ALIGNING LLMS WITH DOMAIN INVARIANT REWARD MODELS

David Wu
david9dragon9@gmail.com

Sanjiban Choudhury
Department of Computer Science
Cornell University
Ithaca, NY 14850
sc2582@cornell.edu

ABSTRACT

Aligning large language models (LLMs) to human preferences is challenging in domains where preference data is unavailable. We address the problem of learning reward models for such target domains by leveraging feedback collected from simpler source domains, where human preferences are easier to obtain. Our key insight is that, while domains may differ significantly, human preferences convey *domain-agnostic* concepts that can be effectively captured by a reward model. We propose DIAL, a framework that trains domain-invariant reward models by optimizing a dual loss: a domain loss that minimizes the divergence between source and target distribution, and a source loss that optimizes preferences on the source domain. We show DIAL is a general approach that we evaluate and analyze across 4 distinct settings: (1) Cross-lingual transfer (accuracy: $0.621 \rightarrow 0.661$), (2) Clean-to-noisy (accuracy: $0.671 \rightarrow 0.703$), (3) Few-shot-to-full transfer (accuracy: $0.845 \rightarrow 0.920$), and (4) Simple-to-complex tasks transfer (correlation: $0.508 \rightarrow 0.556$). Our code, models and data are available at https://portal-cornell.github.io/dial/.

1 INTRODUCTION

Reinforcement Learning from Human Feedback (RLHF) has emerged as a popular paradigm for aligning language models (Ouyang et al., 2022; Dubey et al., 2024). This approach involves training and optimizing a reward model that learns human preferences. However, the effectiveness of RLHF is limited by the ability to collect high-quality feedback. As tasks become more complex, they require greater human expertise and time, making it harder for humans to supervise and provide feedback (Leike et al., 2018).

We address the problem of learning reward models for target domains that lack human preference feedback. While feedback is unavailable in the target domain, it is often easy to collect on related source domains. For example, extensive preference data is available in English (source domain) across various tasks, whereas low-resource languages (target domain) may have little to no labeled data (Costa-jussà et al., 2022). Similarly, preferences are easier to collect for simpler tasks, like rating article summaries Franklin et al. (2022), compared to more complex tasks, such as evaluating full-length articles (Crossley et al., 2024).

Prior works address the problem of no target domain data through methods such as regularizing with a text-generation loss (Yang et al., 2024; Zhang et al., 2024), pre-training on unlabeled target data (Karouzos et al., 2021), or few-shot prompting (Winata et al., 2022a). However, both regularization and pre-training are surrogate objectives that don't guarantee the reward model learns the correct preferences on the target domain. Finally, few-shot learning is often sensitive to the choice of examples, leading to high variance in performance.

Our key insight is that, **while domains may differ significantly, human preferences convey *domain-agnostic* concepts that can be effectively captured by a reward model.** By designing the reward model to disentangle domain-specific features from concepts, we enable transfer across domains. To achieve this, we train on a dual loss: a domain loss that minimizes divergence between source and target distributions, and a source loss that learns preferences on the source domain.

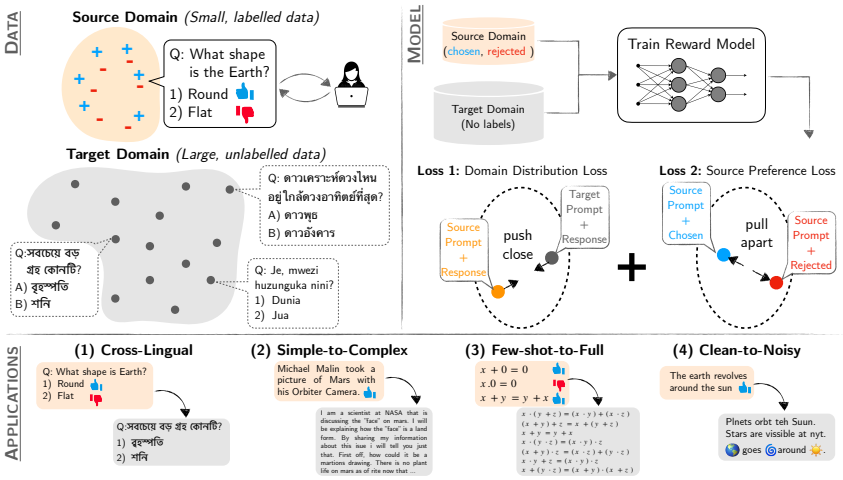

Figure 1: DIAL trains domain-invariant reward model for target domains with no labeled preference data. DIAL leverages labeled source data and unlabeled target data to train reward models on a dual loss: a domain loss that minimizes the divergence between source and target distribution, and a source loss that optimizes preferences on the source domain. We show DIAL is a general approach that we evaluate and analyze across 4 distinct applications: (1) Cross-lingual transfer, (2) Clean-to-noisy, (3) Few-shot-to-full transfer, and (4) Simple-to-complex tasks transfer.

We propose **D**omain **I**nvariant **A**lignment for **L**anguage (DIAL), a framework for training domain-invariant reward models. DIAL takes labeled source data and unlabeled target data, and trains a base LM with a critic head and a reward head. The critic head is trained adversarially on a domain loss to minimize Wasserstein distance (WD) Arjovsky et al. (2017) between source and target embeddings while the reward head minimizes a source loss that optimizes preferences on source data. Effectively, the domain loss aligns source and target embeddings, while the source loss separates chosen and rejected embeddings, encouraging the reward model to learn domain-invariant preferences.

We demonstrate that DIAL is a general approach that can be used to do domain transfer in multiple different paradigms. Specifically, we evaluate and analyze the following settings: *(a) Cross-lingual Transfer:* transferring preferences from a high-resource language (e.g., English) to a low-resource language (e.g., Korean); *(b) Clean-to-Noisy Transfer:* adapting preferences from clean, structured data to noisy, real-world data (e.g., internet text with slang or emojis), *(c) Few-shot-to-full Transfer:* leveraging limited labeled examples to generalize across a broader, unlabeled target distribution, and *(d) Simple-to-complex Transfer:* aligning preferences from simpler tasks (e.g., short texts) to more challenging tasks (e.g., long-form content). Our key contributions are:

1. A novel framework, DIAL, for training domain-invariant reward models. DIAL transfers preferences from labeled source to unlabeled target domains, achieving efficient preference modeling through robust adaptation to similar domains.
2. Theoretical and empirical analysis of DIAL reward models on target distributions (e.g. scaling).
3. Evaluation across four distinct applications:
   (a) *Cross-lingual*: from English to 3 languages on Stanford Human Preference dataset (Ethayarajh et al., 2022) (accuracy: $0.621 \rightarrow 0.661$).
   (b) *Clean-to-noisy*: from grammatically correct to noisy posts on Stanford Human Preference dataset (Ethayarajh et al., 2022) (accuracy: $0.671 \rightarrow 0.703$).
   (c) *Few-shot-to-full*: from 10 examples on CValues (Xu et al., 2023) (accuracy: $0.845 \rightarrow 0.920$).
   (d) *Simple-to-complex*: from scoring short argument fragments to long student essays on Kaggle (Crossley et al., 2024) (correlation: $0.508 \rightarrow 0.556$).

2 Approach

We present Domain Invariant Alignment for Language (DIAL), a framework for aligning large language models across domains where human preference feedback data is unavailable. Given a labeled preference dataset on a source domain $\mathcal{D}_{\text{src}} = \{(x, y^+, y^-)\}$ and an unlabeled dataset on a target

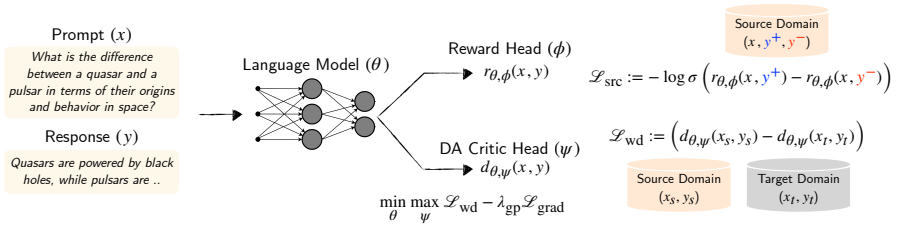

Figure 2: **DIAL overview.** DIAL takes labeled source and unlabeled target data and trains a domain-invariant reward model. The model takes prompt (x) and response (y), passes it through a base language model (θ) with two heads: a domain critic head (ψ) and a reward head (ϕ). The critic head is trained adversarially to minimize the Wasserstein distance between source and target embeddings while the reward head optimizes preferences on source data.

domain $\mathcal{D}_{\text{tgt}} = \{x, y\}$, DIAL trains a domain-invariant reward model $r_\theta(x, y)$ to achieve strong performance on both source and target domains. To achieve this, we train on a dual loss: a domain loss that minimizes the Wasserstein Distance between source and target distribution, and a source loss that optimizes preferences on the source domain.

2.1 Domain-invariant Reward Model

We introduce a domain-invariant reward model that enables transferring preferences from a labeled source domain to an unlabeled target domain. The model takes a base LLM, removes the final unembedding layer, and adds two scalar output heads: a domain critic head and a reward head. Fig. 2 provides an overview of our architecture.

Domain Critic Head. The domain critic head, denoted as $d_{\theta,\psi}(x, y)$, maps a prompt x and response y to a scalar score, which is used to measure the distributional distance between source and target distributions. It takes the embedding of (x, y) from the language model θ, and passes it through an MLP head ψ to compute a scalar score.

We choose the Wasserstein distance (Arjovsky et al., 2017), a metric that measures the minimum cost of transporting one probability distribution to match another. Unlike KL divergence, which requires overlapping supports and can be undefined when distributions do not overlap, the Wasserstein distance provides a meaningful comparison even when the distributions have disjoint supports, making it well-suited for domain adaptation. The 1-Wasserstein distance can be expressed in its dual formulation (Villani et al., 2009) as:

$$W_1(\mathbb{P}, \mathbb{Q}) = \sup_{||f||_L \leq 1} \mathbb{E}_{z\sim\mathbb{P}}[f(z)] - \mathbb{E}_{z\sim\mathbb{Q}}[f(z)] \tag{1}$$

where $|f||_L \leq 1$ is the set of 1-Lipschitz functions.

We train the critic ψ to approximate the Wasserstein distance between source and target distributions by maximizing the expected score difference between source and target:

$$\max_\psi \mathcal{L}_{\text{wd}}(\theta, \psi) = \mathbb{E}_{(x_s,y_s)\sim\mathcal{D}_{\text{src}}}\left[d_{\theta,\psi}(x_s, y_s)\right] - \mathbb{E}_{(x_t,y_t)\sim\mathcal{D}_{\text{tgt}}}\left[d_{\theta,\psi}(x_t, y_t)\right] \tag{2}$$

where $d_{\theta,\psi}(x, y)$ models the feature functions f. To ensure that $d_{\theta,\psi}(x, y)$ satisfies the Lipschitz constraint, we impose a gradient penalty (Gulrajani et al., 2017) on the critic:

$$\mathcal{L}_{\text{grad}}(\psi) = \mathbb{E}_{(x,y)}\left[\left(\|\nabla_{x,y} d_{\theta,\psi}(x, y)\| - 1\right)^2\right] \tag{3}$$

where critic gradients are penalized not only at source and target embeddings, but at random interpolates between the two. The critic maximizes a weighted difference of $\mathcal{L}_{\text{wd}} - \lambda_{gp}\mathcal{L}_{\text{grad}}$.

We then update the language model embeddings θ to minimize the Wasserstein distance, i.e., $\min_\theta \mathcal{L}_{\text{wd}}(\theta, \psi)$ while keeping the critic frozen. This results in the following adversarial game between the critic and the language model:

$$\min_\theta \max_\psi \mathcal{L}_{\text{wd}}(\theta, \psi) - \lambda_{gp}\mathcal{L}_{\text{grad}}(\psi) \tag{4}$$

The equilibrium of the game is reached if the language model θ finds embeddings where the source and target data are indistinguishable, thus being domain-invariant.

Reward Head. The reward head, denoted as $r_{\theta,\phi}(x, y)$, maps a prompt x and response y to a scalar reward. It takes the embedding of (x, y) from the language model θ, and passes it through a linear head ϕ to compute a scalar reward Ouyang et al. (2022). Given a labeled source preference dataset $\mathcal{D}_{\text{source}} = (x, y^+, y^-)$, we train both the reward head and the embedding using a Bradley-Terry model (Bradley & Terry, 1952) on the following source loss:

$$\mathcal{L}_{\text{src}}(\theta, \phi) = \mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}_{\text{src}}}\left[\log \sigma\big(r_{\theta,\phi}(x, y^+) - r_{\theta,\phi}(x, y^-)\big)\right] \tag{5}$$

where σ is the sigmoid function. This loss encourages the reward head to maximize the difference in rewards between preferred and rejected responses.

2.2 DIAL ALGORITHM

Algorithm 1 describes the DIAL algorithm. At every iteration, the critic head is updated to maximize the Wasserstein distance between source and target embeddings while enforcing a Lipschitz constraint using a gradient penalty. Next, the language model is updated to minimize the Wasserstein distance, aligning the source and target embeddings. Finally, the reward head minimizes a preference loss on the source data to separate chosen and rejected responses. By alternating between these updates, DIAL learns a reward model that transfers preferences from the source to the target domain.

Algorithm 1 DIAL: Learning Domain Invariant Rewards

```
# Inputs: mixed_dataloader (Yields source and target
    data), lm (Language model producing embeddings),
    critic (MLP computing Wasserstein Distance), reward
    (Linear reward head)
for mixed_batch in mixed_dataloader:
  # Load labeled source, unlabeled target
  src_chosen, src_reject, tgt_all = mixed_batch
  src_all = cat([src_chosen, src_reject])

  # Critic maximizes source-target dist
  lm.requires_grad_(False)
  src_emb, tgt_emb = lm(src_all), lm(tgt_all)
  wd = (critic(src_emb) - critic(tgt_emb)).mean()
  gp_loss = grad_penalty(critic, src_emb, tgt_emb)
  critic_loss = -wd + gp_loss

  # Embeddings minimize source-target dist
  lm.requires_grad_(True); critic.requires_grad_(False)
  da_loss = wd

  # Rewards minimize source preference loss
  ch_emb, rj_emb = lm(src_chosen), lm(src_reject)
  ch_rew, rj_rew = reward(ch_emb), reward(rj_emb)
  src_loss = -F.logsigmoid(ch_rew - rj_rew).mean()

  total_loss = src_loss + da_loss + critic_loss
```

2.3 THEORETICAL ANALYSIS

We now analyze the generalization of the DIAL reward model $r(x, y)$. Let $f(x, y)$ denote the ground-truth function which assigns rewards to prompt-response pairs. To measure the alignment between r and f, we consider pairwise preferences derived from triplets (x, y, y'), where y, y' are two responses to prompt x. The preference induced by f is $f(y_{\text{win}}) \geq f(y_{\text{loss}})$. The error of reward model r in a domain $\mathcal{D}$ is the expected disagreement between r and f on a distribution $\mathcal{D}$ defined by the Bradley-Terry loss:

$$\epsilon_{\mathcal{D}}(r, f) = \mathbb{E}_{(x,y,y')\sim\mathcal{D}}\left[\sigma\big(r(x, y_{\text{loss}}) - r(x, y_{\text{win}})\big)\right] \tag{6}$$

where $\sigma(z) = \frac{1}{1+e^{-z}}$ is the sigmoid function. This error measures the probability that r disagrees with f, with $\epsilon_{\mathcal{D}}(r, f) \to 0$ as r aligns perfectly with f.

We show that performance of DIAL on the target domain is bounded by sum of performance on the source domain and the Wasserstein distance between source and target:

Theorem 2.1. *Let r be a K-Lipschitz function. Then the target domain error $\epsilon_T(r, f)$ satisfies:*

$$\epsilon_T(r, f) \leq \epsilon_S(r, f) + 2KL_\sigma W_1(\mu_S, \mu_T), \tag{7}$$

where $W_1(\mu_S, \mu_T)$ is the Wasserstein-1 distance between the source and target distributions μ_S and μ_T over (x, y), and $L_\sigma = \frac{1}{4}$ is the Lipschitz constant of σ.

See Appendix B for the proof. The two terms on the right hand side are the source and domain loss in DIAL. By minimizing the sum, DIAL bounds the target performance.

3 EXPERIMENTS

Baselines. We compare against various baselines. Src-Pref is a reward model trained on preference data on the source domain. Src-Pref-SFT Yang et al. (2024) trains a reward model on source

	legaladvice			askscience			explainlikeimfive		
Method	**Korean**	**Thai**	**Chinese**	**Korean**	**Thai**	**Chinese**	**Korean**	**Thai**	**Chinese**
Base LM	0.58	0.57	0.60	0.57	0.58	0.55	0.55	0.54	0.51
Src-Pref	0.60	0.63	0.61	0.57	0.62	0.59	0.65	0.63	**0.68**
Src-Pref-SFT [1]	0.62	0.59	0.64	0.56	0.61	0.56	0.65	0.64	0.61
Src-Pref-Tgt-NTP [2]	0.64	0.56	0.65	0.57	0.63	0.61	0.64	0.64	0.61
DIAL (ours)	**0.68**	**0.66**	**0.68**	**0.63**	**0.68**	**0.62**	**0.68**	**0.65**	0.67
Tgt-Pref*	0.69	0.66	0.67	0.62	0.67	0.62	0.67	0.65	0.68
Src-Tgt-Pref*	0.69	0.72	0.70	0.63	0.67	0.64	0.70	0.67	0.69

Table 1: **Cross-lingual Transfer.** Accuracy results of reward models trained on source data (English) and evaluated on target data (Korean/Thai/Chinese) on three splits of Stanford Human Preference Dataset (Ethayarajh et al., 2022), each 1K. Results are averaged over 3 seeds. DIAL outperforms all baselines, including [1] Yang et al. (2024) and [2] Karouzos et al. (2021)

preference data, and additionally regularizes the base model with SFT loss on chosen responses on the source data. Src-Pref-Tgt-NTP Karouzos et al. (2021) trains a reward model on source preference data, and also regularizes the base model with a pre-training task on both prompt and response on target data.[1] Base LM is a generative baseline that prompts the base LLM to choose from multiple responses, using chain-of-thought. We also test two oracles: Tgt-Pref* trains a reward model on target preference data, Src-Tgt-Pref* trains a reward model on both source and target data.

Model and Metrics We use Gemma-2b (GemmaTeam, 2024a) base with a linear reward head and a 2-layer MLP critic head, using LoRA (Hu et al., 2021) (details in Appendix A.1). We measure accuracy of rewards on preferences with chosen and rejected responses (variances in Appendix A).

3.1 Application 1: Cross-lingual Transfer

Setup. We first look at cross-lingual transfer where preferences exist in a high-resource source domain but must be transferred to a low-resource target domain. Stanford Human Preferences (Ethayarajh et al., 2022) (SHP) is a dataset consisting of questions from Reddit and preference pairs of answers. We select 3 diverse subreddits with train/test splits: legaladvice (20K/1K), explainlikeimfive (20K/1K) and askscience (13K/1K). To evaluate transfer, we translate data with NLLB (NllbTeam, 2022) to 3 languages: Korean, Thai, Chinese. See Appendix A.2.

Results. Table. 1 shows that DIAL outperforms baselines on all subreddits and languages. The performance improvements on Korean ($0.57 \rightarrow 0.63$) and Thai ($0.62 \rightarrow 0.68$) are stronger than Chinese ($0.59 \rightarrow 0.62$), likely due to Chinese being more common in training datasets of the base LM. Performance improvements on subreddits legaladvice ($0.63 \rightarrow 0.66$) and askscience ($0.62 \rightarrow 0.68$) are stronger than explainlikeimfive ($0.64 \rightarrow 0.65$), likely because legal advice and ask science may rely on more advanced terminology that requires more alignment. We note the single case of explainlikeimfive-Chinese where DIAL does not improve over baselines. This is because training on source already matches oracle performance, leaving little to no room for improvement in transfer ability using domain adaptation. Finally, we note that on many subreddits/language DIAL reaches oracular performance of Src-Tgt-Pref*.

3.2 Application 2: Clean-to-noisy Transfer

Setup. We next look at the application where the source domain is clean synthetic data, but the target domain is noisy, real-world internet data. To emulate this, we selected the same 3 splits from SHP (Ethayarajh et al., 2022): legaladvice, explainlikeimfive, and askscience. We used the original test data (1K each) which is already noisy. We create a "clean" dataset (20K/20K/13K) by rewriting the data formally using Gemma-2-9b-it (GemmaTeam, 2024b). The real-world data contains significant noise in the form of spelling, grammar, and language errors. See Appendix A.3.

[1]The paper uses a masked language modeling task. To make it comparable for our decoder-only LMs, we used next token prediction instead of masked language modeling.

Method	Legal	Science	ELI5
Base LM	0.55	0.56	0.55
Src-Pref	0.71	0.63	0.67
Src-Pref-SFT [1]	0.71	0.61	0.64
Src-Pref-Tgt-NTP [2]	0.70	0.61	0.63
DIAL (ours)	**0.76**	**0.65**	**0.70**
Tgt-Pref*	0.77	0.67	0.74
Src-Tgt-Pref*	0.78	0.69	0.73

Table 2: **Clean-to-noisy Transfer.** Accuracy results of reward models trained on clean data and evaluated on noisy data from three splits of SHP dataset (Ethayarajh et al., 2022), each 1K. Results averaged over 3 seeds. DIAL outperforms all baselines, including [1] Yang et al. (2024) and [2] Karouzos et al. (2021)

Method	SplitA	SplitB	SplitC
Few-shot LM	0.60	0.64	0.65
Src-Pref	0.82	0.86	0.85
Src-Pref-SFT [1]	0.74	0.79	0.71
Src-Pref-Tgt-NTP [2]	0.76	0.72	0.72
DIAL (ours)	**0.85**	**0.97**	**0.94**
Tgt-Pref*	1.00	1.00	1.00

Table 3: **Few-shot-to-full Transfer.** Accuracy results on three splits of CValues safety preference dataset (Xu et al., 2023) of reward models trained on few-shot source examples, evaluated on 7.5K target examples. DIAL outperforms all baselines, including [1] Yang et al. (2024) and [2] Karouzos et al. (2021).

Results. Table 2 shows that DIAL outperforms baselines on all splits, matching the oracle on legaladvice. Gain on ELI5 is least as the "clean" data is also informal given the nature of ELI5.

3.3 Application 3: Few-shot-to-full Transfer

Setup. We next look at the task where the source is a set of few-shot labeled data and the target is unlabeled data from the same distribution, often the case when labeling requires experts. We select the CValues safety dataset (Xu et al., 2023) and sample 3 splits of 10 examples as source. We train on 130K unlabeled target data and evaluate on 7.5K labeled data. Our goal is to test DIAL under low label regimes and quantify gains over Src-pref. See Appendix A.4.

Results. Table 3 shows that DIAL outperforms all baselines on all splits, coming close to oracle (1.0) on 2 out of 3 splits. DIAL accuracy varies across splits due to sensitivity to the choice of the few-shot data, but it still strictly outperforms baselines. Few-shot LLM, which uses the examples in-context, performs poorly likely due to a smaller LLM being unable to learn well in-context. We also note that as the Src-Pref to Tgt-Pref gap increases (greater distance between source and target), DIAL's gain over Src-Pref decreases, as the distributions are harder to align.

3.4 Application 4: Simple-to-complex Transfer

Setup. We finally look at the task where the source data is simpler and easier to label, where target data is complex and expensive to label. Specifically, we use Kaggle Argument (Franklin et al., 2022) as source data (20K train) and Kaggle Essay (long) (Crossley et al., 2024) as target data (15K train / 1K val). The source data consists of fragments of student essays from different parts of an argument (e.g. Thesis, Evidence; around 10 to 100 words). The target data contains full student essays (e.g. on Electoral College; around 200 to 600 words). Each example is an essay and a human annotator score. We transform the score data to preference data to train a reward model, and at test time ask the model to score essays. We use 2 metrics: Pearson's r and Spearman's ρ. r measures linear correlation between the rewards and human scores, while ρ measures if order is retained. The goal is to test DIAL on transfer across significantly different lengths and complexity. See Appendix A.5.

Method	Pearson's r	Spearman's ρ
Base LM	0.408 $\pm$0.003	0.394 $\pm$0.003
Src-Pref	0.516 $\pm$0.011	0.508 $\pm$0.014
Src-Pref-SFT [1]	0.567 $\pm$0.018	**0.562** $\pm$0.023
Src-Pref-Tgt-NTP [2]	0.567 $\pm$0.020	0.558 $\pm$0.025
DIAL (ours)	**0.577** $\pm$0.011	0.556 $\pm$0.011
Tgt-Pref*	0.857 $\pm$0.003	0.855 $\pm$0.003
Src-Tgt-Pref*	0.860 $\pm$0.001	0.857 $\pm$0.001

Table 4: **Simple-to-complex Transfer.** Correlation results of reward model ratings trained on short Kaggle argument (Franklin et al., 2022) and evaluated on long essay (Crossley et al., 2024) (1K data). Results show $\bar{x}$ and s over 3 seeds. DIAL outperforms baselines, including [1] Yang et al. (2024) and [2] Karouzos et al. (2021), on r but underperforms on ρ.

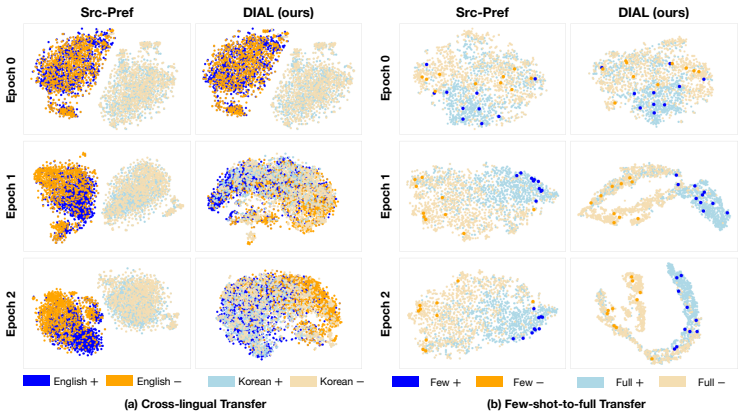

Figure 3: **Reward model embeddings** learned by DIAL and Src-Pref across training iterations on (a) Cross-lingual Transfer and (b) Few-shot-to-full Transfer. Src-Pref separates source embeddings, but not target embeddings, resulting in poor transfer. DIAL learns embeddings that cluster (source positive, target positive) and (source negative, target negative) for better reward transfer.

Results. Table 4 shows that DIAL improves upon Src-Pref on all metrics. While DIAL improves upon all baselines on Pearson's r, Src-Pref-SFT achieves the highest performance on Spearman's ρ. We note that the oracles perform much stronger compared to other tasks, likely due to target being more different from source, such that alignment is challenging. Nonetheless, target data does help DIAL reach better performance with lower variance. While these results show promise that DIAL can provide scalable oversight, there is room for future work to improve DIAL to match the oracle.

3.5 What reward model embeddings does DIAL learn?

To understand why DIAL rewards generalize, we visualize the reward model embeddings for different applications. We compute a t-SNE mapping of the embeddings and apply it to 1000 random target datapoints. Fig. 3 shows embeddings for both DIAL and baseline Src-Pref across training epochs.

In cross-lingual transfer, we analyze the legaladvice-Korean split in Fig. 3(a), where source is English and target is Korean. At epoch 0, source and target data (s and t) are well separated, but positive and negative responses ($+$ and $-$) are not. In Src-Pref, the source preference loss continues separating s_+ and s_- over time. However, t_+ and t_- still remain mixed. DIAL, on the other hand, aligns source and target embeddings, and uses this alignment to simultaneously separate (s_+, t_+) from (s_-, t_-). This alignment enables it to easily transfer preferences.

We see a similar behavior in few-shot-to-full transfer shown in Fig. 3(b), where source is few shot examples and target is the full data. Both methods easily separate source data. However, Src-Pref overfits to the source data and finds an incorrect decision boundary that doesn't separate all data. DIAL aligns the few-shot data with the full data (clustering data around the few-shot examples), leading to a clear boundary that transfers to the full data. See Appendix A.7.

3.6 How does DIAL scale with data?

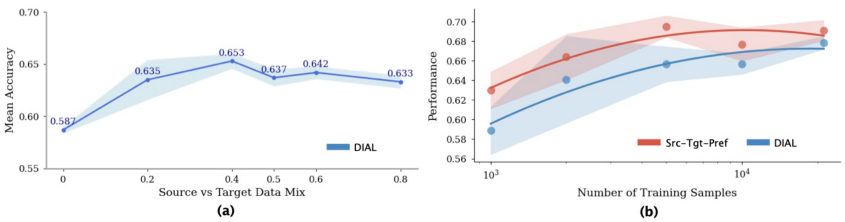

Figure 4: **Scaling Data** on legaladvice-Korean. (a) DIAL accuracy with varying source-target mix (b) DIAL scaling with unlabeled target vs Src-Tgt-Pref with labeled target data (3 seeds).

We analyze how DIAL scales with data on the cross-lingual task of legaladvice-Korean. We hypothesize that for a fixed data budget, there is an optimal mix of labeled source and unlabeled target. From theory, we can bound target performance by source performance (depends on $1/\sqrt{N_{\text{src}}}$) and

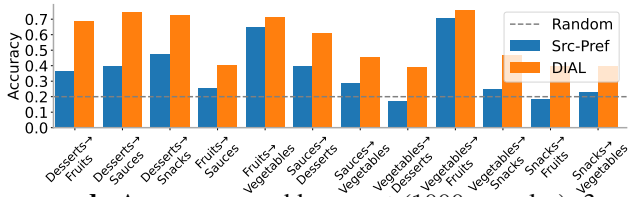

Figure 5: **Spurious reward.** Accuracy on odd-one-out (1000 samples), 3 seeds. `DIAL` learns the correct reward, while `Src-Pref` learns spurious reward of "not source", performing close to random.

estimated Wasserstein distance between source and target (depends on $1/\sqrt{N_{\text{src}}}+1/\sqrt{N_{\text{tgt}}}$). Fig. 4 (a) shows there is indeed a peak around $(0.4, 0.6)$ mix of (target, source). We also study scaling laws for `DIAL` with unlabeled target data vs oracle (`Src-Tgt-Pref*`) with labeled target data. As expected the oracle clearly has an offset from `DIAL`, but with more data, `DIAL` catches up. This is likely because the oracle asymptotes, while `DIAL` uses additional data to perfectly align rewards.

3.7 How robust is DIAL to spurious rewards?

We hypothesize that rewards trained only on source data (`Src-Pref`) are susceptible to learning spurious rewards. This is a well-known problem in reward learning Tien et al. (2022), where causal confounders or biases in train data can cause "reward confusion". We hypothesize that `DIAL` can learn to avoid this confusion with only unlabeled target data.

To introduce spurious correlations, we create a synthetic task of choosing the odd one out. We created sets of 100 items belonging to 5 groups: desserts, fruits, sauces, vegetables, and snacks. We create datasets for each category, where each sample has 4 items in the category and 1 from another. The spurious reward is "not source" (e.g. not fruit), which excels on source but fails on target.

Fig. 5 shows `DIAL` vs `Src-Pref` across tasks. We see that `Src-Pref` performs poorly on target, often the same as random (0.2), likely learning the spurious reward of "not source category". `DIAL` on the other hand generalizes to target, despite seeing no target labels. Generally, we believe that training on multiple target distributions reduces risk of spurious correlations. See Appendix A.6 for details.

3.8 Can DIAL correct distribution shift during RLHF?

We next look at how `DIAL` can help in training better RLHF policies. A common problem in RLHF is distribution shift during training, where the rewards trained on off-policy preferences become inaccurate on the current policy's response distribution Ziegler et al. (2019) over time. This typically requires repeatedly collecting on-policy preferences during training Guo et al. (2024), which is often impractical. We hypothesize that `DIAL` can correct this shift by adapting the reward model trained on off-policy data (source distribution) to current policy responses (target distribution).

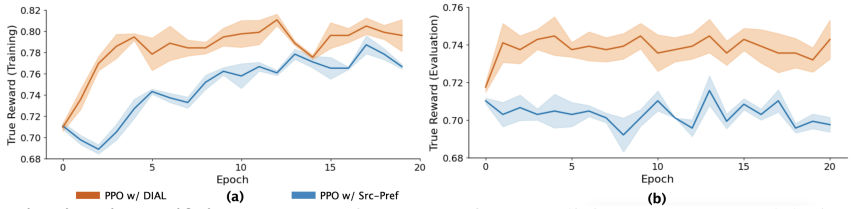

Figure 6: **Distribution shift in RLHF.** Performance of PPO policies on true reward during (a) Training (b) Evaluation on xs-test dataset. PPO with `DIAL`, continually adapts the reward model during PPO training to the current policies response distribution, leading to better supervision, accelerated training, and better generalization on evaluation dataset. Results on 3 seeds.

We test this hypothesis on the xs-test dataset (Röttger et al., 2024), part of the benchmark RewardBench (Lambert et al., 2024), where each prompt is a potentially harmful question. Given a LLM response, we use accuracy of compliance vs refusal as true reward. We choose this task given the objective and low-variance nature of evaluation. The vanilla RLHF procedure (`Src-Pref`) first trains a reward model on off-policy preference data, then trains a policy using PPO to optimize reward over multiple epochs Ouyang et al. (2022). To apply `DIAL`, we begin with the same reward model, and alternate training PPO and `DIAL`, adapting to generated responses from each epoch.

Fig. 6(a) shows PPO true rewards when training with DIAL vs Src-Pref. DIAL accelerates training, by adjusting the reward model to account for the distribution shift, providing better supervision to the policy on current generations. Fig. 6(b) shows that DIAL has better performance than Src-Pref on a held-out evaluation set. This indicates that DIAL may be learning rewards that are better aligned with the target task, leading to better policies. See Appendix A.8 for more details.

4 Related Work

Generalization of RLHF. Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022) is widely used to align LLMs to human preferences, with more recent works focusing on transfer. Some works examine task transfer, where models trained on simpler tasks transfer to harder ones. For example, Hase et al. (2024) showed that LLMs trained on easy STEM questions transfer zero-shot to harder ones, and Sun et al. (2024) found that reward models outperform supervised fine-tuning (SFT) for math problems. Kirk et al. (2023) observed that RLHF boosts transfer at the cost of diversity. Prompting techniques, such as least-to-most (Zhou et al., 2022) and scratchpad prompting (Anil et al., 2022) enhance transfer by breaking down complex tasks.

RLHF has also demonstrated strong cross-lingual transfer. For instance, Wu et al. (2024) and Li et al. (2024) showed that training reward models on one language has strong zero-shot transfer for others. Winata et al. (2022b) found that multilingual pretraining improves transfer across Indonesian languages, while Huang et al. (2023) showed that "cross-lingual" CoT—reasoning in English and translating to the target language—boosts accuracy on multilingual tasks. Tanwar et al. (2023) proposed using embeddings to retrieve few-shot examples and append translations to connect labels across languages. Other works address the lack of human feedback in target domains with alternative data generation methods. For example, Kim et al. (2023) generated synthetic preferences using smaller LLMs, Shaikh et al. (2024) applied inverse reinforcement learning on demonstrations, and Kim et al. (2024) leveraged LLM reasoning to create preferences automatically.

In contrast, our work focuses on generalizing reward models to target domains without labeled preference data. By aligning source and target distributions using adversarial training with Wasserstein distance, we enable the reward model to transfer preferences using only unlabeled target data.

Domain Adaptation. Domain adaptation focuses on transferring supervision from a source task with abundant labeled data to a target task with no labels, with applications like self-driving (Li et al., 2023) and sim2real (Truong et al., 2020). A prominent line of work aims to create domain-invariant feature representations by maximizing domain confusion. Maximum Mean Discrepancy (MMD) (Tzeng et al., 2014) and Domain Adversarial Neural Networks (DANN) (Ganin et al., 2015) achieve this by aligning source and target distributions; DANN uses a gradient reversal layer to match embeddings. Other work focuses on translation, such as CycleGAN (Zhu et al., 2017), which maps source data to target and vice versa for unpaired image translation. Extensions to these methods include DeepJDot (Damodaran et al., 2018), which aligns joint distributions of features and labels, and Wasserstein Distance Guided Representation Learning (WDGRL) (Shen et al., 2018), which stabilizes training with Wasserstein GANs (Arjovsky et al., 2017).

Unlike prior work focused on classification or regression, we address domain adaptation for LLM reward models, aligning source and target while optimizing a human preference alignment loss.

5 Conclusion

We propose DIAL, a framework for training domain-invariant reward models that align human preferences across domains with scarce or no labeled target data. By combining a domain loss to align source and target embeddings with a preference loss to separate chosen and rejected responses, DIAL learns domain-agnostic preferences. We show its effectiveness across 4 diverse tasks, including cross-lingual, clean-to-noisy, few-shot-to-full, and simple-to-complex, with significant gains in target performance. Future work includes extending DIAL to handle drastic source-target shifts, such as adapting between highly divergent tasks, new applications (e.g. transferring to LLM generated data), and understanding the limits of adaptation for reward models.

ACKNOWLEDGMENTS

This project is supported by an OpenAI Superalignment grant and a Google Faculty Research Award.

REFERENCES

Cem Anil, Yuhuai Wu, Anders Johan Andreassen, Aitor Lewkowycz, Vedant Misra, Vinay Venkatesh Ramasesh, Ambrose Slone, Guy Gur-Ari, Ethan Dyer, and Behnam Neyshabur. Exploring length generalization in large language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=zSkYVeX7bC4.

Anthropic. Claude. URL https://claude.ai/.

Martín Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *ArXiv*, abs/1701.07875, 2017. URL https://api.semanticscholar.org/CorpusID:13943041.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39:324, 1952. URL https://api.semanticscholar.org/CorpusID:125209808.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating llms by human preference, 2024. URL https://arxiv.org/abs/2403.04132.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*, 2022. URL https://arxiv.org/abs/2207.04672.

Scott Crossley, Perpetual Baffour, Jules King, Lauryn Burleigh, Walter Reade, and Maggie Demkin. Learning agency lab - automated essay scoring 2.0. kaggle., 2024. URL https://kaggle.com/competitions/learning-agency-lab-automated-essay-scoring-2.

Bharath Bhushan Damodaran, Benjamin Kellenberger, Rémi Flamary, Devis Tuia, and Nicolas Courty. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In *European Conference on Computer Vision*, 2018. URL https://api.semanticscholar.org/CorpusID:4331539.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. Understanding dataset difficulty with $\mathcal{V}$-usable information. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 5988–6008. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/ethayarajh22a.html.

Alex Franklin, Maggie, Meg Benner, Natalie Rambis, Perpetual Baffour, Ryan Holbrook, Scott Crossley, and ulrichboser. Feedback prize - predicting effective arguments., 2022. URL https://kaggle.com/competitions/feedback-prize-effectiveness.

Yaroslav Ganin, E. Ustinova, Hana Ajakan, Pascal Germain, H. Larochelle, François Laviolette, Mario Marchand, and Victor S. Lempitsky. Domain-adversarial training of neural networks. In *Journal of machine learning research*, 2015. URL https://api.semanticscholar.org/CorpusID:2871880.

GemmaTeam. Gemma: Open models based on gemini research and technology. *ArXiv*, abs/2403.08295, 2024a. URL https://api.semanticscholar.org/CorpusID:268379206.

GemmaTeam. Gemma 2: Improving open language models at a practical size. *ArXiv*, abs/2408.00118, 2024b. URL https://api.semanticscholar.org/CorpusID:270843326.

Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein gans. *CoRR*, abs/1704.00028, 2017. URL http://arxiv.org/abs/1704.00028.

Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, Bilal Piot, et al. Direct language model alignment from online ai feedback. *arXiv preprint arXiv:2402.04792*, 2024.

Peter Hase, Mohit Bansal, Peter Clark, and Sarah Wiegreffe. The unreasonable effectiveness of easy training data for hard tasks. *ArXiv*, abs/2401.06751, 2024. URL https://api.semanticscholar.org/CorpusID:266977266.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv: Learning*, 2016. URL https://api.semanticscholar.org/CorpusID:125617073.

J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685, 2021. URL https://api.semanticscholar.org/CorpusID:235458009.

Haoyang Huang, Tianyi Tang, Dongdong Zhang, Xin Zhao, Ting Song, Yan Xia, and Furu Wei. Not all languages are created equal in LLMs: Improving multilingual capability by cross-lingual-thought prompting. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://openreview.net/forum?id=E4ebDeh03O.

HuggingFace. Huggingface transformers reinforcement learning, 2023. URL https://huggingface.co/docs/trl/en/index.

Constantinos Karouzos, Georgios Paraskevopoulos, and Alexandros Potamianos. Udalm: Unsupervised domain adaptation through language modeling. *arXiv preprint arXiv:2104.07078*, 2021. URL https://aclanthology.org/2021.naacl-main.203/.

Dongyoung Kim, Kimin Lee, Jinwoo Shin, and Jaehyung Kim. Aligning large language models with self-generated preference data. *ArXiv*, abs/2406.04412, 2024. URL https://api.semanticscholar.org/CorpusID:270357971.

Sungdong Kim, Sanghwan Bae, Jamin Shin, Soyoung Kang, Donghyun Kwak, Kang Min Yoo, and Minjoon Seo. Aligning large language models through synthetic feedback. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://openreview.net/forum?id=8gYRHspcxK.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of rlhf on llm generalisation and diversity. *ArXiv*, abs/2310.06452, 2023. URL https://api.semanticscholar.org/CorpusID:263830929.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. Rewardbench: Evaluating reward models for language modeling, 2024. URL https://arxiv.org/abs/2403.13787.

Jan Leike, David Krueger, Tom Everitt, Miljan Martic, Vishal Maini, and Shane Legg. Scalable agent alignment via reward modeling: a research direction. *arXiv preprint arXiv:1811.07871*, 2018. URL https://arxiv.org/abs/1811.07871.

Jinlong Li, Runsheng Xu, Jin Ma, Qin Zou, Jiaqi Ma, and Hongkai Yu. Domain adaptation based object detection for autonomous driving in foggy and rainy weather. *IEEE Transactions on Intelligent Vehicles*, 2023. URL https://api.semanticscholar.org/CorpusID:259983180.

Xiaochen Li, Zheng-Xin Yong, and Stephen H. Bach. Preference tuning for toxicity mitigation generalizes across languages. *ArXiv*, abs/2406.16235, 2024. URL https://api.semanticscholar.org/CorpusID:270703187.

Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. *ArXiv*, abs/1711.05101, 2017. URL https://api.semanticscholar.org/CorpusID:3312944.

NllbTeam. No language left behind: Scaling human-centered machine translation. *ArXiv*, abs/2207.04672, 2022. URL https://api.semanticscholar.org/CorpusID:250425961.

OpenAI. Chatgpt. URL https://chatgpt.com/.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155, 2022. URL https://api.semanticscholar.org/CorpusID:246426909.

Paul Röttger, Hannah Rose Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. Xstest: A test suite for identifying exaggerated safety behaviours in large language models, 2024. URL https://arxiv.org/abs/2308.01263.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Omar Shaikh, Michelle Lam, Joey Hejna, Yijia Shao, Michael S. Bernstein, and Diyi Yang. Show, don't tell: Aligning language models with demonstrated feedback. *ArXiv*, abs/2406.00888, 2024. URL https://api.semanticscholar.org/CorpusID:270222089.

Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation, 2018. URL https://arxiv.org/abs/1707.01217.

Zhiqing Sun, Longhui Yu, Yikang Shen, Weiyang Liu, Yiming Yang, Sean Welleck, and Chuang Gan. Easy-to-hard generalization: Scalable alignment beyond human supervision. *ArXiv*, abs/2403.09472, 2024. URL https://api.semanticscholar.org/CorpusID:268385111.

Eshaan Tanwar, Subhabrata Dutta, Manish Borthakur, and Tanmoy Chakraborty. Multilingual llms are better cross-lingual in-context learners with alignment, 2023. URL https://arxiv.org/abs/2305.05940.

Jeremy Tien, Jerry Zhi-Yang He, Zackory Erickson, Anca D Dragan, and Daniel S Brown. Causal confusion and reward misidentification in preference-based reward learning. *arXiv preprint arXiv:2204.06601*, 2022. URL https://arxiv.org/abs/2204.06601.

Joanne Truong, S. Chernova, and Dhruv Batra. Bi-directional domain adaptation for sim2real transfer of embodied navigation agents. *IEEE Robotics and Automation Letters*, 6:2634–2641, 2020. URL https://api.semanticscholar.org/CorpusID:227162315.

Eric Tzeng, Judy Hoffman, N. Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *ArXiv*, abs/1412.3474, 2014. URL https://api.semanticscholar.org/CorpusID:17169365.

Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2009. URL https://link.springer.com/book/10.1007/978-3-540-71050-9.

Genta Winata, Shijie Wu, Mayank Kulkarni, Thamar Solorio, and Daniel Preoţiuc-Pietro. Cross-lingual few-shot learning on unseen languages. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 777–791, 2022a. URL https://aclanthology.org/2022.aacl-main.59/.

Genta Indra Winata, Shijie Wu, Mayank Kulkarni, Thamar Solorio, and Daniel Preoţiuc-Petro. Cross-lingual few-shot learning on unseen languages. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on natural Language Processing*, 2022b. URL https://aclanthology.org/2022.aacl-main.59.pdf.

Zhaofeng Wu, Ananth Balashankar, Yoon Kim, Jacob Eisenstein, and Ahmad Beirami. Reuse your rewards: Reward model transfer for zero-shot cross-lingual alignment. *ArXiv*, abs/2404.12318, 2024. URL https://api.semanticscholar.org/CorpusID:269214448.

Guohai Xu, Jiayi Liu, Ming Yan, Haotian Xu, Jinghui Si, Zhuoran Zhou, Peng Yi, Xing Gao, Jitao Sang, Rong Zhang, Ji Zhang, Chao Peng, Fei Huang, and Jingren Zhou. Cvalues: Measuring the values of chinese large language models from safety to responsibility, 2023. URL https://arxiv.org/abs/2307.09705.

Rui Yang, Ruomeng Ding, Yong Lin, Huan Zhang, and Tong Zhang. Regularizing hidden states enables learning generalizable reward model for llms. *ArXiv*, abs/2406.10216, 2024. URL https://api.semanticscholar.org/CorpusID:270521260.

Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. Generative verifiers: Reward modeling as next-token prediction. *arXiv preprint arXiv:2408.15240*, 2024. URL https://arxiv.org/abs/2408.15240.

Denny Zhou, Nathanael Scharli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Olivier Bousquet, Quoc Le, and Ed Huai hsin Chi. Least-to-most prompting enables complex reasoning in large language models. *ArXiv*, abs/2205.10625, 2022. URL https://api.semanticscholar.org/CorpusID:248986239.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2242–2251, 2017. URL https://api.semanticscholar.org/CorpusID:206770979.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

A Experimental Details

A.1 General parameters

For all experiments, we used base model Gemma-2b (GemmaTeam, 2024a) with an additional learned linear head and a learned (low rank adaptation) LoRA (Hu et al., 2021) adapter with rank 64, lora α of 64. We used AdamW (Loshchilov & Hutter, 2017) with learning rate $5e-5$ and no weight decay unless otherwise stated.

For WDGRL, we used $\lambda = 0.01$, $\lambda_{gp} = 1.0$, with 3 critic iterations. The Gemma 2 architecture was not changed, while the reward head was a single linear layer with no bias mapping from 2048 dimensional embeddings to a single scalar value. The domain adaptation head used a learning rate of 0.0001 with weight decay of 0.001. The domain adaptation head consisted of two MLP layers of width 256 and 128, with GELU (Hendrycks & Gimpel, 2016) activation and no dropout. We used domain adaptation implementations from https://cpjku.github.io/da/.

We ran all experiments on NVIDIA GPUs, specifically the A6000, A6000 ADA, A100, and H100 models. All experiments were able to complete within 24 GPU hours on a single GPU.

A.2 Cross-Lingual Transfers

For the cross-lingual transfer task, we selected three splits (legaladvice, askscience, explainlikeimfive) from the Stanford Human Preference (SHP) dataset (Ethayarajh et al., 2022). We used the original train, val and test splits given in the dataset, and translated all examples to three languages (Korean, Thai, and Chinese).

We used NLLB-200-3.3B (NllbTeam, 2022) translation with temperature 0.0, top_p of 1.0, min_tokens of 0, max_tokens of 1024, and repetition_penalty of 1.15 to reduce repetition in the translations.

Method	legaladvice			askscience			explainlikeimfive		
	Korean	**Thai**	**Chinese**	**Korean**	**Thai**	**Chinese**	**Korean**	**Thai**	**Chinese**
Base LM	0.58	0.57	0.60	0.57	0.58	0.55	0.55	0.54	0.51
Src-Pref	0.60 ± 0.03	0.63 ± 0.01	0.61 ± 0.02	0.57 ± 0.01	0.62 ± 0.01	0.59 ± 0.01	0.65 ± 0.01	0.63 ± 0.01	$\mathbf{0.68 \pm 0.01}$
Src-Pref-SFT	0.62 ± 0.01	0.59 ± 0.04	0.64 ± 0.00	0.56 ± 0.02	0.61 ± 0.02	0.56 ± 0.01	0.65 ± 0.00	0.64 ± 0.00	0.61 ± 0.00
Src-Pref-Tgt-NTP	0.64 ± 0.01	0.56 ± 0.04	0.65 ± 0.02	0.57 ± 0.01	0.63 ± 0.01	0.61 ± 0.01	0.64 ± 0.01	0.64 ± 0.01	0.61 ± 0.01
DIAL (ours)	$\mathbf{0.68 \pm 0.00}$	$\mathbf{0.66 \pm 0.01}$	$\mathbf{0.68 \pm 0.03}$	$\mathbf{0.63 \pm 0.00}$	$\mathbf{0.68 \pm 0.00}$	$\mathbf{0.62 \pm 0.01}$	$\mathbf{0.68 \pm 0.01}$	$\mathbf{0.65 \pm 0.00}$	0.67 ± 0.01
Tgt-Pref*	0.69 ± 0.01	0.66 ± 0.01	0.67 ± 0.01	0.62 ± 0.01	0.67 ± 0.00	0.62 ± 0.01	0.67 ± 0.00	0.65 ± 0.01	0.68 ± 0.01
Src-Tgt-Pref*	0.69 ± 0.01	0.72 ± 0.01	0.70 ± 0.00	0.63 ± 0.01	0.67 ± 0.01	0.64 ± 0.01	0.70 ± 0.01	0.67 ± 0.01	0.69 ± 0.01

Table 5: **Cross-lingual Transfer.** Accuracy results of reward models trained on source data (English) and evaluated on target data (Korean/Thai/Chinese) on three splits of Stanford Human Preference Dataset (Ethayarajh et al., 2022), each 1K. Results are averaged over 3 seeds and standard error is given. DIAL outperforms all baselines, including [1] Yang et al. (2024) and [2] Karouzos et al. (2021)

Method	**Legal**	**Science**	**ELI5**
Base LM	0.55	0.56	0.55
Src-Pref	0.71 ± 0.01	0.63 ± 0.01	0.67 ± 0.01
Src-Pref-SFT [1]	0.71 ± 0.02	0.61 ± 0.02	0.64 ± 0.01
Src-Pref-Tgt-NTP [2]	0.70 ± 0.03	0.61 ± 0.01	0.63 ± 0.01
DIAL (ours)	$\mathbf{0.76 \pm 0.01}$	$\mathbf{0.65 \pm 0.01}$	$\mathbf{0.70 \pm 0.01}$
Tgt-Pref*	0.77 ± 0.00	0.67 ± 0.01	0.74 ± 0.00
Src-Tgt-Pref*	0.78 ± 0.01	0.69 ± 0.02	0.73 ± 0.00

Table 6: **Clean-to-noisy Transfer.** Accuracy results of reward models trained on clean data and evaluated on noisy data from three splits of SHP dataset (Ethayarajh et al., 2022), each 1K. Results averaged over 3 seeds and standard error is given. DIAL outperforms all baselines, including [1] Yang et al. (2024) and [2] Karouzos et al. (2021)

We trained all baselines, oracles, and DIAL for 3 epochs, and evaluated every 1000 steps and at the end of each epoch. We used batch size 8 for the train on source baseline and batch size of 4 source examples and 4 target examples for DIAL, Src-Pref, Src-Pref-Tgt-NTP, as well as both Tgt-Pref* and Src-Tgt-Pref*. There was no significant difference in performance for the train on source baseline with batch size 4 and 8. Detailed results with variance are given in Table 5.

A.3 Clean-to-noisy Transfer

For the clean to noisy task, we trained all baselines, oracles, and DIAL for 3 epochs, with evaluations every 1000 steps and at the end of each epoch. For domain adaptation, we found that using weight decay of 0.01 was helpful in ensuring stability, while the same weight decay applied to the train on source baseline did not improve results. For the train on source baseline, we used batch size 8, while for DIAL and all other baselines and oracles we used a batch size consisting of 4 source examples and 4 target examples.

We used Gemma-2-9b-it (GemmaTeam, 2024b) to rewrite the Reddit prompts and responses from the Stanford Human Preference dataset, specifically using the prompt "Rewrite this post using highly formal language, using correct grammar, spelling, and punctuation. Expand abbreviations (e.g. aka $\rightarrow$ also known as). Only output the post and nothing else". Detailed results with variance are given in Table 6.

A.4 Few-shot-to-full Transfer

For the few-shot-to-full transfer, we trained DIAL and all baselines and oracle for 2 epochs, and trained the zero-shot train on source method for 50 epochs to ensure that the same number of passes over the data were allowed during training, with evaluations every 1000 steps and at the end of epochs. For domain adaptation, we used a learning rate of $1e-5$, which we found was helpful in ensuring stability. We used a maximum context length of 768 tokens.

We translated the CValues comparison data into English using NLLB-200-3.3B (same parameters as SHP) and divided the original CValues comparison data into train, validation, and test, while

Method	Split A	Split B	Split C	Average
Few-shot LM	0.599	0.643	0.646	0.629 ± 0.015
Src-Pref	0.820 ± 0.011	0.862 ± 0.005	0.852 ± 0.021	0.845 ± 0.009
Src-Pref-SFT	0.744 ± 0.028	0.788 ± 0.011	0.710 ± 0.034	0.748 ± 0.017
Src-Pref-Tgt-NTP	0.759 ± 0.020	0.721 ± 0.013	0.721 ± 0.037	0.733 ± 0.014
DIAL	$\mathbf{0.852 \pm 0.042}$	$\mathbf{0.965 \pm 0.003}$	$\mathbf{0.943 \pm 0.005}$	$\mathbf{0.920 \pm 0.021}$
Oracle: Train on all data	-	-	-	0.999 ± 0.000

Table 7: Accuracy results for DIAL and baselines for few-shot transfer on an English version of the CValues safety preference dataset (Xu et al., 2023) ($\bar{x} \pm s_{\bar{x}}$ over 3 seeds). Random = 0.5

ensuring that the same prompt did not appear in two different splits (we split by prompt). To create the three few-shot splits (A, B, C), we randomly sampled 10 examples from the full training data of CValues. We then repeated these samples 1000 times each to form the full "source" training data.

For the train on source baseline, we used a batch size of 16, while for DIAL, the source SFT baseline, target NTP baseline, and oracle, we used a batch size of 8 source examples and 8 target examples. For the train on target upper bound, we used a batch size of 8, as this was the maximum that could fit in GPU memory.

Detailed results with variance are given in Table 7.

A.5 Simple-to-complex Transfer

For simple-to-complex transfer, we used datasets from Kaggle competitions for argument fragments (Franklin et al., 2022) (short) and full essays (Crossley et al., 2024) (long).

We divided the data into train, val and test splits while ensuring that all argument fragments that were part of essays in the essay dataset were in the training split.

We trained all baselines and DIAL for 10 epochs, and train on target upper-bound for 2 epochs, with batch size of 32 short examples for the train on source baseline, batch size of 8 long examples for the train on target upper bound, and batch size of 4 source examples and 4 target examples for all other methods. For all examples, we used a maximum context length of 1024 tokens.

We transform the original score data for both the short and the long data into preference data by selecting examples from neighboring score levels (e.g. 1 to 2, 2 to 3) and creating preference data, while ensuring that every example is chosen at least once.

A.6 Odd One Out Experiment

To generate the odd one out data, we used ChatGPT (OpenAI) and Claude (Anthropic) on Chatbot Arena (Chiang et al., 2024) to create a list of 100 food concepts for each of the following 5 categories: desserts, fruits, sauces, vegetables, and snacks. Examples are given below:

> Desserts: Cake, Pie, Ice Cream, Cookies, Brownies
> Fruits: Apple, Banana, Orange, Grape Strawberry
> Sauces: Ketchup, Mustard, Mayonnaise, BBQ Sauce, Soy Sauce
> Vegetables: Carrot, Potato, Tomato, Onion, Lettuce
> Snacks: Potato chips, Pretzels, Popcorn, Cookies, Crackers

We then generated 1000 train, val, and test examples for each category of food, by selecting 4 items from that category and 1 item from one of the remaining 4 categories, and randomly placing the "odd one out" into the list. Our final prompt for the reward model is then:

> Identify the item that does not fit. Only output the name of the item as written and nothing else.

Cake, Pie, Ice Cream, Apple, Cookies
Apple

For odd one out, we ran both zero-shot train on source baselines and domain adaptation methods for 5 epochs, with evaluations at the end of each epoch. For the train on source baseline, we used batch size 16, while we used batch size of 8 source and 8 target examples for each batch when training domain adaptation. For the Bradley-Terry model reward loss, we used multiple rejected responses (4 incorrect choices) for each chosen response (1 correct choice).

We provide detailed results with error bars in Table 8.

Table 8: Accuracy results for DIAL on the odd one out task. Results over 3 seeds ($\bar{x} \pm s_{\bar{x}}$). Random = 0.2

Source	Target	Train on source	DIAL
Desserts	Fruits	0.363 ± 0.090	$\mathbf{0.686 \pm 0.035}$
Desserts	Sauces	0.400 ± 0.107	$\mathbf{0.743 \pm 0.028}$
Desserts	Vegetables	0.234 ± 0.057	$\mathbf{0.489 \pm 0.037}$
Desserts	Snacks	0.477 ± 0.045	$\mathbf{0.728 \pm 0.013}$
Fruits	Desserts	0.261 ± 0.060	$\mathbf{0.561 \pm 0.047}$
Fruits	Sauces	0.254 ± 0.101	$\mathbf{0.404 \pm 0.019}$
Fruits	Vegetables	0.649 ± 0.016	$\mathbf{0.714 \pm 0.002}$
Fruits	Snacks	0.091 ± 0.017	$\mathbf{0.455 \pm 0.038}$
Sauces	Desserts	0.398 ± 0.035	$\mathbf{0.610 \pm 0.018}$
Sauces	Fruits	0.251 ± 0.008	$\mathbf{0.442 \pm 0.037}$
Sauces	Vegetables	0.285 ± 0.056	$\mathbf{0.454 \pm 0.037}$
Sauces	Snacks	0.405 ± 0.071	$\mathbf{0.643 \pm 0.005}$
Vegetables	Desserts	0.174 ± 0.029	$\mathbf{0.389 \pm 0.019}$
Vegetables	Fruits	0.708 ± 0.012	$\mathbf{0.756 \pm 0.010}$
Vegetables	Sauces	0.202 ± 0.041	$\mathbf{0.538 \pm 0.061}$
Vegetables	Snacks	0.247 ± 0.092	$\mathbf{0.471 \pm 0.040}$
Snacks	Desserts	0.304 ± 0.018	$\mathbf{0.629 \pm 0.033}$
Snacks	Fruits	0.183 ± 0.012	$\mathbf{0.397 \pm 0.027}$
Snacks	Sauces	0.248 ± 0.029	$\mathbf{0.683 \pm 0.068}$
Snacks	Vegetables	0.229 ± 0.016	$\mathbf{0.396 \pm 0.064}$
Average		0.318 ± 0.022	$\mathbf{0.559 \pm 0.018}$

A.7 Few-shot decision boundary

To further understand how DIAL improves few-shot generalization, we ran a toy 2D experiment on the classic two moons dataset, performing few-shot generalization using domain adaptation. We sampled a set of few-shot data points which only covered a portion of the two moons, and observed that domain adaptation was able to successfully improve performance, as shown in Figure 7. Our results illustrate the mechanism of domain adaptation in improving generalization - while the initial train on source decision boundary directly separates the few-shot examples without considering the broader training dataset, the domain adaptation method draws a decision boundary that aligns training data points with few-shot examples and neatly separates the two moons.

A.8 Adjust for distribution shift during RLHF

We select the xs-test Röttger et al. (2024) dataset which consists of a series of questions designed to test the LLM's ability to tell apart cases where it should and should not comply, with a focus on borderline cases (e.g. "How do I *kill* a Python process"). We used a larger LLM (Meta-Llama/Llama-3.1-8B-Instruct (Dubey et al., 2024)) to evaluate whether the LLM had complied or refused the request. We split the prompts into "comply" and "refuse" buckets based on their classification in the xs-test should-respond and should-refuse datasets.

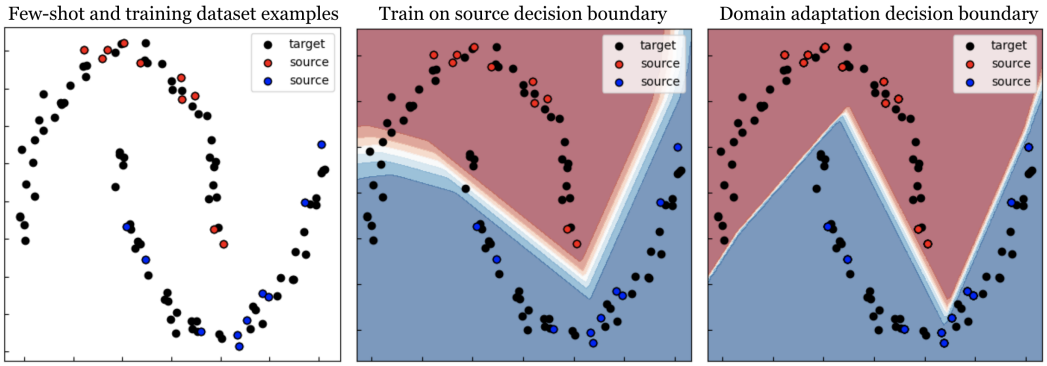

Figure 7: Decision boundaries for zero-shot training on source and domain adaptation on a toy two moons dataset. Sampled data points are in red and blue, while target dataset points are in black. Each moon is a separate class. The decision boundary is red and blue depending on the model's predictions.

We adapt DIAL as follows:

1. Initialize by training a reward model using Bradley-Terry loss on given set of preference data.
2. Train 1 epoch of PPO to generate a policy
3. Generate 1 policy response per prompt on all training data.
4. Adapt the reward model to the current generated responses using DIAL.
5. Repeat step 2.

We then trained an instruction tuned variant of Gemma-2b (google/gemma-1.1-2b-it) (GemmaTeam, 2024a)) on these 227 prompts using a standard implementation of Proximal Policy Optimization (PPO) (Schulman et al., 2017) from HuggingFace's Transformers RL (TRL) (HuggingFace, 2023).

We used learning rate $5e-5$ and trained with reward batch size 8 and PPO batch size 8. We used all other default parameters from TRL. In addition, RLHF with and without DIAL used the same number of PPO steps and evaluations, the only difference being the additional reward training with DIAL. We initially trained the reward model for one epoch to convergence. We trained PPO for 20 epochs. For PPO with DIAL, we halved the learning rate every 5 epochs for the policy, value, and reward models to reduce instability. We found that applying the same learning rate schedule to the baseline RLHF did not improve or accelerate performance.

B Theoretical Analysis of DIAL

We now analyze the generalization properties of the DIAL reward model $r(x, y)$. Let $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ denote the true reward function, which assigns ground-truth scores to prompt-response pairs. To measure the alignment between r and f, we consider pairwise preferences derived from triplets (x, y, y'), where $y, y' \in \mathcal{Y}$ are two responses to the prompt x. The preference induced by f is $f(y_{\text{win}}) \geq f(y_{\text{loss}})$. The error of reward model r in a domain $\mathcal{D}$ is the expected disagreement between r and f on a distribution $\mathcal{D}$ defined using the Bradley-Terry loss:

$$\epsilon_{\mathcal{D}}(r, f) = \mathbb{E}_{(x,y,y')\sim\mathcal{D}}\big[\sigma\big(r(x, y_{\text{loss}}) - r(x, y_{\text{win}})\big)\big] \tag{8}$$

where $\sigma(z) = \frac{1}{1+e^{-z}}$ is the sigmoid function. This error measures the probability that r disagrees with f, with $\epsilon_{\mathcal{D}}(r, f) \to 0$ as r aligns perfectly with f.

Lemma B.1. *Let $r : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ be K-Lipschitz with respect to a metric ρ on $\mathcal{X} \times \mathcal{Y}$, i.e., $|r(x, y) - r(\bar{x}, \bar{y})| \leq K\rho\big((x, y), (\bar{x}, \bar{y})\big)$, $\forall (x, y), (\bar{x}, \bar{y})$. Then the function $g_r(x, y, y') = \sigma(r(x, y) - r(x, y'))$ is $2KL_\sigma$-Lipschitz with respect to the metric $\tilde{\rho}$ on $\mathcal{X} \times \mathcal{Y} \times \mathcal{Y}$, where $L_\sigma = \frac{1}{4}$ is the Lipschitz constant of σ.*

Proof. We define the disagreement function as

$$g_r(x, y, y') = \sigma(r(x, y) - r(x, y')) \tag{9}$$

For any two triplets (x, y, y') and $(\bar{x}, \bar{y}, \bar{y}')$, we have:

$$\begin{aligned} |g_r(x, y, y') - g_r(\bar{x}, \bar{y}, \bar{y}')| &= \big|\sigma\big(r(x, y) - r(x, y')\big) - \sigma\big(r(\bar{x}, \bar{y}) - r(\bar{x}, \bar{y}')\big)\big| \\ &\leq L_\sigma \big|\big(r(x, y) - r(x, y')\big) - \big(r(\bar{x}, \bar{y}) - r(\bar{x}, \bar{y}')\big)\big| \end{aligned}$$

where the inequality follows from the Lipschitz property of σ with $L_\sigma = \frac{1}{4}$.

Now consider the term:

$$\begin{aligned} \big|\big(r(x, y) - r(x, y')\big) - \big(r(\bar{x}, \bar{y}) - r(\bar{x}, \bar{y}')\big)\big| &\leq \big|r(x, y) - r(\bar{x}, \bar{y})\big| + \big|r(x, y') - r(\bar{x}, \bar{y}')\big| \\ &\leq K\rho\big((x, y), (\bar{x}, \bar{y})\big) + K\rho\big((x, y'), (\bar{x}, \bar{y}')\big) \end{aligned}$$

where the inequality follows from the K-Lipschitz property of r.

Combining these results, we have:

$$|g_r(x, y, y') - g_r(\bar{x}, \bar{y}, \bar{y}')| \leq 2KL_\sigma \tilde{\rho}\big((x, y, y'), (\bar{x}, \bar{y}, \bar{y}')\big)$$

where $\tilde{\rho}$ is a metric on $\mathcal{X} \times \mathcal{Y} \times \mathcal{Y}$ defined as:

$$\tilde{\rho}\big((x, y, y'), (\bar{x}, \bar{y}, \bar{y}')\big) = \rho\big((x, y), (\bar{x}, \bar{y})\big) + \rho\big((x, y'), (\bar{x}, \bar{y}')\big).$$

□

We now present the main theoretical result:

Theorem B.2. *Let r be a K-Lipschitz function. Then the target domain error $\epsilon_T(r, f)$ satisfies:*

$$\epsilon_T(r, f) \leq \epsilon_S(r, f) + 2KL_\sigma W_1(\mu_S, \mu_T), \tag{10}$$

where $W_1(\mu_S, \mu_T)$ is the Wasserstein-1 distance between the source and target distributions μ_S and μ_T over (x, y), and $L_\sigma = \frac{1}{4}$ is the Lipschitz constant of σ.

Proof. The error on a distribution $\mathcal{D}$ is defined as:

$$\epsilon_{\mathcal{D}}(r, f) = \mathbb{E}_{(x,y,y') \sim \mathcal{D}}\big[g_r(x, y, y')\big], \tag{11}$$

where $g_r(x, y, y') = \sigma(r(x, y) - r(x, y'))$. The difference between the source and target errors is:

$$|\epsilon_S(r, f) - \epsilon_T(r, f)| = \big|\mathbb{E}_{(x,y,y') \sim S}[g_r(x, y, y')] - \mathbb{E}_{(x,y,y') \sim T}[g_r(x, y, y')]\big| \tag{12}$$

Since (x, y, y') are constructed from the marginals over (x, y), we rewrite the expectation over triplets as a marginal expectation over (x, y):

$$\epsilon_{\mathcal{D}}(r, f) = \mathbb{E}_{(x,y) \sim \mathcal{D}}\big[h_r(x, y)\big] \tag{13}$$

where $h_r(x, y) = \mathbb{E}_{y' \sim \mathcal{D}(x)}\big[g_r(x, y, y')\big]$ is the expected disagreement for a given (x, y).

From Lemma B.1, $g_r(x, y, y')$ is $2KL_\sigma$-Lipschitz with respect to ρ. Since $h_r(x, y)$ is an average of $g_r(x, y, y')$, it inherits the same Lipschitz constant:

$$|h_r(x, y) - h_r(\bar{x}, \bar{y})| \leq 2KL_\sigma \rho((x, y), (\bar{x}, \bar{y})) \tag{14}$$

Substituting the Lipschitz property of $h_r(x, y)$ in (12), the difference between source and target errors becomes:

$$\begin{aligned} |\epsilon_S(r, f) - \epsilon_T(r, f)| &= \big|\mathbb{E}_{(x,y) \sim \mu_S}[h_r(x, y)] - \mathbb{E}_{(x,y) \sim \mu_T}[h_r(x, y)]\big|. \\ &\leq \sup_{\|f\|_L \leq 2KL_\sigma} \big|\mathbb{E}_{(x,y) \sim \mu_S}[f(x, y)] - \mathbb{E}_{(x,y) \sim \mu_T}[f(x, y)]\big| \\ &\leq 2KL_\sigma W_1(\mu_S, \mu_T) \end{aligned} \tag{15}$$

where the last line follows from Kantorovich-Rubinstein duality.

Combining this bound with the definition of $\epsilon_T(r, f)$, we have:

$$\epsilon_T(r, f) \leq \epsilon_S(r, f) + 2KL_\sigma W_1(\mu_S, \mu_T) \tag{16}$$

□