ATRI: Mitigating Multilingual Audio Text Retrieval Inconsistencies by Reducing Data Distribution Errors

Anonymous ACL submission

Abstract

Multilingual audio-text retrieval (ML-ATR) is a challenging task that aims to retrieve audio clips or multilingual texts from databases. However, existing ML-ATR schemes suffer from inconsistencies for instance similarity matching across languages. We theoretically analyze the inconsistency in terms of both multilingual modal alignment direction error and weight error, and propose the theoretical weight error upper bound for quantifying the inconsistency. Based on the analysis of the weight error upper bound, we find that the inconsistency problem stems from the data distribution error caused by random sampling of languages. We propose a consistent ML-ATR scheme using 1-to-k contrastive learning and audio-English co-anchor contrastive learning, aiming to mitigate the negative impact of data distribution error on recall and consistency in ML-ATR. Experimental results on the translated AudioCaps and Clotho datasets show that our scheme achieves state-of-the-art performance on recall and consistency metrics for eight mainstream languages, including English. Our code will be available at https://github.com/ATRI-ACL/ATRI-ACL.

1 Introduction

In an audio-text retrieval (ATR) task, the system searches for matching audio clips or text captions in a database based on cross-modality queries (Elizalde et al., 2019; Koepke et al., 2022; Zhu et al., 2024). With the convergence of audio and text, ATR techniques have seen significant advancements in recent years and are widely applied in content retrieval and multimedia information retrieval. However, most existing ATR systems are designed for monolingual retrieval, and research on multilingual audio-text retrieval (ML-ATR) remains limited (Yan et al., 2024). The shift to ML-ATR brings new challenges, particularly in dealing with high multilingual recall and ensuring the consistency (Nie et al., 2024) of multilingual retrieval results.

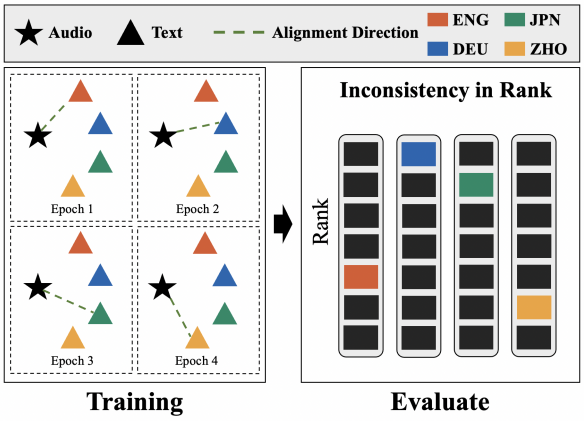

Figure 1: **An illustration of inconsistency issue in current ML-ATR scheme**.

To the best of our knowledge, the existing mainstream ML-ATR scheme has a model training process as shown in Fig. 1, which pairs audio with randomly selected linguistic text in each epoch. This may not allow the model to learn the embedding space of audio and multilingual texts very well, which not only reduces the recall of retrieval, but also makes it difficult to obtain the same retrieval results for audio and multilingual text instances in different languages.

In this paper, we theoretically analyzes the causes of the inconsistency problem in ML-ATR. We first visualize the inconsistency problem in terms of the modal alignment direction error. The alignment direction error leads to the gradient error, which in turn invites the model weights to fail to converge to the optimal weights for multilingual modal alignment during the training process. We further heuristically derive theoretical upper bounds on the weight errors to quantify the adverse effects of inconsistency on the model weights. We analyze the composition of the weight error upper bound and conclude that the root cause of the error inconsistency is the data distribution error in training.

Based on the theoretical analysis, we propose a scheme to mitigate the inconsistency of ML-ATR, called ATRI. ATRI consists of two training strategies: 1-to-K Contrastive Learning (KCL) for the retrieval-performance-first scenario, and Audio-English Co-Anchor Contrastive Learning (CACL) for the overhead-first scenario. KCL theoretically eliminates the data distribution errors in each training epoch, thus achieving state-of-the-art performance in recall and consistency metrics. CACL aligns the other languages with audio and English text to correct the modal alignment direction and reduce the data distribution error. Compared to existing ML-ATR schemes, CACL improves retrieval recall and consistency while offering advantages in training time and GPU memory overhead over KCL.

Our contributions are shown below:

- We analyze the inconsistency in terms of analyzing the modal alignment direction error and weighting error, and demonstrate an upper bound on the weighting error. We further conclude that the root cause of the inconsistency of existing ML-CLAP schemes lies in the distribution error of the training data.

- We propose ATRI, which solves the inconsistency problem in multilingual audio text retrieval by reducing the data distribution error and correcting the modality alignment direction. ATRI contains the CACL and KCL training strategies for overhead-first and performance-first requirements, respectively.

- We evaluate the proposed scheme using the AudioCaps and Clotho datasets translated by Deepseek. The results show that ATRI effectively improves recall and consistency in both monolingual English ATR and ML-ATR tasks, achieving state-of-the-art performance.

2 Related Work

Audio-text retrieval (ATR) (Lou et al., 2022; Xie et al., 2024; Xin et al., 2024) is a task that matches audio with text, which has seen significant advancements and widespread applications in recent years. The prevailing approach involves constructing a shared embedding space for audio and text, enabling seamless feature alignment and retrieving results based on similarity rankings. Widely adopted methods include CLIP-inspired (Yu et al., 2022; Li et al., 2022) comparative audio-text pretraining (Elizalde et al., 2023; Wu et al., 2022; Guzhov et al., 2022) and the triplet-loss method (Mei et al., 2022), both of which have achieved success in learning audio-text joint embedding spaces.

Existing ATR methods predominantly focus on English-centric monolingual tasks, with few solutions for multilingual scenarios (Yan et al., 2024). The scarcity of large-scale, accurately annotated non-English audio-caption datasets has led current ML-ATR methods to rely heavily on machine translation (Tiedemann and Thottingal, 2020; Team et al., 2022) to convert English datasets into multilingual versions. This translation-based strategy (Cousin et al., 2023; Yan et al., 2024) has demonstrated its effectiveness in enhancing datasets for multilingual use, significantly improving the recall performance of ATR systems.

However, existing ML-ATR scheme (Yan et al., 2024) use audio-text pairs with randomly selected languages for training. As analyzed in Sect. 3, the training method employed presents significant challenges in achieving convergence to the optimal weights. This difficulty not only exacerbates issues related to inconsistent cross-language retrieval, but also leads to a degradation in the retrieval performance, particularly in terms of both recall and accuracy.

3 Definition and Inconsistency Analysis

3.1 Formal Definition of ML-ATR

Audio-text retrieval is the task of learning cross-modality alignment between audio and multilingual text captions. Contrastive learning (Chuang et al., 2020) has become the most effective method for learning expressive cross-modality embedding spaces.

Denote a dataset $D = \{(a_i, t_{i1}, ...t_{iK})\}_{i=1}^{N}$ as a multilingual audio text retrieval dataset, where N denotes the size of dataset, K refers the total language number in the dataset, a_i denotes the audio in i-th data, t_{ik} denotes the k-th language in i-th data. Given an audio encoder $f_\theta(\cdot)$ and a multilingual text encoder $g_\phi(\cdot)$, we denote the joint probability distribution as:

$$p(a_i, t_{ik}) = \frac{\exp\left(s(f_\theta(a_i), g_\phi(t_{ik}))/\tau\right)}{\sum_{j=1}^{N}\sum_{l=1}^{K}\exp\left(s(f_\theta(a_j), g_\phi(t_{jl}))/\tau\right)}, \quad (1)$$

$$p(a_i, t_i) = \frac{\exp\left(s(f_\theta(a_i), g_\phi(t_i))/\tau\right)}{\sum_{j=1}^{N}\exp\left(s(f_\theta(a_j), g_\phi(t_j))/\tau\right)}, \quad (2)$$

$s(\cdot)$ denotes the cosine similarity between audio and text embedding. The ideal optimization function of learning the embedding space is

$$\max_{\theta,\phi} \sum_{i=1}^{N} \sum_{k=1}^{K} p(a_i, t_{ik}) \mathbb{E}_{(a_i,t_{ik})}[log\ p(a_i, t_{ik})]. \quad (3)$$

However, instead of training all the languages of a piece of data in an epoch, the existing ML-ATR scheme randomly selects the text of a language to do the training. For each epoch e, a set of random numbers $Q = \{q_1, ...q_N\}, q_i \overset{R}{\leftarrow} \{1, ...K\}$. The optimization function they used is formalized as:

$$p'_e(a_i, t_{iq_i}) = \frac{\exp\left(s(f_\theta(a_i), g_\phi(t_{iq_i}))/\tau\right)}{\sum_{j=1}^{N} \exp\left(s(f_\theta(a_j), g_\phi(t_{jq_j}))/\tau\right)}, \quad (4)$$

$$\max_{\theta,\phi} \sum_{i=1}^{N} p'_e(a_i, t_{iq_i}) \mathbb{E}_{(a_i,t_{iq_i})}[log\ p'_e(a_i, t_{iq_i})]. \quad (5)$$

The probability distribution $p'_e(a_i, t_{iq_i})$ of their scheme is not the same as the original probability distribution $p(a_i, t_{ik})$. This results in a model that does not fit the training data perfectly, making modality alignment ineffective, which in turn results in reduced recall and inconsistency problems.

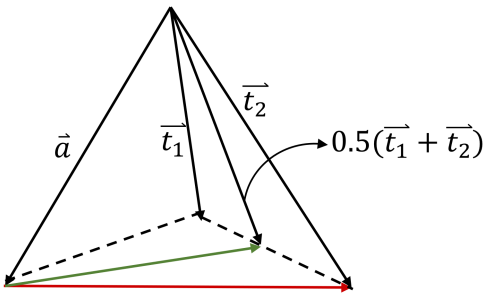

Figure 2: **A visual illustration of inconsistency due to modality alignment errors**.

3.2 Analysis of the Inconsistency Issue

We first analyze the issue of inconsistency from the perspective of modality alignment directional errors. As shown in Fig. 2, an intuitive example of modality alignment error is illustrated. Consider a simple case of bilingual audio-text retrieval, let the embedding of an audio sample be $\vec{a}$, and the embeddings of the corresponding texts in two languages be $\vec{t_1}$ and $\vec{t_2}$. Ideally, the audio embedding $\vec{a}$ should be aligned with the combined representation of both text embeddings $\frac{1}{2}(\vec{t_1} + \vec{t_2})$ (indicated by the green arrow). However, in existing ML-ATR schemes, the audio embedding is only aligned with the text embedding of a randomly selected language within each epoch. For instance, if the selected language is t_2, the audio embedding $\vec{a}$ will be aligned solely towards $\vec{t_2}$ (indicated by the red arrow). The angle between the red and green arrows is the modality alignment direction error, which makes the audio and multilingual text modes not well aligned.

It's obvious that incorrect alignment introduces noise to the gradient, leading to errors between the model weights and their optimal values, making the model's retrieval recall and consistency metrics degrade. We give a theoretical weight error upper bound and analyze its composition to mitigate the inconsistency problem and improve retrieval recall. The detailed proof can be found in Appendix C.

We assume that the optimization algorithm is stochastic gradient descent (SGD) (Zinkevich et al., 2010) to heuristically analyse the upper bound of the weight error. Given that the number of training steps per epoch T, the data distribution obtained by randomly sampling the language according to the existing ATR scheme is denoted as p'_e, and the original data distribution is denoted as p. $\mathbf{w}'_{eT}$ denotes the model weight in the T-th step under the e-th epoch trained with the data distribution p'_e, whereas $\mathbf{w}_{eT}$ denotes the weight that is trained with the data distribution p. If the gradient $\nabla_{\mathbf{w}} \mathbb{E}_{(a,t)}[log\ p(a,t)]$ is $\lambda_{(x,y)}$-Lipschitz (Béthune et al., 2023), then we have the following inequality for weight error upper bound:

$$\begin{aligned} &||\mathbf{w}_{eT} - \mathbf{w}'_{eT}|| \\ \leq &a^T ||\mathbf{w}_{(e-1)T} - \mathbf{w}'_{(e-1)T}|| + \\ &\eta \sum_{(a,t)} ||p(a,t) - p'_e(a,t)|| \sum_{j=1}^{T-1} (a^j g_{max}(\mathbf{w}_{eT-1-j})), \end{aligned} \quad (6)$$

$$g_{max}(\mathbf{w}) = max_{(a,t)} ||\nabla_{\mathbf{w}} \mathbb{E}_{(a,t)}[log\ p(a,t)]||, \quad (7)$$

$$a = 1 + \eta \sum_{(a,t)} p'_e(a,t) \lambda_{(x,y)}. \quad (8)$$

Note: The weight $\mathbf{w}$ consists of the parameter θ for the audio encoder f_θ and the parameter ϕ for the multilingual text encoder g_ϕ in ML-ATR. The data distributions p and p'_e correspond to the Eq. (2) and (4), respectively. For simplicity, we denote (a, t) as all audio-text pairs in the batch of the T-th step, where the text t can be in any one of the

languages. $\sum_{(a,t)} ||p(a,t) - p'_e(a,t)||$ denotes the data distribution error in the batch at step T.

Detailed proof of Eq (6) can be found in Appendix C. Based on Eq. (6), we have the following results:

- Intuitively, the weight error $||\mathbf{w}_{eT} - \mathbf{w}'_{eT}||$ comes from two main sources. One is the weight error after the $(e-1)$-th epoch, i.e. $||\mathbf{w}'_{(e-1)T} - \mathbf{w}_{(e-1)T}||$. The other is caused by the probabilistic distances of the data distributions, i.e. $\sum_{(a,t)} ||p'_e(a,t) - p(a,t)||$. Since $a \geq 1$, the error from both sources increases with epoch and step. In addition, the weight error is also affected by the learning rate η, the number of training steps T and the maximum gradient $g_{max}(\mathbf{w}_{eT-1-j})$.

- Further expansion of Eq. (6) shows that the weighting error arises from the data distribution error of each epoch. Expanding $||\mathbf{w}_{(e-1)T} - \mathbf{w}'_{(e-1)T}||$ in Eq. (6), we find it consist of $||\mathbf{w}_{(e-2)T} - \mathbf{w}'_{(e-2)T}||$ and $||p(a,t) - p'_{e-1}(a,t)||$. Further expanding Eq. (6) to the weight error in 1-th epoch, it can be concluded that the weight error of the existing ML-ATR scheme comes from the data distribution error $\sum_{i=1}^{e} \sum_{(a,t)} ||p(a,t) - p'_i(a,t)||$ due to the randomly selected languages in each epoch. We can mitigate the inconsistency problem and improve the recall by reducing the weight error upper bound by reducing the data distribution error for each epoch.

4 Proposed ML-ATR Scheme

We propose two methods to reduce the data distribution error during training. One is 1-to-K contrastive learning, which has a higher memory overhead. The other is audio-English co-anchor contrastive learning, which achieves performance close to 1-to-K Contrastive Learning while approximating the memory overhead to the existing ML-ATR scheme. Here are the details of the two methods.

4.1 1-to-K Contrastive Learning

Building on our theoretical analyses, we conclude that reducing data distribution error is critical for addressing the inconsistency problem in multilingual audio-text retrieval. To achieve this, we propose 1-to-K Contrastive Learning (KCL), a training strategy that replaces random language sampling with the simultaneous use of all K linguistic texts corresponding to each audio instance. This approach theoretically eliminates data distribution error, corrects modal alignment direction, and significantly enhances both the recall and consistency of retrieval performance. The loss function $\mathcal{L}^{at}_{kcl}$ for the proposed 1-to-K Contrastive Learning in ML-ATR is defined as follows:

$$\mathcal{L}_{kcl} = \frac{1}{2NK}(\mathcal{L}^{a2t}_{kcl} + \mathcal{L}^{t2a}_{kcl}). \tag{9}$$

The loss function $\mathcal{L}^{at}_{kcl}$ consists of two parts, $\mathcal{L}^{a2t}_{kcl}$ and $\mathcal{L}^{t2a}_{kcl}$, and they are calculated as follows:

$$\mathcal{L}^{a2t}_{kcl} = -\sum_{k=1}^{K}\sum_{i=1}^{N} log \frac{\exp(s(f_\theta(a_i), g_\phi(t_{ik}))/\tau)}{\sum_{j=1}^{N}\exp(s(f_\theta(a_i), g_\phi(t_{jk}))/\tau)}, \tag{10}$$

$\mathcal{L}^{a2t}_{kcl}$ denotes the contrastive learning loss function from audio to multilingual text.

$$\mathcal{L}^{t2a}_{kcl} = -\sum_{k=1}^{K}\sum_{i=1}^{N} log \frac{\exp(s(g_\phi(t_{ik}), f_\theta(a_i))/\tau)}{\sum_{j=1}^{N}\exp(s(g_\phi(t_{ik}), f_\theta(a_j))/\tau)}, \tag{11}$$

$\mathcal{L}^{t2a}_{kcl}$ denotes the contrastive learning loss function from multilingual text to audio.

K is the number of languages and N is the number of data instances. As shown in Tab. 4, including multiple multilingual texts in 1-to-K contrastive learning increases GPU memory usage and training time. In practical ML-ATR applications, supporting more languages amplifies these overheads compared to existing schemes.

To address this, we further propose CACL, which improves retrieval consistency and recall without significantly increasing overhead.

4.2 Audio-English Co-Anchor Contrastive Learning

To reduce the weighting error with as little increase in training time and GPU memory consumption as possible, we propose audio-English co-anchor contrastive learning (CACL). During the training process, each data takes its audio, English text, and text in other random languages and does contrastive learning with each other.

For each epoch, given a set of random numbers $Q = \{q_1, ...q_N\}, q_i \overset{R}{\leftarrow} \{2, ...K\}$, get the triplet of the training data (a_i, t_{i1}, t_{iq_i}), where a_i denotes i-th audio, t_{i1} denotes the English text, and t_{iq_i} denotes the text of q_i-th language. We have the training loss $\mathcal{L}_{cacl}$ shown below:

$$\mathcal{L}_{cacl} = \frac{1}{6N}(\mathcal{L}^{ae}_{cacl} + \mathcal{L}^{at}_{cacl} + \mathcal{L}^{et}_{cacl}). \quad (12)$$

The loss function $\mathcal{L}_{cacl}$ consists of three components $\mathcal{L}^{ae}_{cacl}, \mathcal{L}^{at}_{cacl}, \mathcal{L}^{et}_{cacl}$. All three components are based on the following general contrastive learning loss formulation:

$$\begin{aligned}\mathcal{L}^{uv}_{cacl} = &-\sum_{i=1}^{N} log \frac{\exp(s(u_i, v_i)/\tau)}{\sum_{j=1}^{N} \exp(s(u_i, v_j)/\tau)} \\ &-\sum_{i=1}^{N} log \frac{\exp(s(v_i, u_i)/\tau)}{\sum_{j=1}^{N} \exp(s(v_i, u_j)/\tau)},\end{aligned} \quad (13)$$

where u_i and v_i represent input embeddings from different modalities or languages. The three components are defined as follows:

- **Audio-English Alignment** ($\mathcal{L}^{ae}_{cacl}$):

  $u_i = f_\theta(a_i)$ represents audio embeddings, and $v_i = g_\phi(t_{i1})$ represents English text embeddings.

- **Audio-Multilingual Alignment** ($\mathcal{L}^{at}_{cacl}$): $u_i = f_\theta(a_i)$ represents audio embeddings, and $v_i = g_\phi(t_{iq_i})$ represents text embeddings in a randomly selected language.

- **English-Multilingual Alignment** ($\mathcal{L}^{et}_{cacl}$): $u_i = g_\phi(t_{i1})$ represents English text embeddings, and $v_i = g_\phi(t_{iq_i})$ represents text embeddings in a randomly selected language.

The effectiveness of audio-English CACL can be explained from two perspectives:

- From the perspective of modality alignment (Fig. 2), the loss function $\mathcal{L}^{et}_{cacl}$ in CACL brings embeddings of English and other languages closer, reducing the distance between the text embedding $\vec{t_1}, \vec{t_2}$ and the mean $\frac{1}{2}(\vec{t_1} + \vec{t_2})$ and minimizing the deviation in the modality alignment direction of audio and text.

- From the perspective of data distribution error $\sum_{(a,t)} ||p(a,t) - p'_e(a,t)||$ in Eq. (6), CACL's loss functions $\mathcal{L}^{ae}_{cacl}, \mathcal{L}^{at}_{cacl}$ ensures that the model learns more pairs of audio texts in an epoch. The text in them also contains a large percentage of high-quality English text. It makes the data distribution in CACL closer to the original one, and reduces the weight error of the model.

Note that in CACL, the number of texts used for training in each epoch does not increase with the number of languages, which effectively reduces both GPU memory and time overhead in ML-ATR scenarios with a large number of languages. Our experimental results illustrate that CACL approximates the training time and explicit memory overhead of existing ML-ATR schemes, yet achieves recall and consistency metrics close to those of 1-to-K comparative learning.

5 Experiments

5.1 Dataset

We employ the AudioCaps (Kim et al., 2019), and Clotho (Drossos et al., 2020) for our experiments. AudioCaps includes around 49,000 audio samples, each lasting about 10 seconds. Each audio is paired with a single sentence in the training set, while in both the validation and test sets, each audio has five associated sentences. The Clotho dataset consists of 6,974 audio samples, each ranging from 15 to 30 seconds long and annotated with five sentences. It is split into 3,839 training samples, 1,045 validation samples, and 1,045 test samples.

Additionally, to assess our scheme's performance in the ML-ATR task, we use the Deepseek (Bi et al., 2024) API to translate the text from AudioCaps and Clotho into seven widely spoken languages, including French (fra), German (deu), Spanish (spa), Dutch (nld), Catalan (cat), Japanese (jpn), and Chinese (zho).

5.2 Models

Audio Encoder: We utilize the recently proposed CED-Base model (Dinkel et al., 2024), a vision transformer with 86 million parameters for the Audio Encoder. Trained on Audioset through knowledge distillation from a large teacher ensemble, the model processes 64-dimensional Mel-spectrograms derived from a 16 kHz signal. It then extracts non-overlapping 16 × 16 patches from the spectrogram, resulting in 248 patches over a 10-second input (4 × 62).

Text Encoder: The key to multilingual audio-text retrieval is the text encoder's ability to handle texts in multiple languages. In this work, we focus solely on the SONAR-TE model (Duquenne et al., 2023). SONAR-TE generates a single vector bottleneck to encapsulate the entire text, avoiding the token-level cross-attention typically employed in conventional sequence-to-sequence machine translation

models. The fixed-size text representation is derived by pooling the token-level outputs from the encoder. In the following sections, SONAR refers specifically to the text encoder.

5.3 Setup

We use ML-CLAP (Yan et al., 2024) as the baseline, which is the state-of-the-art for ML-ATR tasks. To have a fair comparison, the model is initialized using the pre-trained weights of ML-CLAP and is further fine-tuned on our multilingual Audiocaps and Clotho datasets using three training methods: ML-CLAP, proposed CACL, and proposed KCL.

All models were fine-tuned for 10 epochs on a single A100 80GB PCIe GPU with a batch size of 24, a learning rate of 5×10^{-6}, using the Adam optimizer. The temperature hyperparameter τ was set to 0.07 for all configurations. The audio was sampled at 1.6×10^4. We selected the model with the best recall performance during the fine-tuning period for each scheme to perform the experiments.

5.4 Evaluation Metric

We use the recall of rank k (R@k) and the average precision of rank 10 (mAP10) as the metrics for the retrieval performance of the model to show that reducing data distribution errors improves the retrieval performance in each language. R@k refers to the fact that for a query, R@k is 1 if the target-value item occurs in the first k retrieved items, and 0 otherwise. mAP10 calculates the average precision of all the queries among the first 10 retrieved results. With these two metrics, we can comprehensively evaluate the retrieval performance of the model on multilingual datasets.

To assess the consistency of the embedding space across languages, we use three metrics: embedding space gap $\vec{\triangle}_{gap,k}$ (Liang et al., 2022), average embedding distance $\vec{\triangle}_{dis,k}$, mean rank variance (MRV). The computation of $\vec{\triangle}_{gap,k}$, $\vec{\triangle}_{dis,k}$ and MRV is shown below:

$$\vec{\triangle}_{gap,k} = \frac{1}{N}\sum_{i=1}^{N} g_\phi(t_{i1}) - \frac{1}{N}\sum_{i=1}^{N} g_\phi(t_{ik}), \quad (14)$$

$$\vec{\triangle}_{dis,k} = \frac{1}{N}\sum_{i=1}^{N} ||g_\phi(t_{i1}) - g_\phi(t_{ik})||, \quad (15)$$

$$MRV = \frac{1}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K} |Rank_{ik} - \overline{Rank_j}|^2. \quad (16)$$

$\vec{\triangle}_{gap,k}$ and $\vec{\triangle}_{dis,k}$ denotes the embedding space gap and average embedding distance between English and k-th language respectively. $Rank_{ik}$ denotes the similarity ranking of the k-th language under the i-th data, and $\overline{Rank_i}$ denotes the average similarity ranking under the i-th data.

5.5 Evaluation Result of Recall and Precision

We present a detailed numerical comparison analysis of the experiment results in Tab 1, focusing on the performance improvements of our proposed methods, CACL and KCL, over the baseline ML-CLAP across various languages and datasets.

5.5.1 Analysis of Evaluation Results

Overall, the proposed CACL and KCL consistently outperform ML-CLAP across the majority of languages and datasets in terms of recall at 1 (R@1), recall at 5 (R@5), and mean average precision at the top 10 results (mAP10) for both Text-to-Audio (T2A) and Audio-to-Text (A2T) tasks. Notably, our proposed KCL achieves state-of-the-art performance, delivering a 5% improvement in R@1 for the English-oriented monolingual ATR task and a 4.3% improvement in R@1 for the multilingual ATR task compared to ML-CLAP. This experimental result corroborates our theoretical analysis of the weighting error in Sect. 3. Here is the detailed analysis:

CACL's average metrics across languages are higher than ML-CLAP, while KCL's average metrics across languages have further improvement compared to CACL. Our theoretical analyses in Sect can explain this phenomenon. 3:

- CACL uses audio and text together as the anchor point for modality alignment in other languages, which can effectively reduce the data distribution error and modality alignment error, thus achieving better modality alignment results and improved metrics compared to ML-CLAP.

- Compared to CACL, which mitigates data distribution errors, KCL theoretically eliminates these errors. As a result, KCL achieves superior modality alignment compared to CACL, leading to further improvements in both recall and precision.

5.5.2 Analysis of Special Situations

Occasional Metric Anomalies: We observed occasional anomalies where a small proportion of KCL

Table 1: Recall and precision results for baseline and our method under multilingual AudioCaps and Clotho dataset

Scheme	Lang	AudioCaps						Clotho					
		T2A			A2T			T2A			A2T		
		R@1	R@5	mAP10	R@1	R@5	mAP10	R@1	R@5	mAP10	R@1	R@5	mAP10
ML-CLAP	eng	47.31	80.65	61.44	64.91	90.54	38.62	25.98	54.5	38.15	34.03	61.05	21.19
	fra	45.88	78.92	60.01	61.65	89.39	37.90	24.42	52.51	36.24	30.95	57.59	19.66
	deu	45.60	79.49	59.93	62.65	88.76	37.88	24.08	52.61	36.40	31.62	57.40	19.39
	spa	45.00	79.32	59.62	63.04	88.86	37.38	24.05	52.75	36.22	31.43	57.98	19.65
	nld	45.88	79.64	59.92	62.50	90.33	37.72	23.88	51.53	35.73	31.40	57.98	19.58
	cat	44.36	77.89	58.58	61.65	87.60	36.43	22.83	50.84	34.80	30.91	56.43	18.26
	jpn	43.04	76.86	57.54	59.45	87.81	35.20	23.04	50.34	34.89	31.43	56.55	18.77
	zho	41.70	74.72	55.74	53.67	84.76	33.38	21.65	48.84	33.53	28.41	56.14	17.26
	avg	44.84	78.43	59.09	61.19	88.50	36.81	23.84	51.74	35.74	31.27	57.64	19.22
our CACL	eng	49.05	82.14	63.07	66.31	**91.49**	39.41	26.36	55.19	38.68	34.71	61.34	**21.57**
	fra	46.86	79.97	60.83	63.23	89.48	37.92	**24.90**	**53.09**	36.67	**32.40**	58.55	19.85
	deu	46.21	80.08	60.62	63.13	**89.91**	38.14	24.51	52.86	36.52	**33.36**	58.07	19.49
	spa	46.68	80.52	60.90	63.23	**90.12**	37.45	**24.59**	52.71	**36.72**	32.40	58.17	19.75
	nld	47.41	80.23	61.22	63.23	**90.86**	37.95	24.15	51.75	36.05	32.21	58.65	19.5
	cat	45.27	78.61	59.43	61.23	88.44	36.49	23.28	51.42	35.17	30.67	56.05	18.67
	jpn	44.76	78.50	58.97	61.55	88.67	34.91	23.36	51.53	35.28	**31.82**	**58.26**	18.99
	zho	42.01	76.02	56.23	56.40	86.65	33.93	22.50	49.42	34.01	27.69	**57.59**	17.48
	avg	46.03	79.50	60.15	62.28	89.45	37.02	24.20	52.24	36.27	31.90	58.33	19.41
our KCL	eng	**49.68**	**82.44**	**63.34**	**66.59**	91.34	**40.52**	**26.67**	**55.46**	**38.97**	**36.34**	**64.13**	21.36
	fra	**47.79**	**80.52**	**61.53**	**63.41**	**89.57**	**39.21**	24.61	52.73	**36.79**	31.82	**60.76**	**20.02**
	deu	**47.81**	**80.81**	**61.78**	**63.34**	89.28	**39.02**	**24.90**	**53.25**	**37.02**	33.17	**59.61**	**19.90**
	spa	**47.33**	**80.67**	**61.49**	**63.76**	89.39	**38.73**	24.31	**52.96**	36.55	**33.36**	**61.25**	**20.27**
	nld	**47.92**	**80.76**	**61.70**	**63.55**	90.52	**39.14**	**24.53**	**52.51**	**36.61**	**33.55**	**62.30**	**19.98**
	cat	**46.44**	**79.62**	**60.42**	**62.71**	**89.49**	**37.65**	**23.67**	**51.86**	**35.70**	**31.53**	**57.98**	**18.90**
	jpn	**45.27**	**78.86**	**59.49**	**62.28**	**89.16**	**36.81**	**23.65**	**52.17**	**35.68**	31.25	57.50	**19.49**
	zho	**42.25**	**76.38**	**56.75**	**57.66**	**87.28**	**34.79**	**23.09**	**49.90**	**34.60**	**30.48**	56.34	**17.85**
	avg	**46.81**	**80.00**	**60.81**	**62.91**	**89.50**	**38.23**	**24.42**	**52.60**	**36.49**	**32.68**	**59.98**	**19.72**

metrics were lower than CACL metrics, and some CACL metrics were lower than ML-CLAP metrics. We attribute these discrepancies to noise in the dataset. Specifically, the weight error in Eq. (6) represents the difference between the current and optimal model weights for fitting the training data. If the dataset is too noisy, the optimal weights may not improve the test set's performance. As a result, KCL and CACL, which have lower weight errors, may still underperform ML-CLAP on certain metrics. The higher frequency of such anomalies in the noisier Clotho dataset, compared to Audiocaps, supports this explanation. Given that these anomalies are rare among the 108 evaluated metrics, we consider them acceptable and conclude that they do not impact the overall performance advantage of CACL and KCL in the ML-ATR task.

Performance Gaps Across Languages: The lower metrics for Japanese and Chinese in Tab. 1 are mainly due to their significant syntactic differences from other languages, making them harder for the model to learn. Expanding the dataset for these languages could improve the model's performance by providing more representative data.

Better Replicated Performance: Compared to the original paper, our replicated ML-CLAP model achieves significant improvements across all metrics, mainly due to differences in data quality. Compared to the SONAR-translated text used by baseline, the multilingual text we translated with LLM is of higher quality, which in turn can improve the retrieval performance of the model.

5.6 Evaluation Result of Consistency

5.6.1 Analysis of Embedding Space Consistency

The results of the consistency metrics embedding space gap $\vec{\triangle}_{gap,k}$ and average embedding distance $\vec{\triangle}_{dis,k}$ are shown in Tab. 2. In addition, we give a visualization of the embedding space in Appendix A and case analysis in Appendix B to further illustrate the effectiveness of ATRI in solving the consistency problem.

Smaller values of $\vec{\triangle}_{gap,k}$ and $\vec{\triangle}_{dis,k}$ indicate better alignment of a language's embedding space with English, leading to more consistent retrieval in the ML-ATR task. Compared to the baseline ML-CLAP, CACL achieves an average reduction of 12.9% in Gap and 4.4% in Dis, while KCL reduces Gap by 19.1% and Dis by 14.3%, demonstrating improved cross-language retrieval consistency.

Table 2: Results of spatial differences in the embedding of other languages and English

Scheme	Lang	AudioCaps		Clotho	
		E2T		E2T	
		Gap	Dis	Gap	Dis
ML-CLAP	fra	0.199	0.094	0.120	0.301
	deu	0.210	0.370	0.124	0.289
	spa	0.147	0.290	0.117	0.284
	nld	0.204	0.346	0.121	0.274
	cat	0.151	0.357	0.121	0.307
	jpn	0.237	0.445	0.123	0.353
	zho	0.181	0.414	0.177	0.323
	avg	0.189	0.330	0.129	0.304
our CACL	fra	0.160	0.281	0.112	0.288
	deu	0.194	0.334	0.103	0.261
	spa	0.090	0.210	0.099	0.265
	nld	0.172	0.325	0.106	0.255
	cat	0.104	0.252	0.108	0.280
	jpn	0.217	0.402	0.122	0.359
	zho	0.192	0.381	0.159	0.352
	avg	0.161	0.312	0.115	0.294
our KCL	fra	**0.145**	**0.274**	**0.094**	**0.261**
	deu	**0.155**	**0.290**	**0.084**	**0.231**
	spa	**0.081**	**0.192**	**0.084**	**0.230**
	nld	**0.148**	**0.285**	**0.072**	**0.204**
	cat	**0.092**	**0.245**	**0.087**	**0.243**
	jpn	**0.188**	**0.356**	**0.106**	**0.310**
	zho	**0.181**	**0.379**	**0.123**	**0.312**
	avg	**0.141**	**0.288**	**0.092**	**0.255**

Table 3: Results of Mean Rank Variance

Scheme	AudioCaps	Clotho
	MRV	MRV
ML-CLAP	10.38	347.34
CACL	8.71	274.87
KCL	**7.52**	**263.15**

5.6.2 Analysis of Rank Consistency

MRV quantifies the consistency of search rankings across languages, with lower values indicating more consistent results across languages. Unlike metrics based on embedding space, MRV offers a more direct assessment of model consistency in the ML-ATR task. As shown in Tab. 3, KCL achieves the lowest MRV, representing a 25.9% reduction compared to ML-CLAP, while CACL achieves a 22.3% reduction. This effectively shows that the inconsistency issue can be effectively mitigated by reducing the data distribution error.

We note that the MRV metrics under the Audiocaps dataset are significantly lower than Clotho's. This is due to the fact that the Clotho dataset is much noisier and more difficult to get consistent retrieval results across languages.

Table 4: Evaluation results in GPU memory overheads and time overheads

Scheme	AudioCaps		Clotho	
	GMO(MB)	TO(s)	GMO(MB)	TO(s)
ML-CLAP	22172	3349	30912	1592
our CACL	26788	3745	31528	1714
our KCL	68256	4209	79480	1884

5.7 Evaluation Results about Training Overhead

Tab.4 summarises the GPU memory overhead (GMO) and time overhead (TO) during training for three scenarios: ML-CLAP, CACL, and KCL. KCL training requires simultaneous input of text in eight languages, which significantly increases overhead, resulting in a higher GMO of about 2.8 times and a 27% increase in TO compared to ML-CLAP. In contrast, CACL inputs just twice as much text as ML-CLAP, resulting in a modest increase of about 10% in both overheads. This makes CACL more suitable for scenarios that prioritize lower training overheads, while KCL is more suitable for applications that emphasize retrieval performance.

6 Conlusion

In this paper, we address the inconsistencies in ranking results observed in existing ML-ATR schemes. Through an analysis of modality alignment errors and weighting errors, we identify data distribution errors during training as a key factor impacting cross-lingual modality alignment, ultimately leading to retrieval inconsistencies. To address this, we propose two training strategies: KCL and CACL, designed for scenarios prioritizing retrieval performance and training overhead, respectively. Experimental results demonstrate that both CACL and KCL effectively enhance retrieval performance and consistency in ML-ATR tasks. Notably, KCL achieves state-of-the-art results in both English-oriented monolingual ATR and ML-ATR tasks. Furthermore, the proposed approach of mitigating data distribution errors to reduce inconsistencies holds potential for broader applications, including multilingual modality alignment in image and video modalities.

Limitation

We acknowledge that the upper bound on the weighting error in Eq. (6) is heuristically proven for the SGD optimizer. For more complex optimizers such as Adam, giving a direct upper bound

on the weighting error is difficult. But we provide proof of momentum error upper bound for Adam in the Appendix C.1, and show that our idea of reducing the data distribution error is still feasible under the Adam optimizer by showing that momentum error leads to weight error.

References

Louis Béthune, Thomas Masséna, Thibaut Boissin, Yannick Prudent, Corentin Friedrich, Franck Mamalet, Aurélien Bellet, Mathieu Serrurier, and David Vigouroux. 2023. Dp-sgd without clipping: The lipschitz neural network way. *arXiv preprint arXiv:2305.16202*.

Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. 2024. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*.

Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. 2020. Debiased contrastive learning. *Advances in neural information processing systems*, 33:8765–8775.

Matéo Cousin, Etienne Labbé, and Thomas Pellegrini. 2023. Multilingual audio captioning using machine translated data. *arXiv preprint arXiv:2309.07615*.

Heinrich Dinkel, Yongqing Wang, Zhiyong Yan, Junbo Zhang, and Yujun Wang. 2024. Ced: Consistent ensemble distillation for audio tagging. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 291–295. IEEE.

Konstantinos Drossos, Samuel Lipping, and Tuomas Virtanen. 2020. Clotho: An audio captioning dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 736–740. IEEE.

Paul-Ambroise Duquenne, Holger Schwenk, and Benoît Sagot. 2023. Sonar: sentence-level multimodal and language-agnostic representations. *arXiv e-prints*, pages arXiv–2308.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. 2023. Clap learning audio concepts from natural language supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Benjamin Elizalde, Shuayb Zarar, and Bhiksha Raj. 2019. Cross modal audio search and retrieval with joint embeddings based on text and audio. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4095–4099. IEEE.

Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel. 2022. Audioclip: Extending clip to image, text and audio. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 976–980. IEEE.

Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. 2019. Audiocaps: Generating captions for audios in the wild. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 119–132.

A Sophia Koepke, Andreea-Maria Oncescu, João F Henriques, Zeynep Akata, and Samuel Albanie. 2022. Audio retrieval with natural language queries: A benchmark study. *IEEE Transactions on Multimedia*, 25:2675–2685.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR.

Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. 2022. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625.

Siyu Lou, Xuenan Xu, Mengyue Wu, and Kai Yu. 2022. Audio-text retrieval in context. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4793–4797. IEEE.

Xinhao Mei, Xubo Liu, Jianyuan Sun, Mark D Plumbley, and Wenwu Wang. 2022. On metric learning for audio-text cross-modal retrieval. *arXiv preprint arXiv:2203.15537*.

Zhijie Nie, Richong Zhang, Zhangchi Feng, Hailang Huang, and Xudong Liu. 2024. Improving the consistency in cross-lingual cross-modal retrieval with 1-to-k contrastive learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2272–2283.

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

Jörg Tiedemann and Santhosh Thottingal. 2020. Opus-mt–building open translation services for the world. In *Proceedings of the 22nd annual conference of the European Association for Machine Translation*, pages 479–480.

Ho-Hsiang Wu, Prem Seetharaman, Kundan Kumar, and Juan Pablo Bello. 2022. Wav2clip: Learning robust

audio representations from clip. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4563–4567. IEEE.

Yuxin Xie, Zhihong Zhu, Xianwei Zhuang, Liming Liang, Zhichang Wang, and Yuexian Zou. 2024. Gpa: global and prototype alignment for audio-text retrieval. In *Proc. Interspeech 2024*, pages 5078–5082.

Yifei Xin, Xuxin Cheng, Zhihong Zhu, Xusheng Yang, and Yuexian Zou. 2024. Diffatr: Diffusion-based generative modeling for audio-text retrieval. *arXiv preprint arXiv:2409.10025*.

Zhiyong Yan, Heinrich Dinkel, Yongqing Wang, Jizhong Liu, Junbo Zhang, Yujun Wang, and Bin Wang. 2024. Bridging language gaps in audio-text retrieval. *arXiv preprint arXiv:2406.07012*.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. 2022. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*.

Ge Zhu, Jordan Darefsky, and Zhiyao Duan. 2024. Cacophony: An improved contrastive audio-text model. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Martin Zinkevich, Markus Weimer, Lihong Li, and Alex Smola. 2010. Parallelized stochastic gradient descent. *Advances in neural information processing systems*, 23.

A Embedding Space Visualisation

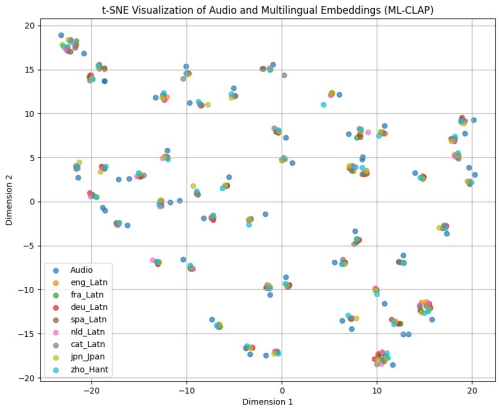

Figure 3: **Visualisation of the ML-CLAP embedding space**.

To further compare the multilingual embedding alignment effects of KCL and ML-CLAP, we randomly select 50 audio-text pairs from AudioCaps. We visualize the embedding spaces of ML-CLAP and KCL after TSNE dimensionality reduction, as

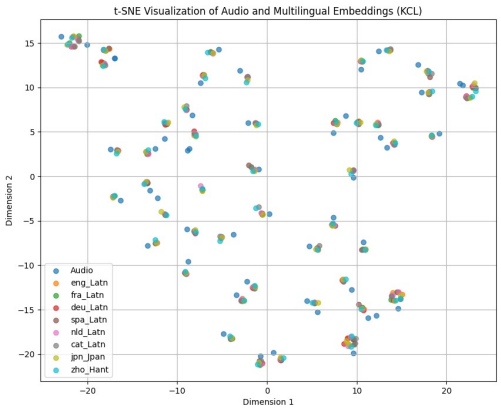

Figure 4: **Visualisation of the KCL embedding space**.

shown in Fig. 3 and Fig. 4, respectively. In KCL, text embeddings with the same semantics across different languages are more compactly clustered compared to ML-CLAP. This indicates that KCL achieves better alignment of multilingual text embeddings, resulting in more consistent retrieval performance across languages.

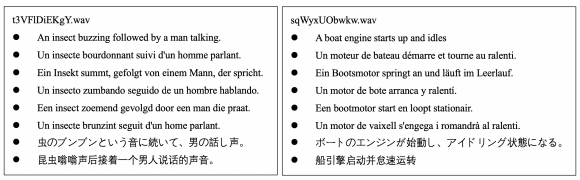

Figure 5: **Two cases selected for analysis**. The audio text pair on the left is Case 1 and the one on the right is Case 2.

Table 5: Retrieval similarity ranking of selected cases

Lang	ML-CLAP		KCL	
	Case 1	Case 2	Case 1	Case 2
eng	5	2	1	0
fra	4	4	1	0
deu	5	4	0	0
spa	2	7	0	0
nld	4	12	0	2
cat	5	12	0	4
jpn	8	0	0	0
zho	13	3	2	0

B Case Analysis

In this section, we select two audio text pairs on the AudioCaps test set shown in Fig. 5 for our case analysis. Tab. 5 shows the results of the retrieval rankings of audio-text pairs under the KCL and ML-CLAP schemes. the retrieval rankings of KCL are generally ahead of those of ML-CLAP,

and the difference in retrieval rankings across languages is much smaller, effectively mitigating the inconsistency issue.

C Proof of Weight Error Upper Bound

We analyze the upper bound on the weighting error heuristically based on the stochastic gradient descent (SGD) optimization algorithm. The following is a detailed theoretical proof of the upper bound on the weighting error in Eq. (6).

Proof. Based on the definition of the SGD optimization algorithm, we have:

$$\begin{aligned}\mathbf{w}_{eT} &= \mathbf{w}_{eT-1} - \eta \sum_{(a,t)} p(a,t) \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)],\\ \mathbf{w}'_{eT} &= \mathbf{w}'_{eT-1} - \eta \sum_{(a,t)} p'_e(a,t) \nabla_{\mathbf{w}'_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)].\end{aligned} \tag{17}$$

$$\begin{aligned}
&||\mathbf{w}_{eT} - \mathbf{w}'_{eT}||\\
=&||\mathbf{w}_{eT-1} - \eta \sum_{(a,t)} p(a,t) \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]\\
&- \mathbf{w}'_{eT-1} + \eta \sum_{(a,t)} p'_e(a,t) \nabla_{\mathbf{w}'_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]||\\
\leq^1&||\mathbf{w}_{eT-1} - \mathbf{w}'_{eT-1}||\\
&+ \eta || \sum_{(a,t)} p'_e(a,t) \nabla_{\mathbf{w}'_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]\\
&- \sum_{(a,t)} p(a,t) \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]||\\
=&||\mathbf{w}_{eT-1} - \mathbf{w}'_{eT-1}||+\\
&+ \eta || \sum_{(a,t)} p'_e(a,t) \nabla_{\mathbf{w}'_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]\\
&- \sum_{(a,t)} p'_e(a,t) \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]\\
&+ \sum_{(a,t)} p'_e(a,t) \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]\\
&- \sum_{(a,t)} p(a,t) \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]||\\
\leq^2&||\mathbf{w}_{eT-1} - \mathbf{w}'_{eT-1}||\\
&+ \eta || \sum_{(a,t)} p'_e(a,t) (\nabla_{\mathbf{w}'_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]\\
&- \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)])||\\
&+ \eta || \sum_{(a,t)} (p'_e(a,t) - p(a,t)) \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]||\\
\leq^3&(1 + \eta \sum_{(a,t)} p'_e(a,t) \lambda_{(a,t)}) ||\mathbf{w}_{eT-1} - \mathbf{w}'_{eT-1}||\\
&+ \eta g_{max}(\mathbf{w}_{eT-1}) \sum_{(a,t)} ||p'_e(a,t) - p(a,t)||.
\end{aligned} \tag{18}$$

The inequality 1 and 2 hold because the Triangle Inequality $|a+b| \leq |a| + |b|$. The inequality 3 holds because

$$g_{max}(\mathbf{w}_{eT-1}) = \max_{(a,t)} ||\nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]||, \tag{19}$$

and we assume that $\nabla_{\mathbf{w}'_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]$ and $\nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)]$ are $\lambda_{(a,t)}$-Lipschitz, Gradient trimming can be used in the code implementation to a certain extent to reduce the gradient change in the training process, indirectly reduce the excessive growth of Lipschitz constant, as far as possible to meet the Lipschitz continuity condition.

Based on Eq. (18), let

$$a = (1 + \eta \sum_{(a,t)} p'_e(a,t) \lambda_{(a,t)}), \tag{20}$$

we have

$$\begin{aligned}
&||\mathbf{w}_{eT} - \mathbf{w}'_{eT}||\\
\leq&a||\mathbf{w}_{eT-1} - \mathbf{w}'_{eT-1}||\\
&+ \eta g_{max}(\mathbf{w}_{eT-1}) \sum_{(a,t)} ||p'_e(a,t) - p(a,t)||\\
\leq&a^2||\mathbf{w}_{eT-2} - \mathbf{w}'_{eT-2}||\\
&+ \eta \sum_{(a,t)} ||p'_e(a,t) - p(a,t)||\\
&(g_{max}(\mathbf{w}_{eT-1}) + a g_{max}(\mathbf{w}_{eT-2}))\\
\leq&a^T||\mathbf{w}_{(e-1)T} - \mathbf{w}'_{(e-1)T}||\\
&+ \eta \sum_{(a,t)} ||p'_e(a,t) - p(a,t)||(\sum_{j=0}^{T-1} a^j g_{max}(\mathbf{w}_{eT-1-j}))).
\end{aligned} \tag{21}$$

Thus Eq. (6) is proved successful.

C.1 Migrating to Adam Optimizer

We first give the parameter update computation procedure for the Adam optimizer:

$$g = \sum_{(a,t)} p(a,t) \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\ p(a,t)] \tag{22}$$

$$m_{eT} = \beta_1 m_{eT-1} + (1 - \beta_1) g \tag{23}$$

$$v_{eT} = \beta_2 v_{eT-1} + (1 - \beta_2) g \circ g \tag{24}$$

$$\hat{m}_{eT} = \frac{m_{eT}}{1 - \beta_1^{eT}} \tag{25}$$

$$\hat{v}_{eT} = \frac{v_{eT}}{1 - \beta_2^{eT}} \tag{26}$$

$$\mathbf{w}_{eT} = \mathbf{w}_{eT-1} - \frac{\eta}{\sqrt{\hat{v}_{eT}}} \hat{m}_{eT}. \tag{27}$$

m_{eT} is the first-order momentum and v_{eT} is the second-order momentum.

We illustrate that the data distribution error also causes weight error in the Adam optimizer by analyzing momentum. The error upper bound of the first-order momentum m_{eT} can be inferred as follows:

$$
\begin{aligned}
&||m_{eT} - m'_{eT}|| \\
=&||\beta_1 m_{eT-1} - \beta_1 m'_{eT-1} \\
&- (1-\beta_1) \sum_{(a,t)} p(a,t) \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\, p(a,t)] \\
&+ (1-\beta_1) \sum_{(a,t)} p'_e(a,t) \nabla_{\mathbf{w}'_{eT-1}} \mathbb{E}_{(a,t)}[log\, p'_e(a,t)])|| \\
\leq&||\beta_1 m_{eT-1} - \beta_1 m'_{eT-1}|| \\
&+ (1-\beta_1)|| \sum_{(a,t)} p(a,t) \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\, p(a,t)] \\
&- \sum_{(a,t)} p'_e(a,t) \nabla_{\mathbf{w}'_{eT-1}} \mathbb{E}_{(a,t)}[log\, p'_e(a,t)])|| \\
\leq&||\beta_1 m_{eT-1} - \beta_1 m'_{eT-1}|| \\
&+ (1-\beta_1)|| \sum_{(a,t)} p'_e(a,t) (\nabla_{\mathbf{w}'_{eT-1}} \mathbb{E}_{(a,t)}[log\, p(a,t)] \\
&- \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\, p(a,t)])|| \\
&+ (1-\beta_1)|| \sum_{(a,t)} (p'_e(a,t) - p(a,t)) \nabla_{\mathbf{w}_{eT-1}} \mathbb{E}_{(a,t)}[log\, p(a,t)]|| \\
\leq&(1 + (1-\beta_1) \sum_{(a,t)} p'_e(a,t) \lambda_{(a,t)})||\mathbf{w}_{eT-1} - \mathbf{w}'_{eT-1}|| \\
&+ (1-\beta_1) g_{max}(\mathbf{w}_{eT-1}) \sum_{(a,t)} ||p'_e(a,t) - p(a,t)||.
\end{aligned}
\tag{28}
$$

Eq. (28) shows that data distribution error still influences the upper bound on the first-order momentum error in the Adam optimizer. Similarly, the second-order momentum error is also affected by this error. These momentum errors accumulate in the weight errors, which makes our theoretical error upper bounds applicable under the Adam optimizer as well.