kimcha0@hknu.ac.kr; Tel.: +82-31-670-5670

Featured Application

The proposed reference-free pairwise classification framework can reduce training cost and implementation complexity in pairwise preference-scoring settings, as demonstrated for offline RL and LLM reward modeling in this study; extension to other preference-scoring settings such as recommendation systems or image-generation preference learning remains to be validated.

Abstract

Offline reinforcement learning (offline RL) and large-language-model (LLM) alignment are typically studied as independent domains and each has developed its own pairwise comparison technique. Prior studies have validated pairwise classification exclusively within offline RL, leaving open the question of whether it generalizes beyond a single domain. This study reconceptualizes pairwise classification not as a technique confined to offline RL, but as a domain-agnostic structural principle for determining which of the two candidates scores higher when a context is shared. We instantiate this principle as a pairwise-Decision Transformer (pairwise-DT), which extends the Decision Transformer (DT) by comparing the Q-values of two transitions in offline RL, and pairwise fine-tuning (pairwise-FT), which compares the preference scores of two responses in the LLM alignment by applying the same pairwise loss across both domains. Through experiments on three MuJoCo environments and the anthropic Helpful and Harmless Reinforcement Learning from Human Feedback (HH-RLHF) dataset, pairwise-FT achieved higher reward accuracy than direct-preference optimization (DPO) across five seeds in approximately half the training time, without a reference model. The pairwise DT outperformed the standard DT in all three MuJoCo environments. Sensitivity analysis over a range of temperature values confirms that this advantage is not an artifact of a favorably chosen hyperparameter; this robustness holds strongly on the LLM side and, to a lesser extent, on the MuJoCo side. On the LLM side, this advantage reflects preference-scoring/reward-modeling performance specifically, rather than full generation-quality alignment.

Keywords: offline reinforcement learning; Decision Transformer; RLHF; Direct-Preference Optimization; pairwise classification; reward modeling; cross-domain generalization

Check for updates

Academic Editor: Jun Chen

Received: 3 August 2026
Revised: 6 September 2026
Accepted: 10 September 2026
Published: 14 September 2026

Copyright: © 2026 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Reinforcement learning (RL) from human feedback (RLHF) [1,2] is the dominant framework for aligning large language models (LLMs) with human preferences. In parallel, offline reinforcement learning (offline RL) [3] has developed into a core research area

that learns control policies solely from a fixed pre-collected dataset without additional environmental interactions. These two domains share a fundamental underlying operation: determining which of the two candidates is preferred within a shared context. This study formally verifies that the underlying operation reduces to the same abstract structure across domains (Section 3.3). In offline RL, this takes the form of comparing the values of two actions in the same state, whereas in the LLM alignment, it takes the form of comparing two responses to the same prompts. Despite this structural similarity, the two fields have largely developed independent methodologies. Whether a single pairwise formulation can be applied to both without a domain-specific redesign has not yet been systematically verified.

Offline RL learns a policy exclusively from a fixed pre-collected dataset. This eliminates the cost of environmental interaction and ensures safety, but it introduces the distinctive problem of instability caused by value overestimation on out-of-distribution actions. Decision Transformer (DT) [4] reframes this challenge as conditional sequence modeling, autoregressively predicting actions given a target return-to-go. Multi-Game DT (MGDT) [5] extends this by incorporating a binary expert/non-expert classifier as an auxiliary objective. However, this binary formulation only distinguishes the absolute expert status and does not capture the relative quality between the two actions; therefore, it fails to fully exploit the ordinal information that is already latent in the data.

In LLM alignment, the standard RLHF pipeline first learns a reward model from pairwise human preference data following the Bradley–Terry model [6] and then uses this reward model to optimize the policy via RL in a two-stage procedure. Direct-Preference Optimization (DPO) [7] compresses this two-stage procedure into a single optimization by reparameterizing the reward as the log-probability ratio of the policy, relative to the fixed reference policy. However, this reference policy must remain fixed throughout the training and requires an additional forward pass at every optimization step. We empirically measure and quantify this computational cost in Section 4.3.

The author's prior work has already demonstrated that reformulating value estimation in offline RL as a pairwise classification, rather than regression, improves training stability. Kim [8] was the first to replace regression-based Q-value estimation with a classification objective, and Kim [9] extended this idea by comparing the Q-values of two pairwise actions to replace the BEAR's distribution-based constraint, improving performance by 3–5×; and Kim [10] further showed, through hard-annealing experiments in multitask offline RL, that pairwise ranking is not merely a temporary warm-up technique but a structural constraint that must be maintained throughout training. However, all the three studies validated this pairwise structure exclusively within offline RL. Thus, the open question remains: is pairwise classification a general principle applicable across domains, or merely a technique tailored to the value-estimation challenges specific to offline RL?

This study advances a fundamentally different perspective: pairwise classification should be understood as a domain-agnostic structural principle for preference learning, rather than a technique confined to offline RL. Any situation in which two candidates are compared under a shared context—whether the underlying score is a Q-value or a language model's preference score—can be reduced to the same classification problem: "which of the two candidates scores higher". From this perspective, the distinction between offline RL and LLM alignment is not a matter of unrelated problems but a choice of which domain-specific scorer (a Q-network or a preference head) instantiates the same abstract objective.

The objective of the unified formulation proposed in this study was not simply to demonstrate the reusability of pairwise loss across distinct domains. Rather, by reducing comparison to direct pairwise judgment, the central aim was to evaluate whether reference-

based regularization—integral to methods such as DPO—can be safely eliminated. If reference-free pairwise classification was to succeed in offline RL merely by chance, its advantages should not be extrapolated to heterogeneous domains such as LLM alignment. Conversely, if pairwise classification constitutes a genuine structural principle, removing the reference model should not impair performance in either domain.

Building on this insight, this study proposes a unified pairwise classification framework instantiated through two domain-specific methods: pairwise-DT for offline RL, which compares the Q-values of two ranked transitions within the same minibatch, and pairwise-FT for LLM alignment, which compares the preference scores of two responses to a common prompt. Both methods are trained with the same pairwise loss, differing only in the domain-specific backbone (pairwise-DT utilizes a Decision Transformer backbone, whereas pairwise-FT utilizes a GPT-2 [11] backbone) that produces the scorer.

Through experiments conducted across both domains, this study empirically shows that this unified formulation is not confined to a single domain. In the LLM alignment, Pairwise-FT consistently achieved a higher reward accuracy than DPO across all five random seeds tested while requiring approximately half of the training time and dispensing with a reference model. In offline RL, pairwise-DT outperformed DT on an environment-mean basis across all three MuJoCo [12] environments tested, corroborating prior single-domain findings. Sensitivity analysis of the temperature parameter further indicates that this robustness holds strongly on the LLM side and, to a lesser extent, on the MuJoCo side, rather than being contingent on any single favorably chosen value. Collectively, these observations establish pairwise classification as a structural principle that generalizes to heterogeneous domains, including LLM alignment, rather than as a phenomenon that arises by chance in offline RL.

This study makes the following four contributions:

- Conceptual reframing: This study reconceptualizes pairwise classification not as a technique confined to offline RL, but as a domain-agnostic structural principle for preference learning, a perspective absent from the author's prior single-domain studies. To be clear, the novelty here is not the pairwise loss itself, which builds directly on the classical Bradley–Terry/RankNet formulation (Section 2.5) and on the authors' own prior single-domain work (Section 2.2); rather, it is the cross-domain empirical demonstration that a reference-free pairwise structure, validated only within offline RL until now, remains competitive with reference-dependent methods in a fully heterogeneous domain such as LLM alignment.
- Unified formulation: We demonstrate that the offline RL objective (comparing the Q-values of two transitions) and the LLM alignment objective (comparing the preference scores of two responses) are reducible to the same abstract pairwise classification problem under shared notation (Section 3.3).
- Cross-domain empirical evidence: Through experiments in both offline RL and LLM alignments, we show that reference-free pairwise classification achieves competitive or superior results compared with reference-dependent alternatives (DPO), backed by statistical evidence (sign test) and hyperparameter robustness (sensitivity analysis).
- Honest treatment of asymmetry: We explicitly acknowledge and analyze the structural asymmetry between domains—LLM alignment compares two responses under a similar prompt, whereas offline RL approximates this with within-batch ranking rather than exactly the same state, and discusses its implications without overstating the unification (Section 3 and part of Section 5).

All empirical claims in this paper are scoped to the specific settings evaluated: three MuJoCo continuous-control environments for the offline-RL side, and a GPT-2 (124 M) model trained on the HH-RLHF helpful-base subset for the LLM side; extrapolation be-

yond these settings is discussed as future work in Section 5.6. The remainder of this paper is organized as follows: Section 2 reviews related work, Section 3 details the proposed unified pairwise classification framework, Section 4 presents the experimental setup and results for both domains, Section 5 discusses the implications of the results, and Section 6 concludes the paper.

2. Related Work

This study focuses on the intersection of three areas: offline reinforcement learning (offline RL), classification-based value estimation, and LLM alignment/preference optimization. This section reviews the prior work in each area and clarifies ways in which the present study diverges from and extends existing approaches.

2.1. Decision-Transformer-Family Offline RL

Decision Transformer (DT) [4] reframes offline RL as a conditional sequence modeling problem over return-to-go (RTG), state, and action sequences. Unlike existing offline RL algorithms (CQL [13] and IQL [14]), DT is trained to autoregressively predict the next action given a target RTG, without explicitly learning a Q-value or the Bellman equation [15]. Multigame decision transformers (MGDTs) [5] extend this structure to learn a single policy across multiple environments/games, discretize the return distribution, and jointly use an auxiliary objective in the form of binary classification over expert actions. While these approaches exploit the representational capacity of sequence models, they lack a structural mechanism for directly comparing the relative merits of two actions.

CQL and IQL address distributional shift through value regularization within a value-based (non-sequence-modeling) offline RL paradigm; because they do not share the sequence-modeling backbone or the pairwise comparison mechanism central to this study's framework, we discuss them as related value-based approaches (Section 2.1) rather than as direct pairwise-structure comparison baselines.

2.2. Classification-Based Offline RL—The Author's Prior Research Lineage

Through three recent studies, the author's research group has conducted a line of work that transforms regression-based value estimation in offline RL into a classification-based formulation and further develops this into a pairwise structure. This lineage is summarized as follows:

- Regression Classification: Kim [8] first reformulated Q-value estimation in continuous actor-critic offline RL from mean-squared-error regression into a classification problem, replacing the regression loss with a cross-entropy/KL-divergence objective and thereby improving training stability over SAC/TD3-style baselines. This work established that a classification-based value target is viable, but did not yet compare two actions directly; it classifies a single action's value bucket rather than judging which of two actions is better.
- Classification → Pairwise Classification (within a single domain): Building on this, Kim [9] introduced a pairwise structure that directly compares the Q-values of two actions sampled from the same state, replacing BEAR's distributional constraint with a binary pairwise classifier and reporting an average 3× and up to 5× performance improvement. This is the first point in the lineage where the core pairwise comparison mechanism used throughout the present paper (Section 3.1) appears, but it was validated only in single-task offline RL.
- Establishing the structural status of Pairwise Classification: Kim [10] then established, through hard-annealing ablations in multitask offline RL, that this pairwise ranking is not a temporary warm-up mechanism but a structural constraint that must

be maintained throughout training—removing it after sufficient training caused an 85% performance loss on the Hopper task within 1K steps. This work confirmed the pairwise structure's necessity within offline RL but did not examine whether it generalizes beyond this single domain, which is the gap the present paper addresses.

Note that this necessity was established in a different setting—multitask offline RL with hard-annealing (removing an already-converged pairwise constraint mid-training)—whereas the present paper's pairing ablation (Section 4.2) tests single-task training from scratch without the pairwise term in a hybrid DT + critic + pairwise architecture; the two findings probe different questions and are not in direct tension, as discussed in Section 5.1.

2.3. RLHF and Preference Optimization

RL from human feedback (RLHF) proposes a two-stage pipeline that learns a reward model from data in which humans select their preferred response between two options, and then uses this reward model to optimize a policy via RL. InstructGPT [2] applied this pipeline to large language models, significantly improving the instruction-following performance. The standard reward model was trained with pairwise loss following the Bradley–Terry model, which maximizes the sigmoid of the score difference between the two responses. Direct-Preference Optimization (DPO) [7] proposed a method that optimizes the policy directly from preference data using the log-probability ratio relative to a reference policy without a separate reward-model training stage. However, the DPO must maintain a fixed reference policy throughout training, incurring an additional forward-pass cost at every training step.

2.4. Relationship to DPO

DPO is the method against which this study directly benchmarks in the LLM alignment domain, and it represents the most closely related prior work methodologically. This study differs from DPO in the following five aspects.

1. Domain scope: The DPO is defined exclusively within a single domain of the LLM alignment. This study applies the same pairwise objective to both offline RL (Pairwise-DT) and LLM alignment (Pairwise-FT) to evaluate the cross-domain generality.
2. Reference-model dependence: The DPO maintains a fixed reference policy throughout the training and requires a forward pass at every step. The pairwise objective in this study directly compares only the score difference between two candidates; therefore, a reference model is not required.
3. Derivation: The objective of the DPO is derived by reparameterizing the RLHF reward under a KL-divergence constraint. The pairwise objective of this study is defined directly as the classification loss, without relying on the KL-divergence-constrained policy optimization framework.
4. Empirical demonstration of efficiency: In Section 4.3, we empirically measure the effect of removing the forward pass of the reference model on the training time (an approximately 1.9× reduction). This computational cost was not addressed in the original DPO study.
5. Generality of domain instantiation: The pairwise objective in this study is instantiated in the same form as pairwise-DT (comparing Q-values) and pairwise-FT (comparing preference scores), whereas DPO's derivation of the DPO is specialized for language-model policies and does not naturally extend to a value-based RL setting.

In summary, both approaches share the common ground of being based on pairwise comparison, yet they diverge in purpose and scope: DPO targets reference-dependent optimization within a single domain, whereas this study targets reference-free classification across two domains.

2.5. Other Related Work

The Bradley–Terry model (1952) is a classical statistical model that estimates the relative preference from pairwise comparisons between two items and underlies the theoretical basis of most pairwise preference-learning approaches, including the present study. Similarly, the RankNet-family methods [16] within the learning-to-rank literature employ a pairwise loss to learn the relative ordering of two candidates. However, these methods have mainly been applied to information retrieval ranking problems, and their application to offline RL or LLM alignment has not been explicitly addressed prior to this study's lineage (Section 2.2).

This review focuses on literature methodologically adjacent to pairwise ranking, reward modeling, preference optimization, and offline reinforcement learning. Work in other domains that was suggested for consideration (e.g., few-shot visual recognition, wildfire risk prediction) is not discussed here, as it does not share a direct methodological connection with the pairwise classification framework proposed in this paper.

3. Method: Unified Pairwise Classification Framework

Section 3.1 adopts, without modification, the pairwise Q-value classifier structure already established in the author Kim's [9] prior work for offline RL. The novelty of this section does not lie in Section 3.1 itself, but rather in extending this structure to the LLM alignment domain in Section 3.2, and explicitly formalizing the correspondence between the two domains in Section 3.3. Figure 1 illustrates the proposed framework.

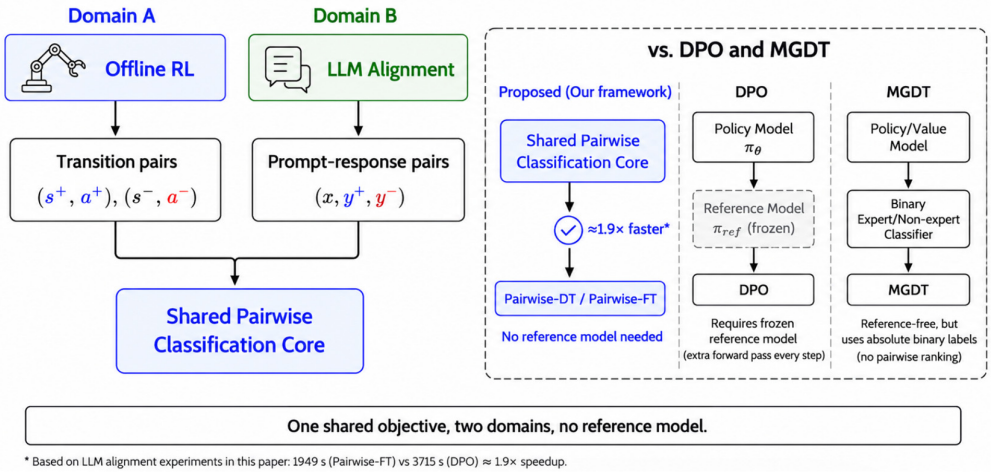


Figure 1. Unified Pairwise Classification Framework (Conceptual Overview). Domain A (Offline RL) and Domain B (LLM Alignment) each construct pairwise samples—transition pairs (s^+, a^+) and (s^-, a^-), and prompt-response pairs (x, y^+, y^-), respectively—which are passed to a Shared Pairwise Classification Core. The right panel contrasts the proposed framework with two comparison baselines (DPO, MGDT): the proposed core requires no reference model, whereas DPO requires a frozen reference policy π_{ref} and an extra forward pass at every training step, whereas MGDT is reference-free but uses binary expert/non-expert classification rather than pairwise ranking. Based on the LLM alignment experiments in this paper, this yields an approximately 1.9× speedup (1949 s for Pairwise-FT vs. 3715 s for DPO).

3.1. Offline RL-Side Formulation

The goal of pairwise Q-value classification in offline RL is as follows: When sampling two transitions, a^+, a^- within a minibatch, if $Q(s, a^+) > Q(s, a^-)$ then a^+ is superior.

However, in an offline dataset, it is difficult to directly obtain two different actions collected at the same state s. In practice, therefore, we rank multiple transitions within a minibatch by the critic's predicted Q-value, using the top half as (s^+, a^+) and the bottom

half as (s^-, a^-). Consequently, s^+ and s^- are generally different states, making this procedure an approximation of a direct comparison within the same state. The implications of this approximation for the correspondence in Section 3.3 are discussed again at the end of that section. To address the concern that critic-based pairing may be self-referential, we conducted a pairing ablation (Section 4.2) comparing this baseline against pairing by ground-truth return-to-go and against removing the pairwise term entirely; the ablation did not confirm that critic-based pairing is necessary, and in one environment (HalfCheetah) the no-pair condition performed better on both mean and variance (see Section 4.2 for full results and discussion). We emphasize that this approximation is not incidental: because offline datasets rarely contain multiple actions collected at the exact same state, s+ and s- are in general distinct states, and this asymmetry is revisited explicitly in Sections 3.3, 5.1, and 5.5 rather than glossed over.

Pairwise classifier training objective:

$$\mathcal{L}_{\mathrm{RL}} = -\mathbb{E}_{(s^+,a^+,s^-,a^-)\sim\mathcal{D}}\left[\log\sigma\left(\frac{Q_\theta(s^+,a^+) - Q_\theta(s^-,a^-)}{\tau}\right)\right] \tag{1}$$

where τ denotes the temperature, σ denotes the sigmoid, and $\mathcal{D}$ denotes the offline dataset. τ controls the sharpness of the sigmoid with respect to the difference between the two scores. A smaller τ yields sharper confidence even for a small score difference, which can destabilize optimization. Conversely, a larger τ flattens the judgment and the training signal weakens. Following the convention of the author Kim's [10] prior work, this study adopts $\tau = 1.0$ as a neutral value.

This formulation learns only the relative order of two actions rather than directly regressing the Q-value, thereby sidestepping the problem of reward scales that vary significantly across environments (e.g., environments with large-magnitude returns versus those with much smaller returns). A regression problem that must match absolute values tends to produce loss magnitudes that differ greatly across differently scaled environments, unbalancing the training signal, whereas a relative-order judgment passed through a sigmoid function yields a loss of the same form regardless of scale. The authors' prior work (Section 2.2) confirmed that this pairwise comparison provides a more stable training signal than regression-based methods, and this study makes direct use of this property.

The total training loss of the proposed method (Pairwise-DT) is the weighted sum of the following three terms:

$$L_{total} = L_{DT} + \lambda_{pair}\cdot L_{pair} + \lambda_{critic}\cdot L_{critic}, \tag{2}$$

where L_{DT} denotes the DT action prediction MSE loss, L_{critic} denotes the MSE loss that regresses the critic's predicted Q-value onto the empirical return-to-go (RTG), and L_{pair} denotes the pairwise classification loss defined in this section. Because the ranking criterion for the pairwise term uses the critic's current prediction with its gradient detached, it functions not as a ground-truth label but as a self-referential auxiliary signal that reinforces the relative order using the model's own estimate. This contrasts with the Pairwise-FT on the LLM alignment side, which was trained with L_{pair} alone.

MGDT's binary classifier (comparison baseline):

$$\mathcal{L}_{\mathrm{MGDT}} = -\mathbb{E}_{(s,a,y)\sim\mathcal{D}}[y\log f_\theta(s,a) + (1-y)\log(1-f_\theta(s,a))],$$

where $f_\theta(s,a)$ denotes the probability of being an expert output by the MGDT classifier, and $y \in \{0, 1\}$ denotes the expert/non-expert label. Here, (s,a) is not a ranked pair as in Pairwise-DT, but an individual transition drawn directly from D.

Ignoring the relative quality between actions: The MGDT's binary classifier assigns each action with an absolute expert/non-expert label. Therefore, if two actions within the same minibatch are both experts or both non-experts, it fails entirely to exploit the relative-merit information between them. In contrast, a pairwise classifier can continuously use which of the two is superior as a training signal, even when the quality of both actions is high or low, providing a denser training signal on offline datasets with a wide quality distribution.

3.2. LLM Alignment-Side Formulation

In the RLHF preference learning, for a response pair (r^+, r^-), r^+ is preferred. The standard RLHF pipeline first learns a separate reward model from preference data following the Bradley–Terry model and then uses this reward model to re-optimize the policy via RL in a two-stage procedure. This procedure requires two separate training loops, reward-model training and policy optimization, which incur substantial implementation complexity and computational cost. The DPO compresses this two-stage procedure into a single optimization, but at the cost of maintaining a fixed reference policy throughout training and performing a forward pass over it at every step.

DPO objective (comparison baseline):

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}\left[\log \sigma\left(\beta \log \frac{\pi_\theta(r^+ \mid x)}{\pi_{\text{ref}}(r^+ \mid x)} - \beta \log \frac{\pi_\theta(r^- \mid x)}{\pi_{\text{ref}}(r^- \mid x)}\right)\right]$$

where π_θ denotes the policy being trained, π_{ref} denotes the reference policy that is fixed throughout training, β controls the strength of the KL divergence constraint, σ denotes the sigmoid, and x denotes the prompt.

Binary preference classifier (comparison baseline):

$$L_{bin} = -E_{(x,r,y)\sim D}[\, y \cdot log\, \sigma(f_\theta(x,r)) \;+\; (1-y) \cdot log(1-\sigma(f_\theta(x,r)))\,]$$

where $y \in \{0,1\}$ denotes the chosen/rejected label, and f_θ shares the same GPT-2 backbone and scalar-head structure as Pairwise-FT, but classifies each response independently, without pairwise comparison.

The Binary preference classifier was chosen as a comparison baseline in order to apply, to LLM alignment, the same logic as MGDT's binary classifier (Section 3.1). Because this baseline, which classifies chosen/rejected as an absolute label, shares the same backbone and reference-free property as Pairwise-FT, it serves as a control that allows the performance difference between the two methods to be attributed solely to "the presence or absence of a pairwise comparison structure." In other words, if DPO functions as the comparison baseline for isolating "whether a method is reference-free," the binary preference classifier functions as the baseline for isolating "the effect of the pairwise structure itself".

Proposed Pairwise-FT objective:

$$\mathcal{L}_{\text{LLM}} = -\mathbb{E}_{(x,r^+,r^-)\sim\mathcal{D}}\left[\log \sigma\left(\frac{f_\theta(x,r^+) - f_\theta(x,r^-)}{\tau}\right)\right] \tag{3}$$

where D denotes an LLM preference dataset of the form $(x,\ r+,\ r-)$ and τ is the same temperature as in Section 3.1.

The proposed Pairwise-FT trains directly based on the difference in preference scores between a response pair without a reward model training stage or a reference policy. This not only simplifies the implementation, but also yields the practical benefit of reducing the computational cost by lowering the number of forward passes required per step (measured comparison in Section 4.3).

3.3. Unified Framework

Abstracting the two objectives shows that they reduce to the same abstract pairwise classification problem under a shared notation, rather than being literally identical operations: the LLM side compares two responses under an exactly shared prompt, whereas the RL side approximates this via within-batch ranking rather than an exactly shared state (Section 3.1).

$$\mathcal{L}_{\text{pair}} = -\mathbb{E}_{(c,o^+,o^-)\sim\mathcal{D}}\left[\log\sigma\left(\frac{g_\theta(c,o^+) - g_\theta(c,o^-)}{\tau}\right)\right] \quad (4)$$

where c denotes the context, o^+/o^- are the candidate pair being compared, and g_θ denotes the domain-agnostic unified scorer; it corresponds to Q_θ in offline RL and to f_θ in LLM alignment. The meaning of each notation is listed in Table 1. Figure 2 shows the corresponding unified pairwise objectives. Minimizing $\mathcal{L}_{\text{pair}}$ encourages the scoring function g_θ to assign a higher score to the preferred candidate o^+ than to o^-, following the same pairwise logistic formulation as the Bradley–Terry model [6]. The temperature τ controls the sharpness of this preference. Thus, Q-value ordering in offline RL and response-preference ordering in LLM alignment can be expressed through the same optimization objective.

Table 1. Notation correspondence across offline RL and LLM alignment.

Symbol	Offline RL	LLM Alignment	Meaning
c(context)	transition (s,a)	prompt x	conditioned context
o^+(preferred)	superior action a^+	preferred response r^+	positive candidate
o^-(inferior)	inferior action a^-	non − preferred response r^-	negative candidate
$g_\theta(c,o)$	Q − value $Q_\theta(s,a)$	preference score $f_\theta(x,r)$	estimated quality/preference score

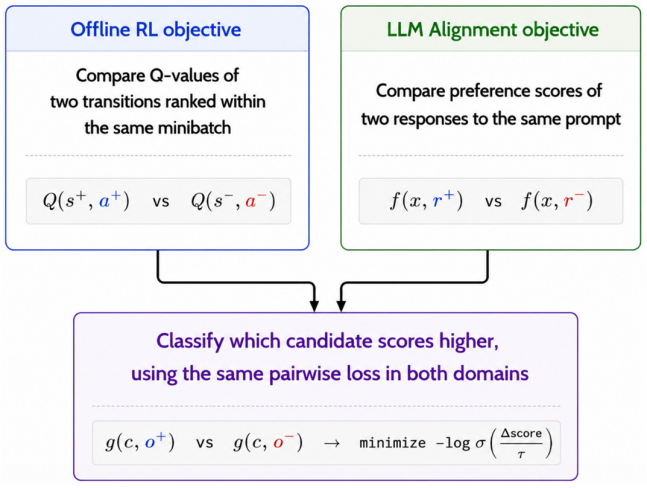


Figure 2. Formal Correspondence and Unified Pairwise Objective. The offline RL objective (comparing Q-values of two transitions ranked within the same minibatch) and the LLM alignment objective (comparing preference scores of two responses to the same prompt) are unified into a single pairwise classification rule: given a context c and a candidate pair (o^+, o^-), the model is trained to classify which candidate scores higher via $g(c,o^+)$ vs. $g(c,o^-)$, minimizing $-log\ \sigma(\Delta score/\tau)$. The correspondence table in Section 3.3 lists the exact mapping between domain-specific notation $(s^+/s^-,\ a^+/a^-,\ Q_\theta;\ x,\ r^+/r^-,\ f_\theta)$ and this unified notation $(c,\ o^+/o^-,\ g_\theta)$.

The core claim of this section is as follows: even across different domains, the structure of "comparing the relative quality of two options given a shared context" reduces to the same abstract form, and empirical results suggest that pairwise ranking often provides a

denser signal than absolute binary labels: Pairwise-FT outperformed Binary on 4/5 seeds, whereas Pairwise-DT versus MGDT was mixed across MuJoCo environments (Section 4).

Whereas prior work has independently developed pairwise formulations within each domain, the formulation in this study explicitly shows that the two objectives ($\mathcal{L}_{\mathrm{RL}}$, $\mathcal{L}_{\mathrm{LLM}}$) are instances of the same abstract pairwise optimization problem.

This abstraction is meaningful not merely because the two loss functions "look similar", but because mapping three components—context, candidate pair, and score function—onto the same positions makes them reduce to the same abstract equation. Offline RL and LLM alignment use different training signals (Q-value, preference score), but the structure that reduces that signal to "which of the two candidates is superior" is itself the same. This study demonstrates this structural correspondence independently for each domain, as described in Section 4.

However, in offline RL, owing to constraints of the offline dataset, o^+ and o^- do not strictly share the same context (state) and are instead approximated by relative ranking based on critic predictions within a minibatch (see Section 3.1). In LLM alignment, by contrast, o^+ and o^- share exactly the same prompt x. This asymmetry does not fully invalidate the structural correspondence between the two domains; however, the fact that the RL side is an approximate form is noted as a limitation in Section 5.5.

The unified formula presented in Section 3.3 clarifies the difference between the DPO and the proposed method.

$$\mathcal{L}_{\mathrm{DPO}} : \beta \log \frac{\pi_\theta(r^+ \mid x)}{\pi_{\mathrm{ref}}(r^+ \mid x)} - \beta \log \frac{\pi_\theta(r^- \mid x)}{\pi_{\mathrm{ref}}(r^- \mid x)}$$

$$\mathcal{L}_{\mathrm{LLM}} : f_\theta(x, r^+) - f_\theta(x, r^-)$$

DPO compares the relative log-probability difference against a reference model—that is, it depends on π_{ref}. The proposed method only highlights the direct-preference-score difference—π_{ref} is unnecessary. This distinction is not merely an implementation detail but constitutes an additional methodological contribution: "removal of dependence on a reference model."

3.4. Algorithm Description

Pairwise DT and FT differ only in the domain and share the same training procedure. Algorithm 1 describes this procedure in a domain-agnostic form, with the concrete correspondences for each domain noted in parentheses.

Algorithm 1. Unified pairwise classification training (Offline RL/LLM alignment)

Input: Offline dataset D (transition triples for RL; prompt–response triples for LLM)
Batch size B, total training steps K, temperature τ
Output: Learned parameters θ of scorer g_θ (Q-network for RL; preference head for LLM)
1: Initialize network parameters θ
2: for step k = 1 to K do
3: Sample a batch of triples (c, o+, o−) ~ D
RL : c ≈ (top/bottom Q − ranked transitions in batch; s+ ≠ s− in general, see Section 3.1)
LLM: c = prompt x, (o+, o−) = response pair (r+, r−)
4: Compute scores g_θ(c, o+), g_θ(c, o−)
RL: g_θ = Q_θ(s, a) (value head over state-action)
LLM: g_θ = f_θ(x, r) (preference head over prompt-response)

5: Compute pairwise loss:
L_pair = −log σ((g_θ(c, o+) − g_θ(c, o−))/τ)
6: Update θ via gradient descent on L_total (RL: L_DT + λ_pair·L_pair + λ_critic·L_critic; LLM: L_pair only)
7: end for
8: return θ
Domain instantiation:
Pairwise-DT (RL): θ parameterizes a DT-backbone value head; D is the offline replay buffer of (s, a, r, s′) transitions grouped into pairs.
Pairwise-FT (LLM): θ parameterizes a GPT-2 backbone with a scalar preference head; D is the HH-RLHF chosen/rejected pair dataset.

Step 3 (pair construction), rewritten as domain-specific sub-steps:

RL: 3a. Sample a minibatch of transitions from the offline replay buffer. 3b. Rank transitions within the batch by the critic's current Q-prediction. 3c. Split into top/bottom halves to form (s+, a+) and (s−, a−); note s+ $\neq$ s− in general.

LLM: 3a. Sample a minibatch of (prompt, chosen, rejected) triples directly from the preference dataset D [17]; no ranking or construction step is needed, since the pair is already given.

In Step 4 (score computation), the two domains use different backbones: Pairwise-DT attaches a scalar value head on top of a Decision Transformer backbone to produce $Q(s, a)$, whereas Pairwise-FT attaches a scalar preference head on top of a GPT-2(124 M) backbone to produce $f(x, r)$. Both cases are reduced to the same abstract structure in that they project the backbone's final hidden state into a single scalar, and this scalar value is substituted into the pairwise loss in Step 5 in the same manner.

DT and GPT-2 were selected as backbones because they are standard baselines widely used in the offline RL and LLM alignment domains, respectively. They share the same backbone as the comparison baselines DT/MGDT, and GPT-2 shares the same backbone as the comparison baselines DPO/binary preference. Therefore, the backbone differences are not confounded with the results. However, the proposed pairwise loss function does not depend on any particular backbone architecture (it is algorithm-agnostic) and, in principle, can be applied to other architectures (e.g., larger language models or CQL/IQL-based critics).

Step 6 (parameter update) uses the AdamW optimizer [18] for both domains: RL is trained for 5000 iterations (batch size following the minibatch construction in Section 3.1), and LLM is trained for 5000 iterations (batch size of 8 for each selected/rejected). Concrete hyperparameters, such as the learning rate and batch size, are specified per domain in Section 4.1 (Experimental Setup).

4. Results

4.1. Common Experimental Setup

- Software: The LLM experiments were implemented in PyTorch [19] (V2)+ Hugging Face Transformers [20] (GPT-2)(V4), and the MuJoCo experiments used an in-house Decision Transformer implemented in PyTorch with a Gymnasium (MuJoCo-v4) (V0) [21,22] environment and Minari [23] (V0)(a loader for the D4RL [24] medium-v2 dataset).
- Hardware: Both the LLM and MuJoCo experiments were run on the same Google Colab A100 GPU.
- MuJoCo uses the standard D4RL normalized score (a metric that normalizes the performance of the random and expert policies to 0 and 100, respectively), and LLM

uses reward accuracy (the proportion of the test set on which the selected response received a higher score than the rejected response).

- Rationale for baseline selection: The specific rationale for selecting DT/MGDT (offline RL) and DPO/binary preference (LLM alignment) as comparison baselines is provided in each "comparison baseline" definition in Sections 3.1 and 3.2.
- RTG normalization for stable gradient training, the RTG values were normalized to each environment's target return prior to critic training (Section 3.1).
- LLM data preprocessing (truncation): Because the HH-RLHF chosen/rejected pairs differed only in the final response of the conversation, sequences exceeding 512 tokens were truncated from the front such that the final response, rather than the conversation history, was preserved. The common approach of truncating from the back was not adopted because it risks losing the final response itself, which distinguishes chosen from rejected.
- DPO log-probability computation: π_θ (the policy being trained) and π_{ref} (the reference policy that remained fixed throughout training) log-probabilities were computed as the average over the actual (non-padding) token span, rather than the sum over the whole sequence (see the DPO objective in Section 3.2).

All three methods (Binary, Pairwise-FT, DPO) were trained with the same standard configuration (learning rate, batch size, number of iterations, AdamW optimizer) without per-method hyperparameter tuning, so as to compare methods under matched conditions rather than each method's individually optimized best case. We note here, and revisit in Section 5.5, that the difference in seed count between the two domains (three seeds for MuJoCo, five seeds for LLM) was not planned in advance; it arose after observing Binary's relatively high seed-to-seed variance (± 0.021) and a seed = 1 reversal case, which led us to extend only the LLM side to five seeds post hoc.

4.2. Validation 1 Results (MuJoCo)

The purpose of this section is to reconfirm that pairwise-DT retains its environment-mean advantage over DT in a new environment within the offline RL Domain A establishing a "Domain A" baseline to place alongside the LLM-domain validation in Section 4.3. The superiority of Pairwise-DT itself is a result already established in the author's prior work (Section 2.2). The novelty of this section lies not in reproducing that result, but in directly mapping it to the LLM-domain findings in Section 4.4, thereby demonstrating the cross-domain consistency of the pairwise principle.

The DT, MGDT, and pairwise-DT were trained for three seeds in three environments: HalfCheetah, Hopper, and Walker2d (medium-v2, via Minari) and evaluated using the D4RL normalized score. The DT backbone is a small transformer with a hidden size of 128, three layers, and one head (under approximately 1M total parameters), which is considerably smaller than GPT-2 (124 M). DT/MGDT were selected as comparison baselines because, as explained in Section 3.1, DT is a standard baseline that uses no pairwise structure, whereas MGDT is an intermediate stage that uses binary expert/non-expert classification. This experiment used the hyperparameter τ defined in Section 3.1 (the sigmoid's sensitivity to the difference between the two scores) at 1.0 (the neutral value). A τ sensitivity analysis (HalfCheetah, $\tau \in \{0.5, 1.0, 2.0\}$) is provided in Appendix C.

Table 2 summarizes the mean normalized score and standard deviation for the DT, MGDT, and Pairwise-DT in each environment, providing the same data as the bars and error bars in Figure 3 in tabular form.

Table 2. Normalized Score by Environment (mean ± population std, three seeds).

Environment	DT	MGDT	Pairwise-DT
HalfCheetah	57.23 ± 6.52	61.21 ± 4.55	59.91 ± 4.06
Hopper	97.41 ± 1.95	100.79 ± 3.30	101.56 ± 0.55
Walker2d	82.26 ± 15.09	84.80 ± 25.63	83.70 ± 20.94

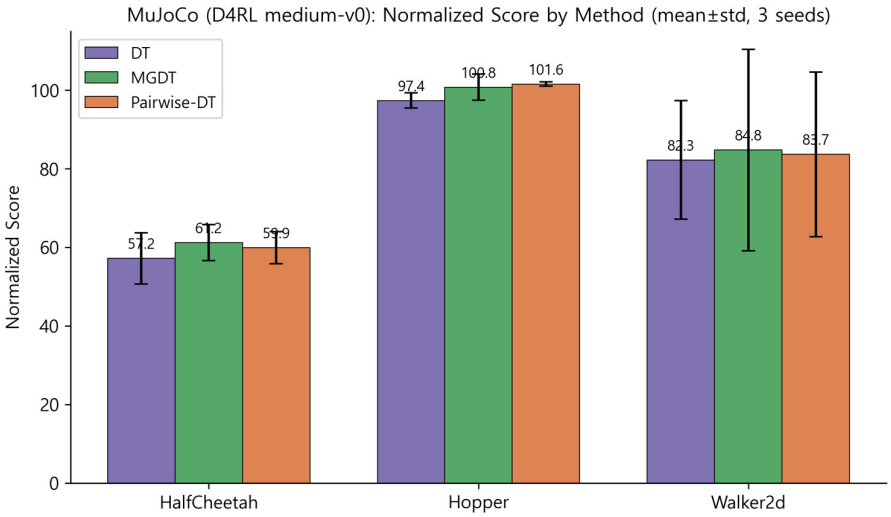


Figure 3. Comparison of Methods by MuJoCo Environment (bars + error bars, three seeds).

Figure 3 shows the mean normalized scores for DT, MGDT, and pairwise-DT across the three environments HalfCheetah/Hopper/Walker2d as bars, with the standard deviation shown as error bars. The error bars are narrowest for Hopper and widest for Walker2d, visually clarifying the difference in result stability across environments. Pairwise-DT showed the highest mean (101.56) and lowest variance (±0.55) on Hopper. On HalfCheetah, the MGDT held a slight edge ($61.21 > 59.91$), and on Walker2d, the error bars of the three methods overlapped substantially, making them statistically difficult to distinguish (std at the 15–26 level). A Welch's *t*-test confirms that neither comparison is statistically significant (Pairwise-DT vs. DT: $t = 0.079$, $p = 0.941$; vs. MGDT: $t = -0.047$, $p = 0.965$), quantitatively supporting the qualitative observation that Walker2d's large variance prevents these methods from being statistically distinguished.

To address the reviewer's concern regarding the self-referential nature of the critic-based pairing (Section 3.1), we conducted an additional pairing ablation study, comparing (a) the original critic-Q pairing used throughout this paper (baseline), (b) pairing based on the ground-truth return-to-go rather than the critic's own prediction (RTGpair), and (c) removing the pairwise term entirely, i.e., training with $L_DT + \lambda_critic \cdot L_critic$ only (no-pair). Each condition was evaluated on HalfCheetah and Walker2d across three seeds (0, 1, 2) with $\tau = 1.0$ (Table 3). In both environments, the no-pair condition scored higher on average than the baseline (+2.2 on HalfCheetah, +2.85 on Walker2d). Notably, on HalfCheetah the no-pair condition also had a lower standard deviation than baseline (1.63 vs. 4.06), i.e., it was better on both the mean and the variance; on Walker2d, no-pair had a higher mean but also a higher variance (26.45 vs. 20.94), so the advantage is less clear-cut. A Welch's *t*-test showed that the difference between baseline and no-pair was not statistically significant in either environment (HalfCheetah: $t = -0.712$, $p = 0.535$; Walker2d: $t = -0.120$, $p = 0.911$); however, with only three seeds, this should be read as "the direction cannot be confirmed with this sample size" rather than "there is no difference". In summary, this ablation does not resolve the reviewer's concern that critic-Q pairing might artificially inflate performance through a confirmation loop; if anything, the

HalfCheetah result suggests the opposite may be true, and we candidly report that the marginal contribution of the pairwise term itself is not well established within this hybrid architecture (DT action-prediction loss + critic regression + pairwise term). This contrasts with the LLM side (Pairwise-FT), which is trained with the pairwise loss alone; there, a within-domain comparison against Binary (Section 3.2, an otherwise-identical baseline without pairwise comparison) offers more direct, though still not conclusive, evidence for the pairwise term's usefulness (Section 5.1). The RL-side result here should be understood in the context of the authors' prior single-domain work (Section 2.2) rather than as new evidence for the pairwise term's necessity. We discuss this asymmetry further as a limitation in Section 4.5 and Section 5.1.

Table 3. Pairing ablation results.

Environment	Condition	Normalized Score (Mean ± Population std, 3 Seeds)
HalfCheetah	baseline (critic-Q pairing)	59.91 ± 4.06
HalfCheetah	RTGpair (ground-truth return-to-go)	54.96 ± 3.41
HalfCheetah	no-pair (pairwise term removed)	62.11 ± 1.63
Walker2d	baseline (critic-Q pairing)	83.70 ± 20.94
Walker2d	RTGpair (ground-truth return-to-go)	82.71 ± 25.42
Walker2d	no-pair (pairwise term removed)	86.55 ± 26.45

A multi-seed τ sensitivity analysis on HalfCheetah (three seeds per τ, Table 4; see also Table A2 in Appendix C) confirms that $\tau = 1.0$ achieves the highest mean among the tested values, consistent with the single-seed estimate reported earlier. However, the range across τ (8.81) is about 2.17× the baseline's three-seed standard deviation (4.06), larger than the single-seed estimate suggested (about 1.39×), so we no longer describe MuJoCo results as robust to τ regardless of choice; instead, $\tau = 1.0$ is directionally favored but the margin should not be overstated. Notably, at $\tau = 2.0$, Pairwise-DT (51.10 ± 1.66) falls clearly below MGDT (61.21 ± 4.55, Table 2) on a multi-seed basis, replacing our earlier single-seed observation (seed = 0, 54.85) with more robust evidence of the same conclusion. Throughout this paper, we limit our claim to "Pairwise-DT exceeds DT on an environment-mean basis"; comparisons against MGDT are reported only as a secondary, environment-dependent observation (Pairwise-DT exceeds MGDT on some environments/seeds but not others) rather than as a general claim of superiority.

Table 4. Temperature sensitivity (MuJoCo: HalfCheetah, Pairwise-DT).

τ	Normalized Score (Mean ± Population std, n Seeds)
0.5	56.28 ± 3.42 (n = 3, seeds 0/1/2)
1.0 (baseline)	59.91 ± 4.06 (n = 3, seeds 0/1/2)
2.0	51.10 ± 1.66 (n = 3, seeds 0/1/2)

In summary, pairwise-DT > DT consistently held across all three environments, whereas the advantage over MGDT varied by environment. The overall trends in the learning curves are presented in Appendix A Figure A1. Walker2d exhibited the largest variance among the three methods (standard deviations at the 15–26 level). The error ranges overlap substantially across methods, making them statistically difficult to distinguish; a Welch's *t*-test (which does not assume equal variances) confirms this quantitatively: Pairwise-DT vs. DT, t = 0.079, $p = 0.941$; Pairwise-DT vs. MGDT, t = −0.047, $p = 0.965$ (both two-sided, not significant). See Appendix A Figure A2 for detailed per-seed distribution. Nonetheless, on an environment-mean basis, the pairwise-DT remained above the DT across all three environments. Note that this result was obtained from the loss structure

defined in Section 3.1 (a weighted sum of three terms in which the pairwise term acts jointly with the DT's action prediction loss and the critic's regression loss). Despite the approximation introduced in the offline RL pairing process (Section 3.1, the $s^+ \neq s^-$ approximation), the pairwise formulation consistently outperformed DT on an environment-mean basis across all three environments. However, as discussed above, the pairing ablation indicates that this environment-mean advantage cannot be attributed with confidence to the pairwise term specifically, given the DT + critic + pairwise architecture's confounded contributions (Section 4.2, Table 3).

4.3. Validation 2 Results (LLM Alignment)

GPT-2 (124 M) was trained on the Anthropic HH-RLHF dataset (helpful-base, train 20*k*/test 2*k*) to compare the binary, Pairwise-FT, and DPO across five seeds. The primary evaluation metric was the Reward Accuracy, with training time (on an A100) reported as an efficiency metric. DPO/binary preference was selected as the comparison baseline because, as explained in Section 3.2, DPO represents the current standard method that depends on a reference model, while binary preference serves as a control that trains on absolute labels without a pairwise structure. This experiment used the hyperparameter τ at 1.0, the same as in Section 3.2. Details of the τ sensitivity analysis are provided in Appendix C. A multi-seed τ sensitivity analysis (Appendix C) confirms that this choice is not arbitrary; this robustness holds strongly on the LLM side, while on the MuJoCo side (Section 4.2) the direction is consistent but less pronounced.

Table 5 summarizes each method's reward accuracy and training time as mean ± standard deviation, corresponding to the source data behind Figure 4 (accuracy) and Figure 5 (training time). To support this conclusion beyond the sign test, we additionally report a paired two-sided *t*-test (t = 4.388, $p = 0.0118$), a 95% confidence interval on the mean paired difference ([0.0134, 0.0598], excluding zero), and a 10,000-resample paired bootstrap (95% CI [0.0225, 0.0510], with 100% of resamples showing a positive mean difference) (Appendix B).

Table 5. Reward Accuracy and Training Time by Method (mean ± population std, five seeds).

Method	Reward Accuracy	Training Time (s)
Binary	0.660 ± 0.021	1904 ± 4
Pairwise-FT	0.678 ± 0.013	1949 ± 3
DPO	0.642 ± 0.005	3715 ± 5

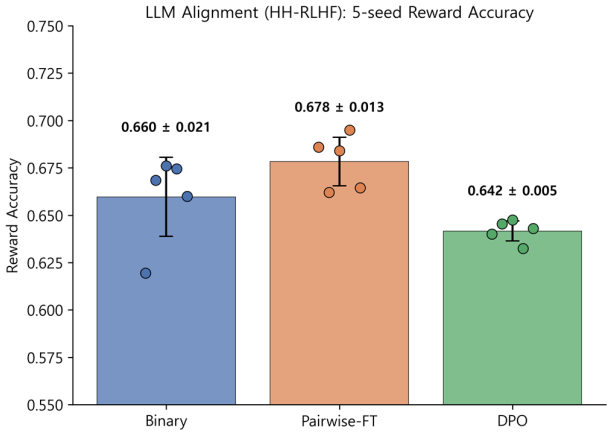


Figure 4. LLM Alignment: Five-Seed Reward Accuracy Comparison (bars + individual seed points).

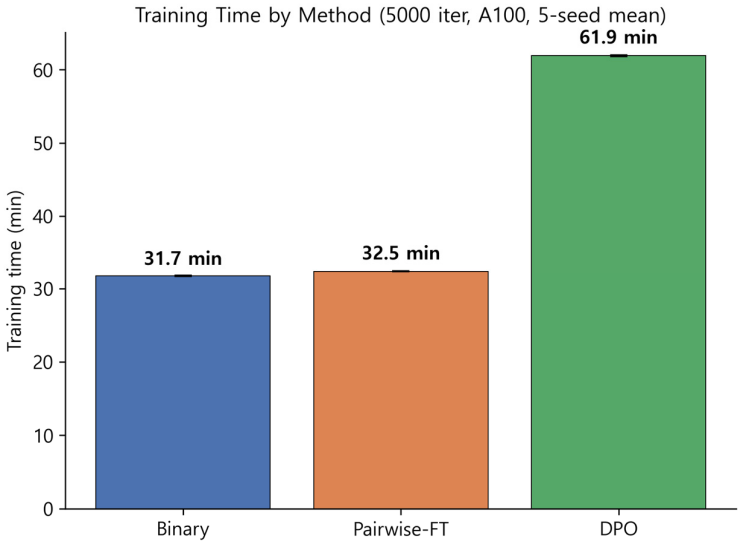


Figure 5. Comparison of Training Time by Method.

Figure 4 shows the five-seed mean reward accuracy for the binary, Pairwise-FT, and DPO, shown as bars, with error bars and individual seed values plotted as points. The seed points for Pairwise-FT cluster more tightly than those for binary. DPO exhibits the smallest seed-to-seed spread overall, but its mean accuracy is lower. Taken together, these results show that Pairwise-FT achieves superior mean performance while remaining more stable than binary.

Figure 5 compares the mean training times (in minutes; see Table 5 for values in seconds) of the three methods. The DPO bar is markedly higher than that of Binary/Pairwise-FT, visually clarifying the difference in computational cost due to the forward pass of the reference model.

Pairwise-FT outperformed binary on four of five seeds (with a narrow loss on one seed), and compared with DPO, it showed higher accuracy on all five seeds while training in approximately 1.9x less time (1949 s *vs*. 3715 s). This shows that reference-free training can match or exceed DPO's reward accuracy while incurring reduced cost specifically attributable to removing the reference-model forward pass, rather than a general reduction in the cost of LLM alignment as a whole. To clarify the source of this efficiency gain, Table 6 reports the trainable/frozen parameter counts and the number of forward passes per training step for each method. Binary and Pairwise-FT each require two forward passes per step (chosen and rejected responses through the single GPT-2 model), whereas DPO requires four (chosen and rejected responses through both the policy and the frozen reference model)—exactly twice as many, consistent with the observed ~1.9× training-time difference.

Table 6. Parameter Counts and Forward Passes per Step by Method.

Method	Trainable Parameters	Frozen Parameters	Forward Passes per Step
Binary	GPT-2 (124 M)	—	2 (chosen, rejected)
Pairwise-FT	GPT-2 (124 M)	—	2 (chosen, rejected)
DPO	GPT-2 (124 M) policy	GPT-2 (124 M) reference	4 (policy ×2, reference ×2)

DPO's markedly lower variance (0.005 vs. 0.013 for Pairwise-FT) is consistent with its KL constraint against a fixed reference policy acting as an implicit regularizer that limits how far the policy can drift across seeds; Pairwise-FT's reference-free training removes this constraint, trading some of this stability for a higher average reward accuracy (Section 5.2).

Appendix B Figure A3 shows the individual per-seed learning curves. Binary exhibited the largest seed-to-seed variation of the three methods (standard deviation ± 0.021). The relationship between the accuracy and training time is shown in Appendix B Figure A5 (Pareto plot). Unlike the DPO, the pairwise-FT achieves this performance without maintaining a frozen reference model, indicating that explicit reference-based regularization is not necessary for accurate pairwise preference learning.

As a supplementary check on generation quality, we conducted a lightweight comparison using Claude (claude-sonnet-5) as an LLM judge. For 100 prompts sampled from the HH-RLHF test set, we compared (a) best-of-four candidates selected by Pairwise-FT/Binary from base GPT-2 generations against (b) a single response sampled from the DPO policy (seed = 0 checkpoints). As shown in Table 7, Pairwise-FT was judged better in 97/100 comparisons (two ties, one DPO win), and Binary in 96/100 (three ties, one DPO win). We caution against over-interpreting these win-rates: unlike the reward-accuracy comparisons in Table 5, this comparison is confounded by a methodological asymmetry—Pairwise-FT/Binary select the best of four candidates using a trained scorer, whereas DPO generates only a single sample from its own policy. Best-of-N selection is known to substantially improve apparent quality independent of the underlying scorer's fidelity, so this result should not be read as evidence that Pairwise-FT/Binary produce better-aligned text than DPO in a fair generation setting; we report it only as a preliminary, heavily caveated signal that the learned scorers are not obviously misaligned with human-judged response quality. A fair comparison would require sampling an equal number of candidates from each policy, which we leave to future work (Section 5.6). In addition, the judge model (claude-sonnet-5) does not support the temperature parameter at the time of this study (confirmed via direct API testing: the parameter is rejected as deprecated for this model); judge calls were therefore made using the API's default sampling behavior rather than a fixed temperature = 0, which may introduce minor non-determinism in individual verdicts. This is reported as a further limitation in Section 4.5. Interpreted through the lens of downstream policy optimization, this preliminary result offers, at best, necessary-condition-level evidence: it suggests the learned scorers are not obviously misaligned with human-judged quality, which is a prerequisite (though not sufficient on its own) for using them as reward models in a full policy-optimization pipeline (e.g., PPO); a sufficient demonstration would require the full downstream connection left to future work (Section 5.6). Full per-comparison results are provided in Appendix D.

Table 7. Generation-Quality Check (LLM Judge).

Comparison	Win (n, %)	Tie	DPO Win
Pairwise-FT vs. DPO	97 (97%)	2	1
Binary vs. DPO	96 (96%)	3	1

A multi-seed τ sensitivity analysis (three seeds per off-baseline τ, Table 8; see Table A3 in Appendix C) confirms that $\tau = 1.0$ achieves the highest mean reward accuracy among the tested values. The range across τ (0.0131) is about $1.02\times$ the baseline's five-seed standard deviation (0.0129), indicating that this domain's results are robust to the choice of τ within the tested range, consistent with the single-seed estimate reported earlier.

Table 8. Temperature Sensitivity (LLM, Pairwise-FT).

τ	Reward Accuracy (Mean ± Population std, n Seeds)
0.5	0.6728 ± 0.0038 (n = 3, seeds 0/1/4)
1.0 (baseline)	0.6783 ± 0.0129 (n = 5, seeds 0/1/2/3/4)
2.0	0.6652 ± 0.0031 (n = 3, seeds 0/1/4)

4.4. Synthesis from the Unified-Framework Perspective

By synthesizing the results from both domains, the objective proposed in this study $\mathcal{L}_{\text{pair}}$, achieved competitive or superior performance compared to each domain's alternatives (comparison baselines: DT/MGDT, DPO/binary preference) in both offline RL and LLM alignment. However, the two results differ in character: the MuJoCo (Domain A) result is a reconfirmation of a result already established in the author's prior work, whereas the LLM (Domain B) result was newly validated in this study.

- Pairwise-FT $\geq$ DPO (LLM, Domain B, Figure 6): gap ≥ 0 on 5/5 seeds, $+3.7$ pp, $\sim 1.9x$ faster, reference-free—the central claim newly validated in this study.

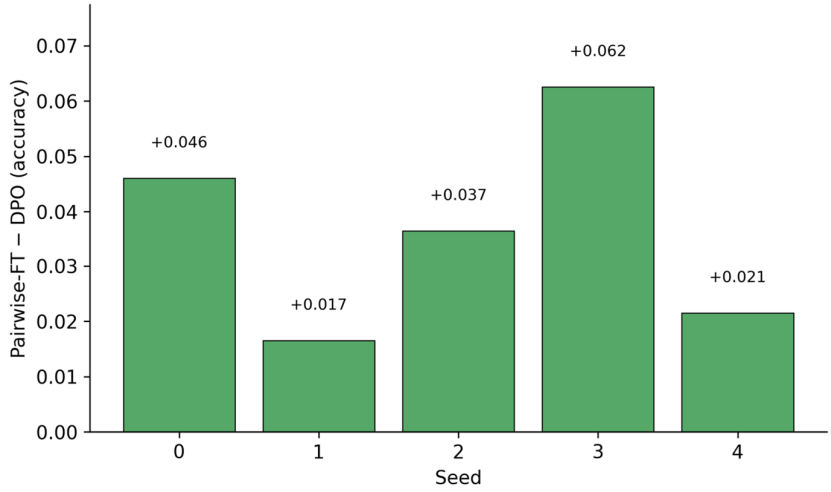


Figure 6. Pairwise-FT−DPO gap (by seed, 5/5 seeds ≥ 0).

- Pairwise > DT (MuJoCo, Domain A, Figure 7): gap > 0 in all three environments—the most statistically consistent result, but a reconfirmation of a result already established in the author's prior work (Section 2.2); its main role is as corresponding evidence that, placed alongside the LLM result, shows "the same structure holds in both domains".
- Pairwise > MGDT/binary: mixed depending on the environment/seed, used only as supporting evidence in both domains.

That Pairwise-FT outperformed DPO on all five seeds corresponds, from a one-sided sign-test perspective, to $p = 1/2^5 = 1/32 \approx 0.03$, suggesting that this result is unlikely to be due to chance; the two-sided paired t-test, confidence interval, and bootstrap reported above provide corroborating evidence not subject to this one-sidedness concern. However, for MuJoCo (three seeds, with scales differing by environment), the sample was small, and pooling across environments was difficult; therefore, a separate statistical test was not applied.

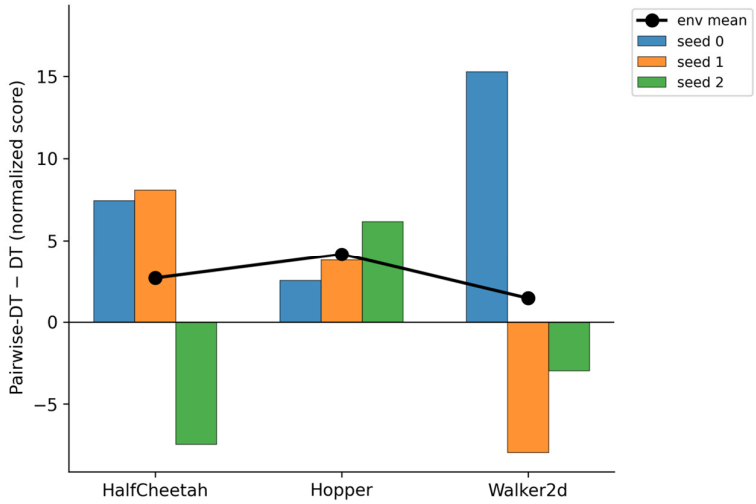


Figure 7. Pairwise-DT−DT gap (environment × seed, env mean > 0 in all three).

4.5. Experimental Limitations

- Only the HH-RLHF helpful base subset was used, with a training subsample of 20k.
- Reward Accuracy reflects reward-modeling performance rather than actual generation quality; the preliminary generation-quality check in Section 4.3 (Table 7) partially addresses this but has its own limitations, detailed below.
- The experiment was conducted at the scale of GPT-2 (124 M) with 5000 iterations; generalization to larger models or token counts was not confirmed.
- MuJoCo used three seeds while LLM used five seeds, as noted in Section 4.1; this difference was not planned in advance but arose after observing Binary's seed-to-seed variance ($\pm$0.021) and a seed = 1 reversal case, leading us to extend only the LLM side to five seeds post hoc.
- Loss–accuracy decoupling was observed in which loss and Reward Accuracy do not always move together (Appendix B Figure A4); therefore, train loss was used only as an auxiliary metric.
- (1) We did not evaluate a random-pairing ablation due to computational constraints, leaving it as a natural complement to the RTG-pairing and no-pair conditions reported in Section 4.2. (2) Our RL-side comparisons are limited to DT/MGDT, and our LLM-side comparisons to DPO/Binary; comparisons against value-based methods such as CQL/IQL or more recent reference-free preference-learning methods were not performed. (3) We did not measure fine-grained computational metrics such as GPU memory, tokens/second, or FLOPs; the reported training-time differences (Section 4.3) reflect wall-clock time on a single Google Colab A100 GPU only. (4) The lightweight generation-quality check reported in Section 4.3 (Table 7) is confounded by a methodological asymmetry between best-of-four selection (Pairwise-FT/Binary) and single-sample generation (DPO), and should be read only as a preliminary signal rather than definitive evidence of superior generation quality. In addition, the judge model (claude-sonnet-5) did not support the temperature parameter at the time of this study, so judge calls used the API's default sampling rather than a fixed temperature = 0.

5. Discussion

This study extended the pairwise classification structure from a single domain of offline RL to LLM alignment and empirically demonstrated that this structure generalizes across domains, finding domain-dependent support: strong on the LLM side, more limited on the RL side. This section analyzes the implications of the experimental results and discusses the limitations of the proposed method and directions for future work.

5.1. Why Does the Pairwise Formulation Work in Both Domains?

The experimental results suggest that the pairwise formulation works in both domains because of scale invariance.

Scale Separation Across Domains: Q-value (RL) and preference score (LLM) differ fundamentally in scale and meaning. Q-values vary significantly with the reward structure of each environment (e.g., HalfCheetah's large-magnitude returns versus Hopper's small returns), whereas a preference score produced by GPT-2's scalar head has no absolute meaning on its own. Had we attempted to directly compare or regress the absolute values across the two domains, it would have been difficult to define a unified loss function because of their different scales and semantics. By contrast, pairwise comparison requires only the relative ordering within the same context. Thus, whether the signal is a Q-value or a preference score, the question reduces to a common form: "which of the two is larger?" Owing to this relative comparison structure, two scoring functions with different meanings can be unified under the same sigmoid-based loss.

Preservation of Reference-Free Optimization: The fact that performance was maintained without a reference model in both domains is interpreted as arising because the pairwise comparison itself already provides a relatively discriminative signal. DPO's reference model serves as a KL-divergence anchor that prevents the policy from drifting too far during training. By contrast, the Pairwise-FT/Pairwise-DT in this study do not directly update a policy or collect new data; instead, they train a scorer on a fixed offline/preference dataset. Consequently, the failure mode that DPO is designed to prevent—"policy divergence during training"—simply does not arise. This provides a structural explanation for why a reference model is unnecessary in the pairwise formulation.

This explanation, however, does not establish that the pairwise structure is necessary in both domains. The pairing ablation in Section 4.2 found that, on HalfCheetah, removing the pairwise term (no-pair) achieved both a higher mean and a lower variance than the baseline, and no statistically significant difference was detected in either environment. That is, within this hybrid architecture (DT action-prediction loss + critic regression + pairwise term), the marginal contribution of the pairwise term itself is not established at our sample size. The scale-invariance argument above explains why the same pairwise structure can be applied in both domains, not that removing it would necessarily degrade performance on the RL side.

On the LLM side, by contrast, a direct within-domain comparison is available: Binary (Section 3.2) shares the same GPT-2 backbone and scalar head as Pairwise-FT but classifies each response independently rather than comparing pairs, making it the LLM-side analogue of the no-pair condition. Pairwise-FT outperformed Binary in four of five seeds (Table 5: 0.678 ± 0.013 vs. 0.660 ± 0.021), suggesting that the pairwise structure itself is useful in this domain—a claim distinct from, and not to be conflated with, Pairwise-FT's advantage over DPO (Section 4.3, 5/5 seeds, $p = 0.0118$), which speaks to reference-free optimization rather than to the pairwise structure per se. Taken together, the pairwise term's contribution appears domain-dependent in this study: supported (though not conclusively) on the LLM side, and not established on the RL side.

5.2. Why Is a Reference Model Unnecessary?

The DPO imposes a KL divergence constraint relative to the reference policy to prevent reward hacking (a phenomenon in which the policy shifts excessively in a direction that deceives the reward model) during policy optimization. However, pairwise FT does not directly optimize a policy; instead, it trains a scorer (preference classifier) on fixed chosen/rejected pairs in a supervised manner. In this setup, there is no closed loop in which the policy generates new responses during training and is then rewarded for those

responses; therefore, there is no pathway through which reward hacking could arise in the first place. This is the structural reason Pairwise-FT achieves competitive performance with DPO without a reference model, and the empirical results in Section 4.3 (5/5 *seeds*, $p \approx 0.03$) are consistent with this explanation.

This asymmetry also appears as a stability-versus-accuracy trade-off: DPO's KL constraint against a fixed reference policy limits how far the policy can drift, yielding a markedly lower seed-to-seed variance (0.005) at the cost of a lower mean reward accuracy, whereas Pairwise-FT's reference-free training removes this constraint and achieves a higher mean (0.678) with correspondingly higher variance (0.013). We view this as a genuine trade-off rather than an unambiguous advantage for either method: reference-based regularization does provide real stability benefits, which reference-free pairwise classification does not automatically inherit.

5.3. What the Cross-Domain Asymmetry Suggests

The approximation acknowledged in Section 3.1 (that o^+ and o^- do not strictly share the same context in RL) and seed-to-seed variation patterns such as the seed = 1 reversal for Binary and the large variance for Walker2d show that consistency across multiple seeds is a more important evaluation criterion than the best performance of a single run. For example, the Pairwise-DT gap for seed = 0 on Walker2d was $+15.3$ while it was -7.9 for seed = 1 (Appendix A Figure A2), illustrating the risk of judging relative merit from a single seed alone. Therefore, this study reports both the mean and standard deviation across five seeds (LLM) and three seeds (MuJoCo) and does not conceal individual counterexamples across seeds (in three of the nine combinations, three environments $\times$ three seeds, DT actually outperformed Pairwise-DT). What matters is not the best performance of any one seed but the average trend and its variance across multiple seeds.

5.4. Robustness to τ

The multi-seed τ sensitivity analysis ($\tau \in \{0.5, 1.0, 2.0\}$) shows domain-dependent robustness. On the LLM side, the range across τ (0.0131) is about $1.02\times$ the baseline's five-seed standard deviation (0.0129), so the choice of $\tau = 1.0$ is not favorably cherry-picked and results are robust within the tested range. On the MuJoCo (HalfCheetah) side, $\tau = 1.0$ remains the best-performing value on a three-seed basis, but the range across τ (8.81) is about $2.17\times$ the baseline's three-seed standard deviation (4.06)—larger than our earlier single-seed estimate suggested (about $1.39\times$). We therefore no longer claim MuJoCo results hold "regardless of the hyperparameter choice"; instead, $\tau = 1.0$ is directionally favored, and this robustness claim should be read as domain-dependent, stronger on the LLM side than on the RL side.

5.5. Limitations

Limited domain installation: This study validated only two domains: offline RL (three MuJoCo environments) and LLM alignment (one dataset). Whether the principle generalizes to other domains that may have different pairwise structures, such as image generation, preference learning, or recommendation systems, remains to be confirmed.

Limited τ Sensitivity Coverage: Although the τ sensitivity analysis was extended to multiple seeds (Section 5.4), its scope remains limited: the off-baseline values ($\tau = 0.5,\ 2.0$) were each tested with only three seeds, and on the MuJoCo side, the sweep was conducted on a single environment (HalfCheetah) rather than across all three environments tested elsewhere in this study. A more comprehensive sweep across additional seeds and environments is left to future work.

Approximate Context Sharing in the RL Domain: As explained in Section 3.1, in offline RL, $(o^+,\ o^-)$ is approximated by within-batch ranking based on critic predictions and does

not strictly share the same state. This asymmetry contrasts with the strict context sharing on the LLM side.

Model Scale: LLM experiments were conducted at the GPT-2 (124 M) for 5000 iterations. It was not confirmed whether the same trend was reproduced in larger models or with more data.

5.6. Directions for Future Work

- Extension to Larger Language Models: It remains to be confirmed whether the same trend observed here reproduces consistently in models of larger scale (e.g., 1B–7B parameters) from families such as LLaMA or Qwen.
- Connection to Downstream Policy Optimization: An end-to-end validation is needed that connects the learned preference scorer to the actual policy optimization, such as PPO or best-of-n, to evaluate the actual generation quality rather than reward accuracy alone. Extend the lightweight generation-quality check in Section 4.3 to a full downstream policy optimization setting (e.g., using the learned scorer as a reward model in PPO or best-of-n reranking during training, not only at evaluation time), to move from the necessary-condition-level signal reported here toward a sufficient demonstration of alignment quality.
- Extension to a Third Domain: Applying the same pairwise structure to a third heterogeneous domain such as image-generation preference learning or recommendation systems would further verify the generality of the unified framework.
- Theoretical Analysis: This study is based on empirical observations. Identifying the theoretical conditions under which a pairwise structure can prevent policy divergence without a reference model would enable a more principled design of reference-free training.
- Domain-specific τ Conditioning: This study used the same $\tau = 1.0$ in both domains. Conditioning τ on domain- or task-specific characteristics may yield further performance gains.
- Future work should plan and report the same number of seeds across domains from the outset, rather than extending seed count post hoc as was done here for the LLM side after observing higher-than-expected seed-to-seed variance.

6. Conclusions

This study reinterpreted pairwise classification not as a technique confined to either offline RL or LLM alignment, but as a domain-agnostic structural principle. Empirical results showed domain-dependent support for this structure across both heterogeneous domains. The proposed unified framework applies a single pairwise objective: given a context in which two candidates differ in relative quality, determine which is superior. This formulation applies in the same form to both offline RL (pairwise-DT) and LLM alignment (pairwise-FT). The core of this design is that, by relying solely on relative order rather than absolute values, it can unify the training signals of two domains with different scales and semantic systems (Q-value and preference score) under a single loss structure while still training without a reference model.

This study confirmed the following through a three-stage validation: (1) in LLM alignment, Pairwise-FT achieved higher reward accuracy than DPO on all five seeds, while requiring approximately half of DPO's training time and operating without a reference model. This LLM-side conclusion is further supported by a paired t-test and a bootstrap confidence interval excluding zero (Section 4.3), in addition to the sign test; (2) in offline RL, Pairwise-DT outperformed DT on an environment-mean basis on MuJoCo (HalfCheetah, Hopper, Walker2d), reconfirming results established in prior work; (3) a multi-seed

sensitivity analysis showed that the conclusion holds strongly on the LLM side and, to a lesser extent, on the MuJoCo side, rather than depending on a single favorably chosen hyperparameter value. This shows that pairwise classification is not a result confined to favorable hyperparameter settings or isolated seeds, but rather a structural principle that operates consistently across heterogeneous domains.

The contribution of this study lies in demonstrating that the reference-free pairwise classification structure, previously validated only within offline RL through the author's prior work, generalizes as a domain-agnostic principle, even to the fully heterogeneous domain of LLM alignment. At the same time, this study does not overstate the advantages of this structure over binary preference or the MGDT. Both alternatives share the same practical advantage of being reference-free, and the strongest claim this study makes is not that "pairwise is always superior to every alternative", but that "pairwise provides competitive or superior performance compared with reference-dependent alternatives (DPO), without a reference model, in each of the two domains".

Future work will pursue several directions: extending the framework to larger language models, conducting end-to-end validation that connects the learned scorer directly to policy optimization, exploring a third domain, identifying the theoretical conditions under which reference-free training is possible, and developing domain-specific sensitivity analyses for hyperparameter conditioning. We hope that the perspective proposed in this study, "pairwise classification as a cross-domain principle," offers a new trajectory for research on both offline RL and LLM alignments. A limitation of this study is that, on the offline RL side, it adopted a within-batch approximation rather than a fully identical context comparison. This approximation stems from the practical constraint that it is difficult to obtain multiple actions in the same state in an offline dataset, and this trade-off is judged reasonable given the constraints of the offline learning setting.

Funding: This study received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions of this study are included in this article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Additional MuJoCo Results

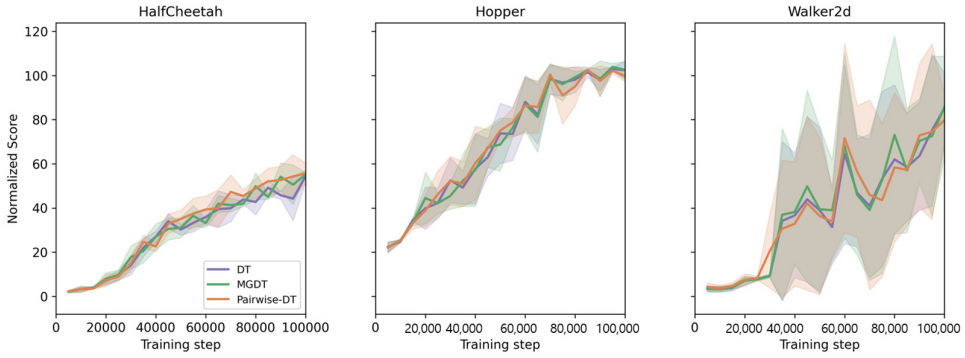


Figure A1. Learning Curves by Environment (mean ± std band, three seeds, 100K steps).

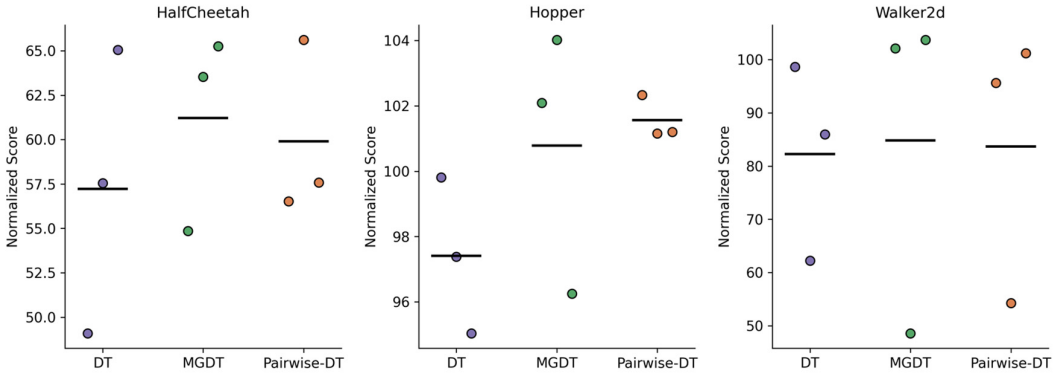


Figure A2. Three-Seed Scatter by Environment (horizontal line = mean).

Appendix B. Additional LLM Alignment Results

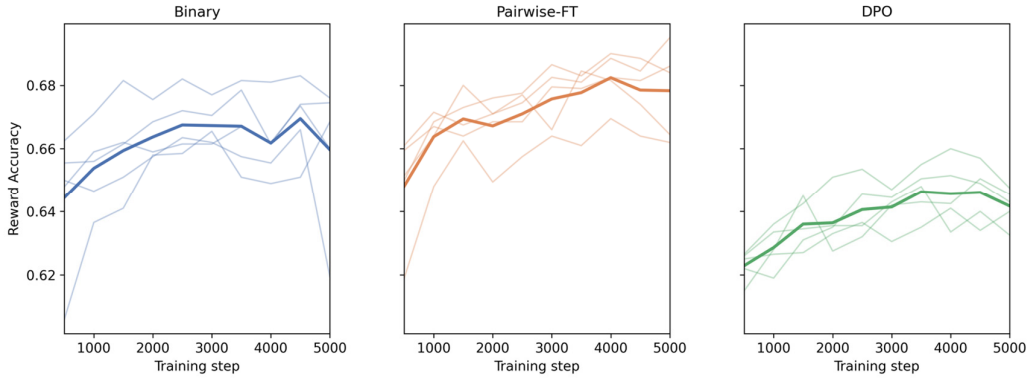


Figure A3. Five-Seed Learning Curves by Method (thin lines = individual seeds, thick line = mean).

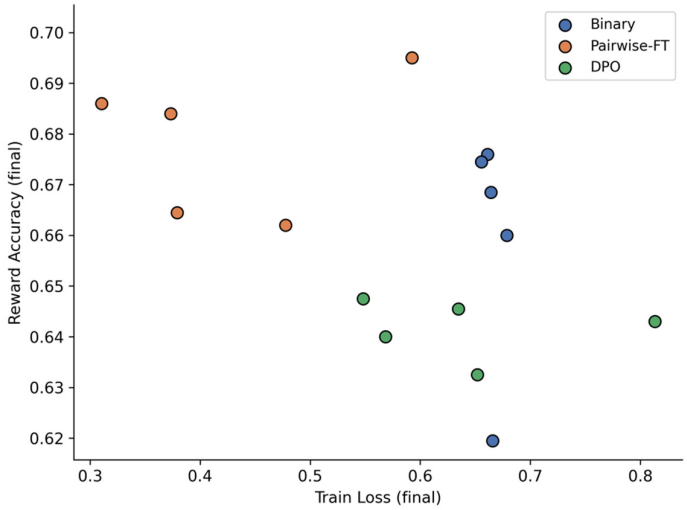


Figure A4. Loss–Accuracy Decoupling Scatter Plot (five seeds × three methods, n = 15).

Table A1. Per-seed paired difference (Pairwise-FT−DPO reward accuracy).

Seed	Pairwise-FT	DPO	Δ
0	0.6860	0.6400	+0.0460
1	0.6620	0.6455	+0.0165
2	0.6840	0.6475	+0.0365
3	0.6950	0.6325	+0.0625
4	0.6645	0.6430	+0.0215

This Table A1 provides the per-seed raw data underlying the paired *t*-test, confidence interval, and bootstrap analysis reported in Section 4.3.

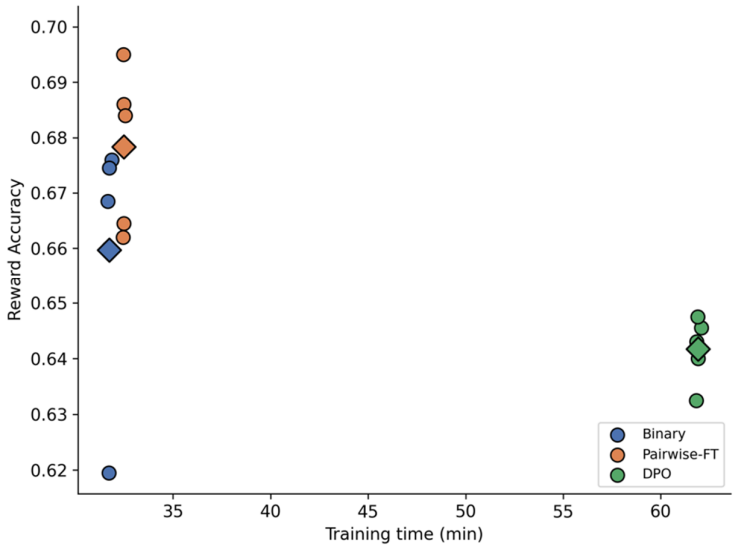


Figure A5. Accuracy vs. Training Time (Pareto, ◇ = method mean).

Appendix C. Temperature Sensitivity Analysis

Table A2. (MuJoCo, HalfCheetah, Pairwise-DT).

τ	Normalized Score (Mean ± Population std, n Seeds)
0.5	56.28 ± 3.42 (n = 3, seeds 0/1/2)
1.0 (baseline)	59.91 ± 4.06 (n = 3, seeds 0/1/2)
2.0	51.10 ± 1.66 (n = 3, seeds 0/1/2)

A multi-seed τ sensitivity analysis on HalfCheetah (three seeds per τ) confirms that $\tau = 1.0$ achieves the highest mean among the tested values, consistent with the single-seed estimate previously obtained. However, the range across τ (8.81) is about 2.17× the baseline's three-seed standard deviation (4.06), larger than the earlier single-seed estimate suggested (about 1.39×), so this study no longer describes MuJoCo results as robust to τ regardless of choice; $\tau = 1.0$ is directionally favored, but the margin should not be overstated. At $\tau = 2.0$, Pairwise-DT (51.10 ± 1.66) falls clearly below MGDT (61.21 ± 4.55, Table 2) on a multi-seed basis, replacing the earlier single-seed observation (seed = 0, 54.85) with more robust evidence of the same conclusion.

Table A3. (LLM, Pairwise-FT).

τ	Reward Accuracy (Mean ± Population std, n Seeds)
0.5	0.6728 ± 0.0038 (n = 3, seeds 0/1/4)
1.0 (baseline)	0.6783 ± 0.0129 (n = 5, seeds 0/1/2/3/4)
2.0	0.6652 ± 0.0031 (n = 3, seeds 0/1/4)

A multi-seed τ sensitivity analysis on the LLM side (three seeds per off-baseline τ) confirms that $\tau = 1.0$ achieves the highest mean reward accuracy among the tested values. The range across τ (0.0131) is about 1.02× the baseline's five-seed standard deviation (0.0129), indicating that this domain's results are robust to the choice of τ within the tested range, consistent with the single-seed estimate previously obtained.

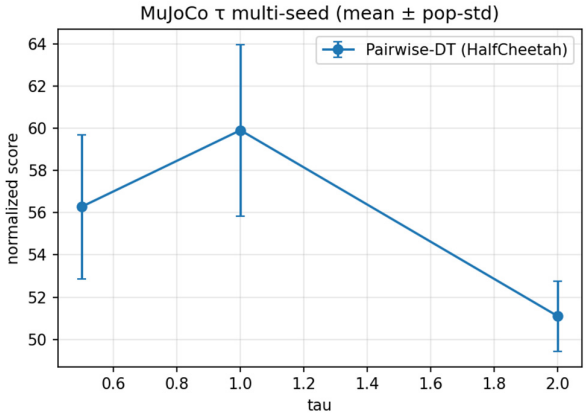


Figure A6. Multi-seed τ sensitivity on MuJoCo (HalfCheetah, Pairwise-DT). Points show mean normalized score across seeds per τ (n = 3); error bars show population standard deviation.

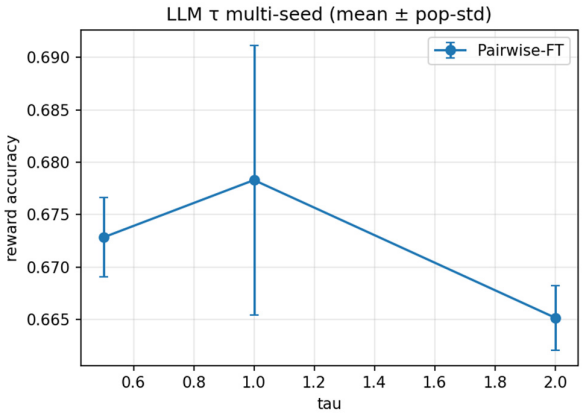


Figure A7. Multi-seed τ sensitivity on LLM (Pairwise-FT). Points show mean reward accuracy across seeds per τ (n = 3 for τ = 0.5/2.0, n = 5 for τ = 1.0 baseline); error bars show population standard deviation.

As shown in Figure A6, $\tau = 1.0$ achieves the highest mean on MuJoCo but the range across τ is comparatively large; as shown in Figure A7, the LLM side shows a similarly peaked trend at τ = 1.0 with a much smaller range relative to its baseline variance.

Appendix D. Win-Rate/LLM-Judge Comparison: Detailed Results

Table A4 reports the full win-rate results (Section 4.3) from the lightweight generation-quality check using Claude (claude-sonnet-5) as an LLM judge, for readers consulting this appendix independently of the main text. Per-prompt verdicts are archived separately (judge_verdicts.csv).

Table A4. Win-rate comparison (100 prompts, HH-RLHF test set).

Comparison	Win (n, %)	Tie	DPO Win
Pairwise-FT vs. DPO	97 (97%)	2	1
Binary vs. DPO	96 (96%)	3	1

As emphasized in Section 4.3, this comparison is confounded by a methodological asymmetry (best-of-four selection vs. a single DPO sample) and the judge model's temperature parameter was unavailable at the time of this study; these win-rates should be read as a preliminary, heavily caveated signal rather than as evidence of superior generation quality.

References

1. Christiano, P.F.; Leike, J.; Brown, T.; Martic, M.; Legg, S.; Amodei, D. Deep reinforcement learning from human preferences. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 4299–4307.
2. Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. Training language models to follow instructions with human feedback. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 27730–27744. [CrossRef]
3. Levine, S.; Kumar, A.; Tucker, G.; Fu, J. Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems. *arXiv* **2020**, arXiv:2005.01643.
4. Chen, L.; Lu, K.; Rajeswaran, A.; Lee, K.; Grover, A.; Laskin, M.; Abbeel, P.; Srinivas, A.; Mordatch, I. Decision Transformer: Reinforcement Learning via Sequence Modeling. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 15084–15097.
5. Lee, K.-H.; Nachum, O.; Yang, M.; Lee, L.; Freeman, D.; Guadarrama, S.; Fischer, I.; Xu, W.; Jang, E.; Michalewski, H.; et al. Multi-Game Decision Transformers. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 27921–27936. [CrossRef]
6. Bradley, R.A.; Terry, M.E. Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika* **1952**, *39*, 324–345. [CrossRef]
7. Rafailov, R.; Sharma, A.; Mitchell, E.; Ermon, S.; Manning, C.D.; Finn, C. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 53728–53741. [CrossRef]
8. Kim, C. Classification-Based Q-Value Estimation for Continuous Actor-Critic Reinforcement Learning. *Symmetry* **2025**, *17*, 638. [CrossRef]
9. Kim, C. Efficient Classification-Based Constraints for Offline Reinforcement Learning. *Appl. Sci.* **2025**, *15*, 12197. [CrossRef]
10. Kim, C. Sampling as a Structural Constraint for Stable Multitask Offline Reinforcement Learning. *Appl. Sci.* **2026**, *16*, 3511. [CrossRef]
11. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language Models are Unsupervised Multitask Learners. *OpenAI Blog* **2019**, *1*, 9.
12. Todorov, E.; Erez, T.; Tassa, Y. MuJoCo: A Physics Engine for Model-based Control. In *Proceeding of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, Portugal, 7–12 October 2012*; IEEE: New York, NY, USA, 2012; pp. 5026–5033.
13. Kumar, A.; Zhou, A.; Tucker, G.; Levine, S. Conservative Q-Learning for Offline Reinforcement Learning. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1179–1191.
14. Kostrikov, I.; Nair, A.; Levine, S. Offline Reinforcement Learning with Implicit Q-Learning. *arXiv* **2021**, arXiv:2110.06169.
15. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*, 2nd ed.; MIT Press: Cambridge, MA, USA, 2018.
16. Burges, C.J.C.; Shaked, T.; Renshaw, E.; Lazier, A.; Deeds, M.; Hamilton, N.; Hullender, G. Learning to rank using gradient descent. In *Proceedings of the 22nd International Conference on Machine Learning*; ACM Digital Library: New York, NY, USA, 2005; pp. 89–96.
17. Bai, Y.; Jones, A.; Ndousse, K.; Askell, A.; Chen, A.; DasSarma, N.; Drain, D.; Fort, S.; Ganguli, D.; Henighan, T.; et al. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback. *arXiv* **2022**, arXiv:2204.05862.
18. Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. ICLR 2019. *arXiv* **2017**, arXiv:1711.05101.
19. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 8026–8037.
20. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020; pp. 38–45.
21. Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; Zaremba, W. OpenAI Gym. *arXiv* **2016**, arXiv:1606.01540.
22. Farama Foundation. Gymnasium: A Standard API for Reinforcement Learning. Available online: https://gymnasium.farama.org/ (accessed on 21 July 2026).
23. Farama Foundation. Minari: A Standard Format for Offline Reinforcement Learning Datasets. Available online: https://minari.farama.org/ (accessed on 21 July 2026).
24. Fu, J.; Kumar, A.; Nachum, O.; Tucker, G.; Levine, S. D4RL: Datasets for Deep Data-Driven Reinforcement Learning. *arXiv* **2020**, arXiv:2004.07219.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.