International Journal of Computer Vision (2024) 132:3656–3677
https://doi.org/10.1007/s11263-024-02064-0

Learning with Noisy Correspondence

Zhenyu Huang[1] · Peng Hu[1] · Guocheng Niu[2] · Xinyan Xiao[2] · Jiancheng Lv[1] · Xi Peng[1]

Received: 17 March 2023 / Accepted: 18 March 2024 / Published online: 13 April 2024

Abstract
This paper studies a new learning paradigm for noisy labels, i.e., noisy correspondence (NC). Unlike the well-studied noisy labels that consider the errors in the category annotation of a sample, the NC refers to the errors in the alignment relationship of two data points. Although such false positive pairs are common especially in the data harvested from the Internet, which however are neglected by most existing works. By taking cross-modal retrieval as a showcase, we propose a method called learning with noisy correspondence (LNC). In brief, the LNC first roughly obtains the clean and noisy subsets from the original data and then rectifies the false positive pairs by using a novel adaptive prediction function. Finally, the LNC adopts a novel triplet loss with soft margins to endow cross-modal retrieval the robustness to the NC. To verify the effectiveness of the proposed LNC, we conduct experiments on six benchmark datasets in image-text and video-text retrieval tasks. Besides the effectiveness of the LNC, the experimental results show the necessity of the explicit solution to the NC faced by not only the standard model training paradigm but also the pre-training and fine-tuning paradigms.

Keywords Noisy labels · Cross-modal retrieval · Multimodal learning · Misalignment

1 Introduction

In machine intelligence, the correspondence (alignment relationship) between two data points plays a crucial role in various tasks and applications including cross-modal retrieval (Xu et al., 2017; Yang et al., 2017; Deng et al., 2018; Lee et al., 2018), visual question answering (Zhao et al., 2017; Wu et al., 2017b), visual caption (Anderson et al., 2018; Li et al., 2019; Wu et al., 2017a), object re-identification (Zheng et al., 2012; Ye et al., 2021), and so on.

In practice, it is expensive even impossible to collect a large number of data pairs that are well aligned. In fact, it is common that some negative pairs are wrongly treated as positive, especially when more and more works are trying to leverage the data pairs harvested from the Internet. For example, Conceptual Captions (CC) (Sharma et al., 2018) uses the web images and the associated Alt-text HTML attributes to form data pairs. Similarly, HowTo100M (Miech et al., 2019) collects billions of video-caption pairs by treating the associated subtitles of the video as the descriptive captions. However, such easily accessed data from the Internet inevitably contain mismatched data pairs (i.e., false positive pairs), even with various rigorous filtering rules. According to the expert evaluations (Sharma et al., 2018; Miech et al., 2019), there are still about 3–20% mismatched image-text pairs in the CC and about 50% mismatched video-caption pairs in HowTo100M. Although several works have realized the existence of the noisy pairs, there are few studies have been conducted to explore how to endow neural networks with robustness to such noise.

To the best of our knowledge, this work is one of the first studies on the noisy correspondence (NC) which is remarkably different from the vanilla noisy labels. For better clarity,

Communicated by Gang Hua.

✉ Xi Peng
pengx.gm@gmail.com

Zhenyu Huang
zyhuang.gm@gmail.com

Peng Hu
penghu.ml@gmail.com

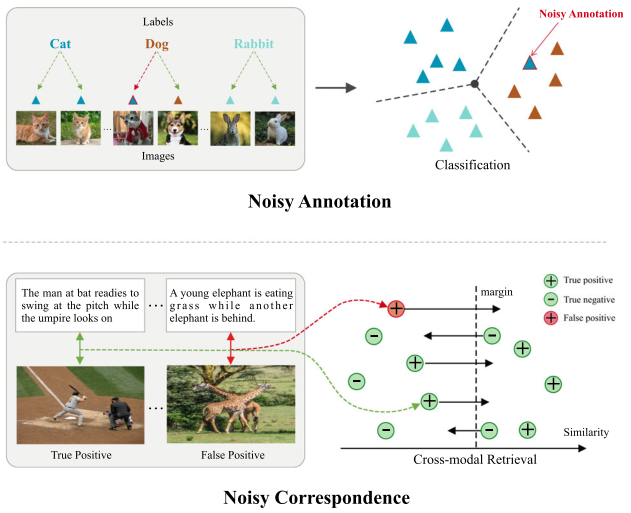
Guocheng Niu
niuguocheng@baidu.com

Xinyan Xiao
xiaoxinyan@baidu.com

Jiancheng Lv
lvjiancheng@scu.edu.cn

[1] College of Computer Science, Sichuan University, Chengdu, China

[2] Baidu Inc., Beijing, China



Noisy Annotation

Noisy Correspondence

Fig. 1 Noisy Annotation vs. Noisy Correspondence. We study a new paradigm for noisy labels in multimodal learning, i.e., noisy correspondence. As shown in the figure, the noisy and clean samples are highlighted in red and green colors, respectively (Color figure online)

we denote the vanilla noisy labels by noisy annotations (NA) in the following as it usually focuses on the category-level annotation errors. Specifically, on the one hand, the correspondence denotes the matching relationship between two given data samples, whereas the well-studied annotation indicates the category for a given data point. On the other hand, the difference lies in these two types of labels lead to different applications, *e.g.*, cross-modal instance retrieval usually aims to build the mapping between two points using the correspondence, whereas the classification task often build the mapping between the point and the accompanied annotation. For a visual understanding of the difference between the two learning paradigms, Fig. 1 presents some examples. As shown, a typical example of noisy annotation is that "Dog" is wrongly assigned to the cat image. Differently, an example of noisy correspondence is that "A young elephant is eating grass while another elephant is behind." is matched to the image of two giraffes eating leaves but treated as positive.

To study the influence of noisy correspondence, we take a typical correspondence-based application as an evaluation showcase, i.e., cross-modal retrieval which leverages the correspondence between cross-modal data to retrieve the related sample from another modality. In such an evaluation scenario, we propose a novel method for achieving robust cross-modal retrieval, termed Learning with Noisy Correspondence (LNC). In brief, the LNC consists of three stages. In the first stage, it warmups two individual models with a momentum regularization and randomly noise abandon technique that are used to avoid noise over-fitting. Then, the LNC divides the given data into two subsets, i.e., the "noisy" and "clean" subsets by leveraging the per-sample loss difference. In the second stage, the LNC designs a prediction module for correspondence refinement to further identify the false and true positives from the clean and noisy subsets, respectively. In the last stage, the LNC adopts a novel matching loss by recasting the rectified correspondence as the soft margins of triplet loss to achieve robust cross-modal retrieval.

The major contributions of our work are summarized as follows:

- This paper shows a new learning paradigm termed noisy correspondence. The NC could be regarded as a member of the noisy label family but with significant differences. In brief, the NC refers to the matching errors in data pairs rather than the category annotation errors of a sample. As far as we know, this work is one of the first studies on this untouched problem.
- In the scenario of cross-modal retrieval, we propose a novel method that could adaptively detect the false positive pairs and rectify them. Besides the aforementioned novelty in the learning setting, the major technical novelty of LNC is that the soft rectified correspondence is recast as the soft margins of the matching loss for training so that robust cross-modal retrieval is achieved.
- Extensive experiments on six benchmark datasets in terms of two cross-modal retrieval tasks, i.e., image-text and video-text retrieval, demonstrate the effectiveness of LNC in handling the synthesized and real-world noisy correspondence. Furthermore, the comparisons and analysis with big models pre-trained on massive data demonstrate the necessity of the NC-specified solution, especially in the era of foundation models.

2 Related Work

In this section, we review the recent progress in noisy annotations and cross-modal retrieval.

2.1 Noisy Annotations

In recent years, huge success has been achieved in handling the noisy annotations (Liu & Tao, 2015; Han et al., 2018; Feng & An, 2019; Song et al., 2020; Bai et al., 2021). In general, most of the existing works resort to designing robust network architecture, adding a regularization to the loss, weighting different loss terms, or identifying the clean from noisy samples. In this paper, we mainly review the last two types of methods which are more related to this study.

As one typical method of loss re-weighting, Patrini et al. (2017) proposes modeling the noise process using a label transition matrix, and thus the underlying noise transition pattern is discovered for correcting the loss. Reed et al. (2014) eliminates the negative impact from the noisy annotations with a well-designed bootstrapping loss. Differently, the clean sample identifying methods aims to identify the most probable clean samples from the noisy data for training by using the memorization effect of neural networks (Arpit et al., 2017), i.e., neural networks are apt to fit the simple patterns first and then gradually fit to the noisy samples. By using such a working mechanism, Arazo et al. (2019) proposes treating the small-loss samples as the clean ones. Moreover, to avoid the self-selection bias in the single network, Co-teaching (Han et al., 2018; Yu et al., 2019) leverage two individual networks to filter out the noises in an alternative fashion. Recently, DivideMix (Li et al., 2020) adopts the MixMatch technique (Berthelot et al., 2019) to further boost the classification performance by treating the clean and noisy samples as the labeled and unlabeled data in a semi-supervised learning framework.

Different from the above studies, this work considers another different problem, i.e., some unrelated data pairs are wrongly treated as positive. Moreover, our method is different from the above methods in the technical aspect. Specifically, it is impractical and even impossible to overcome the noisy correspondence challenge by directly using these existing noisy annotation algorithms due to the following two reasons. On the one hand, most existing noisy annotations works focuses on the NA problem and take the classification as the evaluation scenario. These method propose to rectify the noisy annotations by using the prediction of a classifier. As the retrieval models usually only output the similarity of given pairs, there are many challenges to rectify the noisy correspondence with the similarity, *e.g.*, how to adaptively distinct the true positives from the data by using the similarity only. On the other hand, even if the noisy correspondence could be well rectified, almost all existing matching models are incompatible with the soft rectified correspondence which are real-valued instead of binary (positive or negative pair). To overcome these task-specific challenges, we propose a prediction function that could adaptively predict and rectify the correspondence of the given pairs. Moreover, to leverage the rectified soft correspondence, the LNC recasts the soft correspondence into the soft margins of the matching loss function in an elegant manner.

2.2 Cross-Modal Retrieval

Cross-modal retrieval aims to project different modalities into a common space wherein the cross-modal data are aligned and comparable. In other words, the cross-modal samples are adjacent if and only if they are similar in semantics, and vice versa. In general, there are two types of cross-modal retrieval methods: (i) Coarse-grained Cross-modal Retrieval, which leverages multiple networks to represent different modalities and jointly learns the global feature embedding (Kiros et al., 2014; Wang et al., 2016; Faghri et al., 2017; Torabi et al., 2016). To further improve the retrieval performance, VSE++ (Faghri et al., 2017) employs the hard negatives mining techniques to enhance the discrimination performance in the triplet loss. (ii) Fine-grained Cross-modal Retrieval, which aims to preserve the fine-grained semantic similarity for cross-modal retrieval (Lee et al., 2018; Li et al., 2019a; Diao et al., 2021). For example, SCAN (Lee et al., 2018) learns the latent semantic correspondence between image regions and words at the feature level using the bottom-up attention (Anderson et al., 2018) and GRU respectively. CT-SAN (Yu et al., 2017) detects the key concept words from the videos using a semantic attention mechanism. JSFusion (Yu et al., 2018) leverages the hierarchical attention mechanisms to learn the common representations in a bottom-up manner. SGRAF (Diao et al., 2021) proposes to reason the similarity on the constructed similarity graph while eliminating the meaningless cross-modal correspondence using an attention filtration technique.

Although these methods have achieved promising results, they highly rely on the well-matched cross-modal pairs that are extremely expensive and time-consuming to collect. In real-world applications, a huge number of data pairs are harvested from the Internet (Sharma et al., 2018; Jia et al., 2021) and some of them are mismatched. Shekhar et al. (2017) introduces a FOIL-COCO dataset by incorporating one mistake word in the caption (foil words), to assess the robustness of vision-language models. The investigation reveals that these models exhibit a deficiency in robustness when confronted with the foil dataset. Some works (Sharma et al., 2018; Jia et al., 2021) have realized the existence of these mismatched pairs, however they neglect the corresponding influence and believe that the robustness could be achieved with more data. In other words, there is no work to explicitly study this problem and this paper shows the necessity of a noisy correspondence oriented algorithm.

3 Learning with Noisy Correspondence

In this section, we elaborate on the implementation details of our method LNC. For ease of presentation, we first formulate the noisy correspondence problem in Sect. 3.1. Then we elaborate on the co-divide module which is designed to distinguish the clean samples from the noisy data in Sect.3.2, along with a novel warmup step including a momentum regularization and a randomly noise abandon technique. In Sect. 3.3, we elaborate on how to use the co-rectify module to rectify the relationships of the false positive pairs. Then, in Sect. 3.4, we introduce how to integrate the above co-divide and co-rectify modules so that the cross-modal retrieval model could be endowed with the robustness against the NC. At last, we will discuss the difference between this work and the preliminary version NCR (Huang et al., 2021).

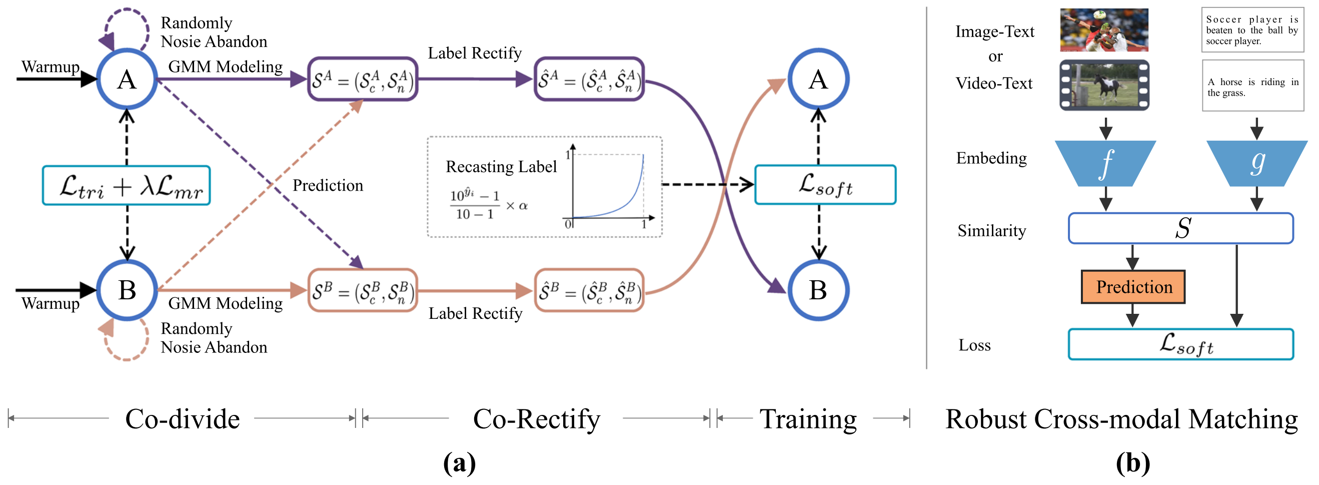

Randomly
Nosie Abandon
Warmup
A
GMM Modeling
$\mathcal{S}^A = (\mathcal{S}^A_c, \mathcal{S}^A_n)$
Label Rectify
$\hat{\mathcal{S}}^A = (\hat{\mathcal{S}}^A_c, \hat{\mathcal{S}}^A_n)$
A
$\mathcal{L}_{tri} + \lambda\mathcal{L}_{mr}$
Prediction
Recasting Label
$\frac{10^{\hat{y}_i} - 1}{10 - 1} \times \alpha$
1
0
1
$\mathcal{L}_{soft}$
Warmup
B
GMM Modeling
$\mathcal{S}^B = (\mathcal{S}^B_c, \mathcal{S}^B_n)$
Label Rectify
$\hat{\mathcal{S}}^B = (\hat{\mathcal{S}}^B_c, \hat{\mathcal{S}}^B_n)$
B
Randomly
Nosie Abandon
Image-Text
or
Video-Text
Soccer player is beaten to the ball by soccer player.
A horse is riding in the grass.
Embeding
f
g
Similarity
S
Prediction
Loss
$\mathcal{L}_{soft}$
Co-divide
Co-Rectify
Training
Robust Cross-modal Matching
(a)
(b)

Fig. 2 Overview of LNC. **a** LNC training pipeline. LNC leverages two individual cross-modal retrieval networks (A, B) similar to co-teaching. In brief, a warmup step is processed on the noisy data by combing the triplet loss $\mathcal{L}_w$ and the momentum regularization. Then, at each epoch, LNC obtains relatively clean and noisy subsets by using the loss difference of data w.r.t. network A and B, i.e., $\mathcal{S}^A = (\mathcal{S}^A_c, \mathcal{S}^A_n)$ and $\mathcal{S}^B = (\mathcal{S}^B_c, \mathcal{S}^B_n)$, where $\mathcal{S}^A_c$ and $\mathcal{S}^B_c$ denote the clean subsets obtained by using network A and B, and $\mathcal{S}^A_n$ and $\mathcal{S}^B_n$ denote the noisy subsets obtained by using network A and B. Then, LNC rectify the correspondence of $\{\mathcal{S}^A, \mathcal{S}^B\}$ by using the designed prediction function, yielding the rectified data $\{\hat{\mathcal{S}^A}, \hat{\mathcal{S}^B}\}$. Finally, LNC train the network B and A by using $\hat{\mathcal{S}^A}$ and $\hat{\mathcal{S}^B}$ in a swapping manner. **b** Robust cross-modal retrieval model. We first use the modal-specific networks f and g to embed both visual and textual data into a common space. Then, we compute the cosine similarity $S(\cdot)$ on the visual and textual embeddings, i.e., $f(\cdot)$ and $g(\cdot)$. Finally, we adopts a novel matching loss $\mathcal{L}_{soft}$ to achieve robust training

3.1 Problem Formulation

For a given query, cross-modal retrieval methods aim to seek the relevant instance from the gallery across modality, of which the key is to learn a common space wherein the semantic-relevant cross-modal pairs keep adjacent. Without loss of generality, we take image-text retrieval as an example and first introduce the used mathematical notations for clarity. Formally, for the paired image-text training data $\mathcal{D} = \{(\mathbf{v}_i, \mathbf{t}_i, y_i)\}_{i=1}^N$, we use $\mathbf{v}_i$ and $\mathbf{t}_i$ to denote the i-th image and associated text, and $y_i \in \{0, 1\}$ to indicate whether the corresponding pair is matched (positive) or not (negative). In the traditional cross-modal retrieval, it assumes all image-text training pairs are truly matched, i.e., true positive. Differently, we assume that the existence of noisy correspondence will lead to false positive cases, i.e., some samples of $y_i = 0$ are wrongly annotated as $y_i = 1$.

The framework of LCNL is shown in Fig. 2. To achieve robust cross-modal retrieval, LNC adopts a retrieval model that consists of two modal-specific networks f, g and a similarity function S. The two networks f and g are used to compute the visual and textual embeddings of $\mathbf{v}$ and $\mathbf{t}$, respectively. After that, the similarity function S computes the semantic similarity $S(f(\mathbf{v}), g(\mathbf{t}))$ (denoted by $S(\mathbf{v}, \mathbf{t})$ for simplicity in the following) across modalities (Fig. 2).

3.2 Co-Divide

Some pioneer works Arpit et al. (2017) have experimentally shown that neural networks are apt to learn simple samples in the early training period and then gradually fit the noisy ones with further training. Such a so-called memorization effect implies that the clean samples have relatively low losses than the noisy ones in the early training stage. Based on this property, we propose to treat the data pairs with small loss as the clean samples and the pairs with larger loss as the noisy samples similar to (Han et al., 2018; Yu et al., 2019; Arazo et al., 2019; Li et al., 2020). This premise is founded on the assumption that the network optimization process will primarily focus on the clean samples, thereby considering the larger-loss samples as noisy pairs. Specifically, by feeding the noisy data into a given retrieval model (f, g, S), we first obtain the per-sample loss by

$$\ell_{(f,g,S)} = \{\ell_i\}_{i=1}^N = \{\mathcal{L}_{tri}(\mathbf{v}_i, \mathbf{t}_i)\}_{i=1}^N, \tag{1}$$

where $\mathcal{L}_{tri}(\mathbf{v}_i, \mathbf{t}_i)$ is the vanilla triplet loss defined by

$$\begin{aligned} \mathcal{L}_{tri}(\mathbf{v}_i, \mathbf{t}_i)) = & \sum_{\hat{\mathbf{t}}} [\alpha - S(\mathbf{v}_i, \mathbf{t}_i) + S(\mathbf{v}_i, \hat{\mathbf{t}})]_+ \\ & + \sum_{\hat{\mathbf{v}}} [\alpha - S(\mathbf{v}_i, \mathbf{t}_i) + S(\hat{\mathbf{v}}, \mathbf{t}_i)]_+, \end{aligned} \tag{2}$$

where $(\mathbf{v}_i, \mathbf{t}_i)$ is a positive image-text pair, $\alpha > 0$ is a positive margin, and $[x]_+ = \max(0, x)$. Specifically, in the loss function, the first term enforces that the given positive pairs have larger affinity than the negatives by taking $\mathbf{v}$ as queries and all other text $\hat{\mathbf{t}}$ as negatives. Similarly, the second term takes $\mathbf{t}$ as queries and all other images $\hat{\mathbf{v}}$ as negatives.

With the computed per-sample losses, we adopt a two-component Gaussian Mixture Model (GMM) to fit the loss distribution:

$$p(\ell|\theta) = \sum\nolimits_{m=1}^{2} \beta_m \phi(\ell|m), \tag{3}$$

where $\phi(\ell|m)$ and β_m are the probability density and the mixture coefficient of the m-th component in the GMM optimized by the EM algorithm. To distinct the noisy samples from the clean ones, we first compute the posterior probability as the clean confidence of i-th data pair $w_i = p(m|\ell_i) = p(m)p(\ell_i|m)/p(\ell_i)$, where m denotes the component with lower mean (small loss) in GMM. Then, one could easily split the data into clean and noisy partitions by thresholding $\{w_i\}_{i=1}^{N}$. For simplicity, the threshold is fixed to 0.5 in our experiments.

Following Han et al. (2018), to avoid the error accumulation in noisy/clean division from the single network, we adopt the co-teaching paradigm to divide the training data. Specifically, we individually train two networks $A = \{f^A, g^A, S^A\}$ and $B = \{f^B, g^B, S^B\}$ with different initial network parameters and training batches. At each epoch, we model the per-sample loss distribution computed by using the network A or B, and split the original dataset into clean and noisy subsets to train the other network, i.e., co-divide. Notably, the memorization effect works in the initial training period of neural works, hence we use the vanilla triplet loss $\mathcal{L}_{tri}$ to warmup the networks before performing co-divide.

In the warmup period, neural networks would eventually fit the noisy ones with further training, thus degrading the performance of co-divide. To alleviate this noise overfitting issue, we propose a novel momentum regularization (MR) for the warmup training loss and a noise discard technique. To be specific, the MR is designed to prevent the overfitting of noisy pairs by augmenting the loss contribution from the true positives. More specifically, the MR is defined by

$$\mathcal{L}_{mr} = \frac{1}{N} \| P(\mathbf{v}_i, \mathbf{t}_i) - \boldsymbol{\gamma}_i \|^2, \tag{4}$$

where $P(\mathbf{v}_i, \mathbf{t}_i)$ is the model prediction of the given pair $(\mathbf{v}_i, \mathbf{t}_i)$ which will be introduced in the next section. $\boldsymbol{\gamma}_i$ is the i-th value in $\boldsymbol{\gamma}$ which is computed by the past model predictions. In detail, let P_i^{τ} be the model predictions of the i-th pair $(\mathbf{v}_i, \mathbf{t}_i)$ at epoch τ, $\boldsymbol{\gamma}_i^{\tau}$ is computed by:

$$\boldsymbol{\gamma}_i^{\tau} = \beta \boldsymbol{\gamma}_i^{\tau-1} + (1 - \beta) P_i^{\tau}, \tag{5}$$

where β is a momentum factor, $\boldsymbol{\gamma}_i^0 = 1$. To understand why the regularization could avoid the noisy over-fitting, we first review the memorization effect of the vanilla triplet. To be specific, in the early memorization stage, the randomly initialized networks treat the noisy and clean samples equally, then gradually fit the clean samples and then the noisy ones. As a result, after a few optimization steps, the model will fit the clean samples well and the corresponding gradient will approximate to zero, while the noisy pairs tend to dominate the gradient. In such a situation, the model will eventually over-fit to the noisy pairs. Our momentum regularization will penalize the gradients from the noisy samples while enlarging the clean ones even the network is already converged to the clean samples. Han et al. (2019) proposed a similar strategy which leverages both the original labels and the pseudo labels produced in a self-learning manner for model training. Note that even our momentum regularizer share some similar characteristics to Han et al. (2019), they are different in the objective and formulation. First, our regularizer is proposed to avoid noise overfitting in the warmup period by balancing the loss contribution of the clean and noisy samples while the work Han et al. (2019) aims to improve the model performance by optimizing the model with the pseudo labels predicted in a self-learning fashion. Second, the formulations of two works are different. Our regularizer aims to minimize the ℓ_2 loss between the model output and the given target, whereas (Han et al., 2019) adopts a cross-entropy loss function. Third, our regularizer enforces the model prediction closer to the target that is updated by the past model predictions with momentum while (Han et al., 2019) directly uses the model predictions as the target.

With these losses together, our warmup loss is defined as,

$$\mathcal{L}_w = \mathcal{L}_{tri} + \lambda \mathcal{L}_{mr}, \tag{6}$$

where λ is a balance factor. Besides the above momentum regularization, we propose discarding the confident noisy pairs with low confidence w to further avoid noise overfitting during warmup. In detail, the proposed Randomly Noise Abandon (RNA) strategy discards half of the pairs which are detected as noisy one by both two networks (A, B) at the same time, i.e.,

$$\{(\mathbf{v}_j, \mathbf{t}_j) | w_j^A < 0.5, \text{ and } w_j^B < 0.5, \forall j \in N\}, \tag{7}$$

where w_j^A and w_j^B are the clean confidence of j-th pair estimated by network A and B respectively.

3.3 Co-rectify

After obtaining the "clean" subset $\mathcal{S}_c = \{(\mathbf{v}_i^c, \mathbf{t}_i^c, y_i^c, w_i)\}_{i=1}^{N_c}$ and "noisy" subset $\mathcal{S}_n = \{\mathbf{v}_i^n, \mathbf{t}_i^n\}_{i=1}^{N_n}$ by the co-divide module, the co-rectify module is used to further rectify the correspondence to recall the true positive pairs in $\mathcal{S}_n$ and discard the false positives in $\mathcal{S}_c$. In detail, the co-rectify module will rectify the correspondence of $\{\mathcal{S}_c, \mathcal{S}_n\}$ by combining the clean confidence w_i and the predictions from the model k through

$$\begin{cases} \forall(\mathbf{v}_i^c, \mathbf{t}_i^c, y_i^c, w_i^c) \in \mathcal{S}_c, \hat{y}_i^c = w_i y_i^c + (1 - w_i) P^k(\mathbf{v}_i^c, \mathbf{t}_i^c), \\ \forall(\mathbf{v}_i^n, \mathbf{t}_i^n) \in \mathcal{S}_n, \hat{y}_i^n = (P^A(\mathbf{v}_i^n, \mathbf{t}_i^n) + P^B(\mathbf{v}_i^n, \mathbf{t}_i^n))/2, \end{cases} \tag{8}$$

where $k \in \{A, B\}$, $P^A(\mathbf{v}, \mathbf{t})$ and $P^B(\mathbf{v}, \mathbf{t})$ are the predictions from network A and B.

The above co-rectify module is designed to achieve the following goals. In brief, for $\mathcal{S}_c$ whose most pairs are probably true positive, Eq. 8 uses the original correspondence y_i^c and the model prediction $P(\mathbf{v}_i^c, \mathbf{t}_i^c)$ for rectifying the correspondence. For $\mathcal{S}_n$ whose most pairs are probably false positive, Eq. 8 will discard the unreliable correspondence and correct it by combing the predictions from both network A and B, i.e., $P^A(\mathbf{v}_i^n, \mathbf{t}_i^n)$ and $P^B(\mathbf{v}_i^n, \mathbf{t}_i^n)$.

Different from the classification, cross-modal retrieval outputs the similarity for ranking and cannot directly predict the correspondence. Hence, to predict the correspondence using the retrieval model, we propose a novel prediction function $P(\mathbf{v}, \mathbf{t})$ as below:

$$\begin{aligned} s &= S(\mathbf{v}, \mathbf{t}) - \left(\frac{1}{b} \sum_{\hat{\mathbf{t}}} S(\mathbf{v}, \hat{\mathbf{t}}) + \frac{1}{b} \sum_{\hat{\mathbf{v}}} S(\hat{\mathbf{v}}, \mathbf{t}) \right) /2, \\ P(\mathbf{v}, \mathbf{t}) &= \Theta(s)/\tau, \end{aligned} \tag{9}$$

where s is the similarity margin from $(\mathbf{v}, \mathbf{t})$ to the average similarity of all negative pairs in a batch, b denotes the batch-size, $\Theta(s)$ clamps s into $[0, \alpha]$, and τ is the average s of top 10% pairs with largest s in a batch. The value of τ takes the assumption that there are at least 10% clean pairs in the data for anchoring the positive correspondence. Intuitively, we treat the correspondence of the pairs as 1 when s is larger than τ, otherwise $[0, 1)$.

3.4 Robust Cross-Modal Matching

Exiting cross-modal retrieval models assume that the given pairs are either positive or negative ($y \in \{0, 1\}$), which are incompatible with the soft correspondence ($y \in [0, 1]$) obtained by LNC. To leverage the soft correspondence for achieving robust cross-modal retrieval, we propose recasting the rectified correspondence as the soft margins in the matching loss. Mathematically,

$$\begin{aligned} \mathcal{L}_{soft} = & [\hat{\alpha}_i - S(\mathbf{v}_i, \mathbf{t}_i) + S(\mathbf{v}_i, \hat{\mathbf{t}}_h)]_+ \\ & + [\hat{\alpha}_i - S(\mathbf{v}_i, \mathbf{t}_i) + S(\hat{\mathbf{v}}_h, \mathbf{t}_i)]_+, \end{aligned} \tag{10}$$

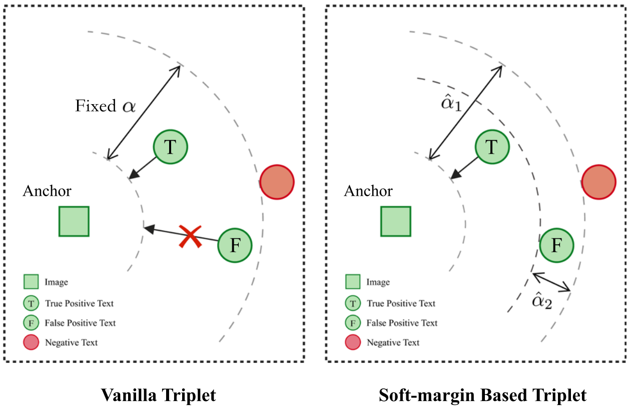


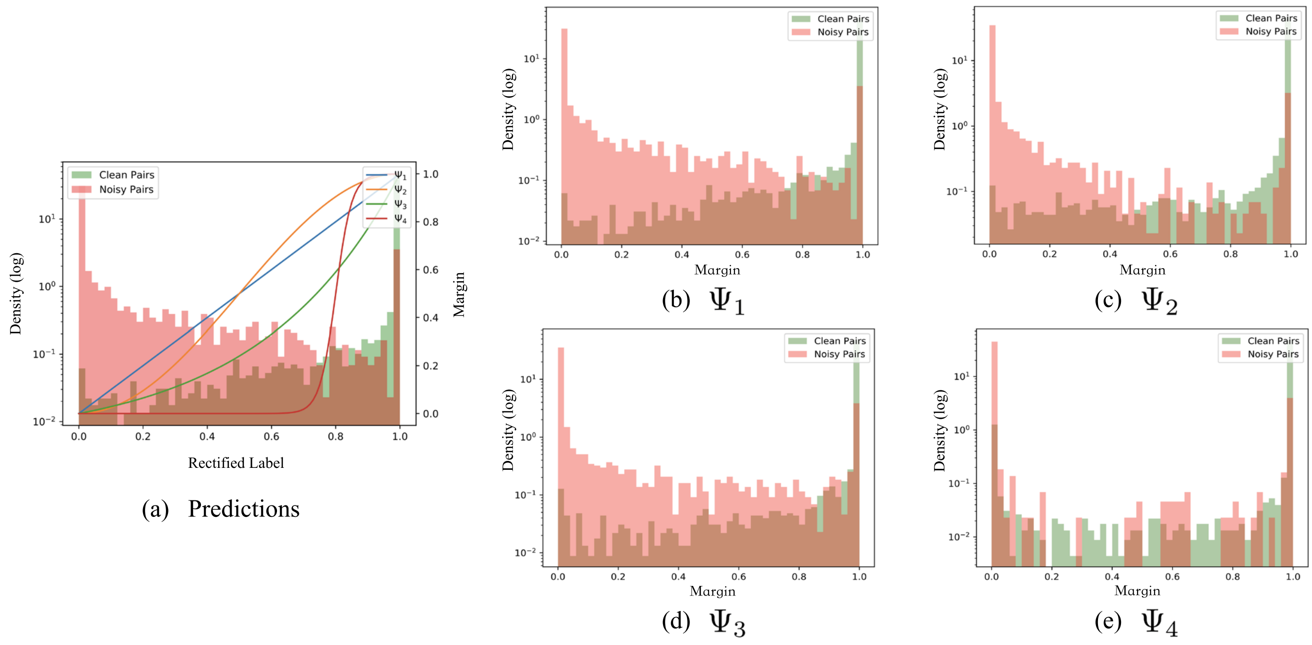
Fig. 3 The Vanilla Triplet versus Our Soft-margin based Triplet. The proposed loss could adaptively assign different margins to the pairs according to the soft rectified correspondence. Specifically, $\mathcal{L}_{soft}$ will pull the true positive sample closer to the anchor until they are closer than the negative with a large margin $\hat{\alpha}_1$. For the positive samples, $\mathcal{L}_{soft}$ will pull it closer to the anchor with a small margin $\hat{\alpha}_2$

where $\hat{\mathbf{v}}_h = \text{argmax}_{\mathbf{v}_j \neq \mathbf{v}_i} S(\mathbf{v}_j, \mathbf{t}_i)$ and $\hat{\mathbf{t}}_h = \text{argmax}_{\mathbf{t}_j \neq \mathbf{t}_i} S(\mathbf{v}_i, \mathbf{t}_j)$ are the most similar negatives to $\mathbf{t}_i$ and $\mathbf{v}_i$ in a batch, respectively. The soft margin $\hat{\alpha}_i$ is adaptively determined by the rectified correspondence $\hat{y}_i$ with a recasting function as $\hat{\alpha}_i = \Psi(\hat{y}_i)$.

To achieve robust retrieval, the function Ψ is designed to preserve the uncertainty of the soft rectified correspondence into margins, i.e., enforcing that the confident positive pair (y close to 1) is closer than the negatives with a large margin, while the confident negative pair (y close to 1) closer than the negatives with a small (even zero) margin. A toy example is shown in Fig. 3. For this purpose, we design four alternative recasting functions for Ψ to transform the rectified correspondence to margins as below,

$$\begin{aligned} \Psi_1 &= \hat{y}_i \times \alpha, \\ \Psi_2 &= \frac{10^{\hat{y}_i} - 1}{10 - 1} \times \alpha, \\ \Psi_3 &= (\sin(\pi \times \hat{y}_i - \pi/2)/2 + 1/2) \times \alpha, \\ \Psi_4 &= \text{sigmoid}((10 + 100 * (div - 0.5)) * (\hat{y}_i - div)) \times \alpha, \end{aligned} \tag{11}$$

where $\hat{y}_i$ is the rectified correspondence, and div is the average value of the rectified correspondence to divide noisy subset and clean subset. All the above recasting functions are designed to assign a large margin to the pairs with higher $\hat{y}_i$ (close to 1), and a small one otherwise. We provide a visualization of the four recasting functions on the Flickr30K with 50% noise in Fig. 4 ($\alpha = 1$ for better visualization), including Ψ_1 (linear), Ψ_2 (exponential), Ψ_3 (sin), and Ψ_4 (sigmoid). As shown, the first function is straightforward by directly multiplying margins by the rectified correspondence.



(a) Predictions

(b) Ψ_1

(c) Ψ_2

(d) Ψ_3

(e) Ψ_4

Fig. 4 A visualization of four correspondence recasting functions on Flickr30K with 50% noise

Such a strategy is too simple and might ignore the distribution diversity of the clean and noisy data. As shown in Fig. 4a, most noisy pairs are located in the left area while the clean ones are in the right area. To leverage such characteristics for boosting performance, we design three non-linear functions as follows: (i) Ψ_2 is designed to restrain the margins of pairs with small $\mathbf{y}$ and enlarge that of pairs with higher $\mathbf{y}$. As shown in Fig. 4c, the margins of noisy pairs (red) are shifted to the left (close to 0). In other words, the margins of noisy pairs are restrained; (ii) Ψ_3 is designed to restrain the confident noisy pairs ($\mathbf{y} < 0.5$) and amplify the confident clean pairs ($\mathbf{y} > 0.5$); (iii) similar to Ψ_3, Ψ_4 aims at restraining the confident noisy pairs and amplifying the confident clean pairs but with a flexible boundary (div) of clean and noisy pairs. The flexible boundary div is adaptively determined by average values of noisy and clean subsets obtained by co-divide. However, a key question is whether the given boundary (0.5 or div) of Ψ_3 and Ψ_4 is matched to the real data distribution. As shown in Fig. 4d–e, the given boundary does not split the data well, demonstrating the difficulty in boundary selection in the real world. A quantitative comparison of the four recasting functions is provided in the experiment.

3.5 Discussions

This paper extends our work NCR (Huang et al., 2021) which was published at NeurIPS 2021 as oral. The extensions are in the following aspects. In detail, (i) To avoid the noise over-fitting, we propose a novel momentum regularization. Specifically, the memorization effect shows that neural networks will first fit the simple samples and eventually over-fit to the noisy samples. Such a noisy over-fitting issue will inevitably affect the performance of our co-divide module. To prevent the memorization of noises, we propose the regularization to amplify the contribution of confident clean samples w.r.t loss and restrain that of noisy samples. (See Eqs. 4–5 in Sect. 3.2) (ii) To further eliminate the negative impact from the noisy pairs, we design a randomly noise abandon strategy to improve the co-divide module. Specifically, the strategy first identifies the confident noisy samples based on the memorization effect and then randomly discards half of the most confident noisy sample identified by both two networks. (See Eq. 7 in Sect. 3.2) (iii) To verify the effectiveness of our method in wider scenarios, we extend the application of NCR from the image-text to the video-text retrieval. Moreover, we verify the effectiveness of LNC on three benchmark video datasets including MSR-VTT, LSMDC, and YouCook2 with various noise ratios. The experimental results demonstrate the effectiveness of the proposed method in handling the NC in video-text data. (See Sect. 4.2) (iv) To reveal the wide influence and existence of noisy correspondence, we investigate the proposed method in the pre-training paradigm. In brief, we provide new discussions and experimental comparisons of LNC with two pre-training settings, i.e., besides the fine-tune period, we examine the influence of NC on the pre-training period. The experimental results demonstrate the superiority of the proposed LNC even compared with the big models pre-trained on billions of data. (See Sect. 4.4) (v) We carry out new experiments to show the noise detection capacity of LNC with various noise ratios (See Sect. 4.5.2)

(vi) We conduct new analysis on four margin recasting functions for achieving robust cross-modal retrieval on Flickr30K with 50% noise. (See Sects. 3.4 and 4.5.2)

4 Experiment

In this section, we present quantitative and qualitative experiment results to verify the effectiveness of the proposed method for robust cross-modal retrieval. In the experiments, we take image-text retrieval and video-text retrieval as the showcase for evaluation. Specifically, we evaluate the proposed method on six benchmark datasets including the image-caption datasets Flickr30K (Young et al., 2014), MS-COCO (Lin et al., 2014), Conceptual Captions (Sharma et al., 2018) and video-caption datasets YouCook2 (Zhou et al., 2018), MSR-VTT (Xu et al., 2016), LSMDC (Rohrbach et al., 2017) to show the effectiveness of our method for handling the possible noisy correspondence. Moreover, to investigate the effectiveness of our method in the pre-training model for handling the NC in fine-tune or pre-training period, we conduct two comparison experiments including (i) NC in fine-tuning: we conduct the comparison to CLIP (Radford et al., 2021) that is pre-trained on millions of image-text pairs on the noisy MS-COCO data and (ii) NC in pre-training: we conduct the comparison to the model pre-trained on the large-scale video dataset HowTo100M (Miech et al., 2019) on the MSR-VTT data. For a comprehensive evaluation, we report R@1, R@5, and R@10 for all experiments as in Lee et al. (2018).

4.1 Image-Text Retrieval

In this section, we conduct experiments on the image-text retrieval task. Image-text retrieval aims to retrieve related images or captions with a given query text or image. We first verify our method on the two benchmark datasets MS-COCO and Flickr30K with synthetic noisy correspondence compared to the state-of-the-art. Then we conduct the experiment on a subset of Conceptual Captions with noisy correspondence from the wild.

4.1.1 Experiment Settings

Implementation details: LNC is a general framework of learning with the noisy correspondence to endow the cross-modal retrieval model with robustness against the NC. For evaluation, we choose the SOTA method SGR (Diao et al., 2021) for extending the robustness to show the effectiveness of LNC. Specifically, we use a fully connected layer (i.e., f) and a bidirectional gated recurrent unit (Schuster & Paliwal, 1997) (i.e., g) to embed the image and sentence into a shared space. Then we compute the affinity S of the given image and text by combining the local and global embeddings with a similarity reasoning method used in Kuang et al. (2019). Following Lee et al. (2018), to extract the local representations for each image, we use the detector of Faster-RCNN (Ren et al., 2015) from Anderson et al. (2018) to extract a 2048-dimensional feature for each top 36 region proposals. We adopt the Adam optimizer (Kingma & Ba, 2014) for network optimization. The training batch-size is fixed to 128. The initial margin α is set to 0.2 through all experiments. For fair evaluation, we keep the network f and g unchanged as in SGR (Diao et al., 2021). In the testing stage, we compute the average similarity by using both the outputs from network A and B to retrieve related samples. As for the correspondence recasting function, we apply the Ψ_2 in all following experiments. More details about the implementation are provided in the supplementary material.

Datasets: We verify our method on three benchmark datasets. Specifically, Flickr30K contains 31,000 images and each has 5 associated textual descriptions. Following the data partition in Lee et al. (2018), there are 29,000 images for training, and the rest for validation and testing (1000 images for each). MS-COCO contains 123,287 images and each has 5 associated textual descriptions. Following the data partition in Lee et al. (2018), there are 113,287 images for training, and the rest for validation and testing (5000 images for each). Since the Flickr30K and MS-COCO are carefully labeled without noise, here we randomly shuffle a specific percentage of the textual descriptions of images in the training data to simulate the noisy correspondence. The shuffle percentage is denoted as the noise ratio. Conceptual Captions (CC) (Sharma et al., 2018) contains 3.3M images with single caption each. Because CC is collected from the Internet without human annotation, there are about 3% $\sim$ 20% mismatched pairs according to the expert evaluation (Sharma et al., 2018). In our experiments, due to the computation resource limitation, we only use a subset dataset CC152K from CC. Specifically, CC152K contains 150,000 image-text pairs for training, 1000 for validation and 1000 for testing. The training samples are selected from the training split in CC, and the validation and testing ones are selected from the validation split in CC.

Baselines: We conduct comparison between the proposed method LNC to the following methods including SCAN (Lee et al., 2018), VSRN (Li et al., 2019a), IMRAM (Chen et al., 2020), SGRAF (Diao et al., 2021) (including two different models SGR and SAF). In Flickr30K and MS-COCO, we conduct experiments with various NC ratios (0%, 20%, and 50%). Moreover, we report the performance of SGR which is only trained on the clean data (denoted by SGR-C). Note that, since the used training data for SGR-C have no noisy correspondence, SGR-C is a strong baseline for showing the effectiveness of LNC. In the non-noise case, we directly report the original results in the references. In the experi-

Table 1 Image-text retrieval results on Flickr30K and MS-COCO 1K

Noise	Method	Flickr30K						MS-COCO					
		Image → Text			Text → Image			Image → Text			Text → Image		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
0%	SCAN	67.4	90.3	95.8	48.6	77.7	85.2	69.2	93.6	97.6	56.0	86.5	93.5
	VSRN	71.3	90.6	96.0	54.7	81.8	88.2	76.2	94.8	98.2	62.8	89.7	95.1
	IMRAM	74.1	93.0	96.6	53.9	79.4	87.2	76.7	95.6	98.5	61.7	89.1	95.0
	SAF	73.7	93.3	96.3	56.1	81.5	88.0	76.1	95.4	98.3	61.8	89.4	95.3
	SGR	75.2	93.3	96.6	56.2	81.0	86.5	78.0	95.8	98.2	61.4	89.3	95.4
	SGRAF	77.8	94.1	97.4	58.5	83.0	88.8	79.6	96.2	98.5	63.2	**90.7**	96.1
	NAAF	**81.9**	**96.1**	**98.3**	**61.0**	**85.3**	**90.6**	**80.5**	**96.5**	**98.8**	**64.1**	**90.7**	**96.5**
	NCR	77.3	94.0	97.5	59.6	84.4	89.9	78.7	95.8	98.5	63.3	90.4	95.8
	LNC	77.4	93.7	97.5	59.6	84.1	89.6	78.7	96.0	98.6	63.9	90.4	95.7
20%	SCAN	59.1	83.4	90.4	36.6	67.0	77.5	66.2	91.0	96.4	45.0	80.2	89.3
	VSRN	58.1	82.6	89.3	40.7	68.7	78.2	25.1	59.0	74.8	17.6	49.0	64.1
	IMRAM	63.0	86.0	91.3	41.4	71.2	80.5	68.6	92.8	97.6	55.7	85.0	91.0
	SAF	51.0	79.3	88.0	38.3	66.5	76.2	67.3	92.5	96.6	53.4	84.5	92.4
	SGR*	62.8	86.2	92.2	44.4	72.3	80.4	67.8	91.7	96.2	52.9	83.5	90.1
	NAAF	65.7	88.7	93.9	53.8	80.6	87.2	69.0	92.9	96.6	58.1	85.3	90.7
	SGR-C	74.7	92.2	95.6	54.8	81.3	88.3	75.4	95.2	97.9	60.1	88.5	94.8
	NCR	75.0	**93.9**	**97.5**	58.3	83.0	89.0	77.7	95.5	98.2	62.5	89.3	95.3
	LNC	**76.3**	93.7	96.9	**58.4**	**83.8**	**89.8**	**78.2**	**95.8**	**98.5**	**62.6**	**89.4**	**95.4**
50%	SCAN	27.7	57.6	68.8	16.2	39.3	49.8	40.8	73.5	84.9	5.4	15.1	21.0
	VSRN	14.3	37.6	50.0	12.1	30.0	39.4	23.5	54.7	69.3	16.0	47.8	65.9
	IMRAM	9.1	26.6	38.2	2.7	8.4	12.7	21.3	60.2	75.9	22.3	52.8	64.3
	SAF	30.3	63.6	75.4	27.9	53.7	65.1	30.4	67.8	82.3	33.5	69.0	82.8
	SGR*	36.9	68.1	80.2	29.3	56.2	67.0	60.6	87.4	93.6	46.0	74.2	79.0
	NAAF	23.3	49.2	60.8	6.8	18.5	26.0	51.0	80.9	88.8	38.1	70.8	78.1
	SGR-C	69.8	90.3	94.8	50.1	77.5	85.2	71.7	94.1	97.7	57.0	86.6	93.7
	NCR	72.9	**93.0**	**96.3**	**54.3**	79.8	86.5	74.6	94.6	97.8	59.1	87.8	94.5
	LNC	**73.0**	92.5	96.1	54.1	**80.5**	**87.2**	**75.8**	**94.9**	**97.9**	**59.8**	**88.1**	**94.6**

The best and second best results are highlight in bold and underline
*Indicates the adding a warmup process to SGR

ments with NC, we report the best results from three times training the baseline models with default settings. Note that, we experimentally found SGR is sensitive to the NC. Thus we add a warmup process to SGR (denoted by SGR*) by training the network with the vanilla matching loss for a few epochs, which shows reasonable results in our experiments.

4.1.2 Results on Flickr30K and MS-COCO

In Table 1, we provide the comparison results of LNC compared to the baselines on the Flickr30K and MS-COCO datasets. As shown, in the non-noise case, LNC achieves competitive retrieval performance even compared to the current SOTA SGRAF. Such a result shows that LNC could achieve SOTA performance in the clean data even though it is proposed to handle the NC. In the noisy cases, LNC achieves the performance and shows a large performance margin to all baselines even the SGR-C that is only trained on clean data. Specifically, compared to SGR-C, LNC improves R@1 (I2T, T2I) by (3.5%, 2.0%) and (3.2%, 4.0%) in Flickr30K with 20% and 50% noise, and (2.8%, 2.5%) and (4.1%, 2.8%) in MS-COCO with 20% and 50% noise. In brief, the experimental results demonstrate the robustness of LNC against the noisy correspondence with various ratios.

4.1.3 Results on CC152K

In Table 2, we provide experimental results on the subset of CC data, i.e., CC152K. As shown, the proposed method LNC consistently outperforms the baselines by a large margin in both image and text retrieval. Specifically, compared to the best baseline (except the preliminary version NCR), LNC

Table 2 Image-text retrieval results on CC152K

Method	Image → Text			Text → Image		
	R@1	R@5	R@10	R@1	R@5	R@10
SCAN	30.5	55.3	65.3	26.9	53.0	64.7
VSRN	32.6	61.3	70.5	32.5	59.4	70.4
IMRAM	33.1	57.6	68.1	29.0	56.8	67.4
SAF	31.7	59.3	68.2	31.9	59.0	67.9
SGR	11.3	29.7	39.6	13.1	30.1	41.6
SGR*	<u>35.0</u>	63.4	<u>73.3</u>	34.9	63.0	72.8
NAAF	32.5	59.1	69.5	33.0	61.0	70.4
NCR	**39.5**	**64.5**	**73.5**	<u>40.3</u>	<u>64.6</u>	<u>73.2</u>
LNC	**39.5**	<u>64.0</u>	73.1	**40.6**	**64.8**	**73.5**

The best and second best results are highlight in bold and underline
*Indicates the adding a warmup process to SGR

improves R@1 by 4.5% and 5.7% in text and image retrieval, respectively. Moreover, SGR* outperforms SGR with a large performance margin, showing the sensitivity and unreliability of triplet loss with hard negatives to the NC.

4.2 Video-Text Retrieval

In this section, we conduct experiments on the video-text retrieval task to show the effectiveness of LNC. Video-text retrieval aims to retrieve related video clips with given query text. We verify our method on the three public datasets including MSR-VTT, LSMDC, and YouCook2 with synthetic noisy correspondence similar to the image-text retrieval task.

4.2.1 Experiment Settings

Implementation: For video-text retrieval, following to Miech et al. (2019), we first use the 2D and 3D CNNs to obtain the frame-level and video-level representations. Specifically, we extract the 2D features by a pre-trained Resnet-152 (He et al., 2016) on ImageNet with one frame per second rate; and extract the 3D features by a pre-trained ResNeXt-101 16-frames model (Hara et al., 2018) on Kinetics (Carreira & Zisserman, 2017). For the text, we use the word2vec embedding model (Mikolov et al., 2013) pre-trained on GoogleNews to obtain the text representations. Similar to Miech et al. (2019), we adopt the embedding function used in Miech et al. (2018) to embed the video clip and text into a shared. Then we compute the cosine affinity between the video clip and captions for retrieval. We adopt the triplet loss with the Adam optimizer for network optimization.

Datasets: MSR-VTT (Xu et al., 2016) collects 200k unique video clips with human-annotated captions from YouTube. Following Xu et al. (2016), we take the testing data of MSR-VTT retrieval for evaluation. LSMDC (Rohrbach et al., 2017) contains 101k unique video clip-caption pairs about movies. We take the testing data (1000 pairs) of LSMDC for evaluation. YouCook2 (Zhou et al., 2018) contains 14k video clips along with human-annotated captions about cooking. We take the 3.5k validation pairs for evaluation. As MSR-VTT, LSMDC and YouCook2 are well annotated, similar to the images, we randomly shuffle the captions of video clips for a specific ratio.

4.2.2 Results on MSR-VTT, LSMDC and YouCook2

Table 3 shows the quantitative results on the three video datasets with various noise ratios including 20% and 50%. As shown, LNC generally outperforms the baselines on three datasets. Specifically, LNC improves R@1 (20% and 50% noise) by (1.8%, 2.8%), (1.2%, 0.2%) and (0.9%, 0.1%) progress in MSR-VTT, LSMDC and YouCook2 than the baseline respectively. Moreover, similar to the image-text retrieval, LNC still achieves competitive performance in the noise-free case.

4.3 Comparison to the Noisy Annotation Learning Method

To empirically show the effectiveness of our method, we have added the comparisons with DivideMix (Li et al., 2020) for dealing with noisy correspondence. As the standard noisy annotation methods cannot be directly used to address the NC challenge as demonstrated in the Sect. 2.1, we take a two-stage pipeline by employing DivideMix to filter out the noisy samples and use the remaining clean samples to train the state-of-the-art cross-modal matching model, i.e., SGR (Diao et al., 2021). Table 4 shows the experiments on the Flickr30K dataset with 20% noise. One could observe that our method outperforms the DivideMix by 5.2% and 4.9% in terms of R@1 on image retrieval and text retrieval, which verifies the effectiveness of the proposed method and the necessity of NC-specific methods.

4.4 Comparison to the Large-Scaled Pre-trained Models

We conduct comparison experiments between LNC and the large-scaled pre-trained models in two different settings, i.e., NC in the fine-tune period and NC in the pre-training period. For the first setting, we assume the noisy correspondence existed in the fine-tuning data thus degrading the downstream tasks. To show the superiority of LNC in such an NC setting, we compare it to the recently proposed large-scale pre-trained model CLIP (Radford et al., 2021) on the image-text retrieval task. We directly apply the released pre-trained CLIP model (ViT-B/32) on the Noisy MS-COCO (i.e., with 20% and 50% noisy correspondence). For the second setting, we assume

Table 3 Video-text retrieval on MSR-VTT, LSMDC and YouCook2

Noise	Method	MSR-VTT				LSMDC				YouCook2			
		R@1 ↑	R@5 ↑	R@10 ↑	MR ↓	R@1 ↑	R@5 ↑	R@10 ↑	MR ↓	R@1 ↑	R@5 ↑	R@10 ↑	MR ↓
0%	Random	0.1	0.5	1.0	500	0.1	0.5	1.0	500	0.03	0.15	0.3	1675
	Torabi et al. (2016)	4.2	12.9	19.9	55	4.3	12.6	18.9	98	–	–	–	–
	Kiros et al. (2014)	3.8	12.7	17.1	66	3.1	10.4	16.5	79	–	–	–	–
	Kaufman et al. (2017)	4.7	16.6	24.1	41	4.7	15.9	23.4	64	–	–	–	–
	Yu et al. (2017)	4.4	16.6	22.3	35	4.5	14.1	20.9	67	–	–	–	–
	Yu et al. (2018)	10.2	31.2	43.2	13	**9.1**	**21.2**	**34.1**	**36**	–	–	–	–
	Miech et al. (2019)	12.1	35.0	**48.0**	**12**	7.2	18.3	25.0	44	4.2	**13.7**	**21.5**	**65**
	LNC	**13.4**	**35.9**	47.7	**12**	7.6	18.2	26.4	48	**4.4**	13.1	20.4	66
20%	Miech et al. (2019)	9.1	27.4	40.5	**16**	4.2	**13.7**	**22.4**	**56**	2.5	9.5	15.2	**110**
	LNC	**10.9**	**28.9**	**41.7**	**16**	**5.4**	13.5	21.3	68	**3.4**	**10.6**	**16.08**	129
50%	Miech et al. (2019)	5.4	20.5	31.4	**30**	3.4	9.8	**17.8**	69	1.6	**5.6**	8.8	**268**
	LNC	**8.2**	**23.3**	**32.6**	34	**3.6**	**10**	15.6	**68**	**1.7**	**5.6**	**8.9**	292

The best results are highlight in bold

Table 4 Comparison with DivideMix on Flickr30K

Methods	Image → Text			Text → Image		
	R@1	R@5	R@10	R@1	R@5	R@10
DivideMix+SGR	71.1	91.1	94.8	53.5	78.1	85.8
LNC	**76.3**	**93.7**	**96.9**	**58.4**	**83.8**	**89.8**

Bold indicate the default setting

the noisy correspondence existed in the pre-training data. To show the effectiveness of LNC in handling the NC in the pre-training, we compare our method to the large model H100M (Miech et al., 2019) on the video-text retrieval task. H100M is pre-trained on the billions of video clips that are reported to have many wrongly-matched video-captions pairs.

4.4.1 NC in the Fine-Tuning

CLIP is a large model pre-trained on massive image-text pairs collected from the Internet in which a large number of mismatched image-text pairs are wrongly treated as positive. CLIP proposes using hundreds of million data to alleviate the influence of the noisy pairs. In contrast, we believe that an NC-specific algorithm (i.e., LNC) is essential for handling the NC problem in the pre-training era.

In the following experiments, we compare the LNC to the CLIP by using the MS-COCO for evaluation. The experiments are conducted on two settings, i.e., Zero-shot and Fine-tune. In detail, in zero-shot, we directly perform inference on MS-COCO by using the released pre-trained CLIP (compared to the LNC trained on the original MS-COCO). In fine-tune, we first fine-tune the released CLIP model on MS-COCO and perform the inference on the testing data (compared to LNC trained on Noisy MS-COCO). Because the authors only released some CLIP models and the testing script, we adopt the non-official training script [1] to fine-tune the released CLIP in the fine-tune setting. Note that CLIP (ViT-L/14†) is unreleased and we only report the results from the original paper (Radford et al., 2021) in the zero-shot setting and compare it to the LNC trained on the original MS-COCO. As shown in Table 5, although CLIP uses massive training data for pre-training (about 400 million pairs), the noisy correspondence will inevitably degrade the performance during fine-tuning. Conversely, NCR achieves better matching performance compared to CLIP in the noisy MS-COCO data, showing the necessity and importance of algorithm design for handling the NC problem.

[1] https://github.com/Zasder3/train-CLIP-FT

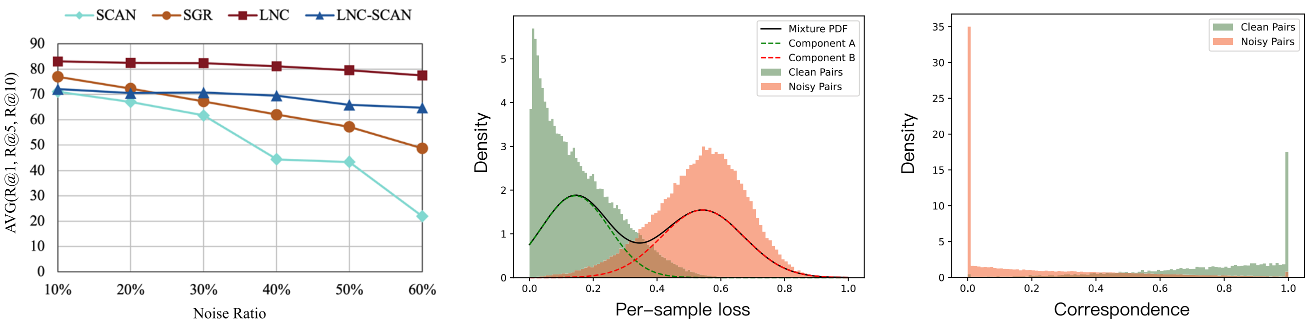
Table 5 Comparison with CLIP on MS-COCO 5K

Noise Ratio	Methods	Image → Text			Text → Image		
		R@1	R@5	R@10	R@1	R@5	R@10
0%, Zero-Shot	CLIP (ViT-L/14†)	**58.4**	81.5	88.1	37.8	62.4	72.2
	CLIP (ViT-B/32)	50.2	74.6	83.6	30.4	56.0	66.8
0%, Fine-tune	CLIP (ViT-B/32)	56.2	82.2	89.7	**45.3**	**73.5**	**83.0**
	LNC	**58.2**	**84.6**	**91.6**	41.9	71.0	81.6
20%, Fine-tune	CLIP (ViT-B/32)	21.4	49.6	63.3	14.8	37.6	49.6
	LNC	**57.3**	**83.9**	**90.9**	**41.1**	**69.8**	**80.4**
50%, Fine-tune	CLIP (ViT-B/32)	10.9	27.8	38.3	7.8	19.5	26.8
	LNC	**53.8**	**81.5**	**89.4**	**38.5**	**67.0**	**78.0**

Table 6 Video-text retrieval on MSR-VTT with pre-trained model on HowTo100M

Noise	Method	MSR-VTT			
		R@1 ↑	R@5 ↑	R@10 ↑	MR ↓
No Pre-training	Random	0.1	0.5	1.0	500
	Torabi et al. (2016)	4.2	12.9	19.9	55
	Kiros et al. (2014)	3.8	12.7	17.1	66
	Kaufman et al. (2017)	4.7	16.6	24.1	41
	Yu et al. (2017)	4.4	16.6	22.3	35
	Yu et al. (2018)	10.2	31.2	43.2	13
	Miech et al. (2019)	12.1	35.0	**48.0**	**12**
	LNC	**13.4**	**35.9**	47.7	**12**
Pre-training: Zero-shot	Miech et al. (2019)	7.5	21.2	**29.6**	38
	LNC	**8.5**	**21.8**	29.4	**34**
Pre-training: Finetune	Miech et al. (2019)	14.9	40.2	52.8	9
	LNC	**17.3**	**40.7**	**55.7**	**8**

The best results are highlight in bold



(a) The Noise Ratio (b) Co-divide (c) Co-rectify

Fig. 5 **a** The Retrieval performance of LNC and LNC-SCAN on Flickr30K. **b** The visualization of loss distribution and GMM fitting results. **c** The visualization of the rectified correspondence

4.4.2 NC in the Pre-training

HowTo100M (Miech et al., 2019) is a large-scale dataset that consists of 136 million video clips collected from Youtube. In general, the videos describe over 23k different visual tasks. Specifically, the data are collected by searching for YouTube videos related to the given task with a query about *how* to proceed with the task. The associated subtitles are treated as the corresponding caption of the video clips. As pointed out by Miech et al. (2019), as the captions of HowTo100M are automatically generated through the narration, the dataset inevitably contains weakly/wrongly paired data. The authors have evaluated some sampled pairs from the dataset and found that about 51% are at least weakly related. In other words, there are about 50% NC in HowTo100M.

To show the effectiveness of the proposed method on such a large-scale dataset from the wild, we directly use our LNC to endow the model (Miech et al., 2019) with robustness against the noisy correspondence. Specifically, we first pre-train the model (Miech et al., 2019) under the LNC framework on the billions of video clips with corresponding captions and then evaluate it on the MSR-VTT for the video-text retrieval task. The experimental results are shown in Table 6. One could observe that our extension outperforms the baseline in both zero-shot and fine-tune settings. Such a result demonstrates the effectiveness of the LNC and the necessity in explicitly handling the noisy correspondence in the pre-training era.

4.5 Analysis Experiment

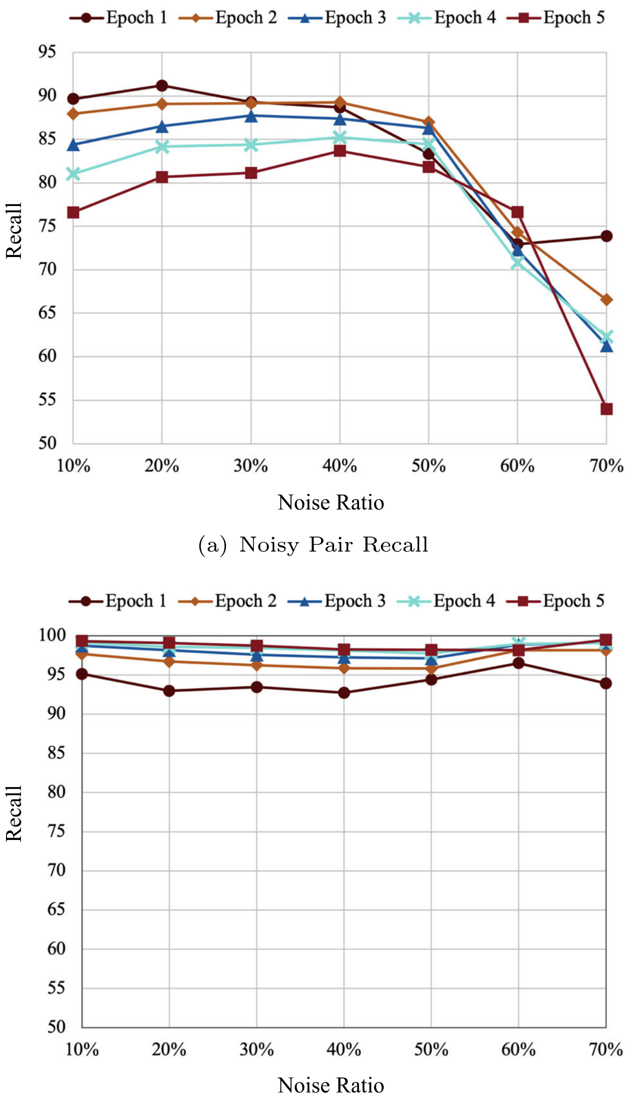
In this section, we conduct analysis experiments to provide a comprehensive evaluation on the proposed method.

4.5.1 Robustness & Generalization

To show the robustness of LNC in handling various noise ratios, we evaluate LNC on Flickr30K with different noise ratios including 0%, 10%, . . ., 60%. Moreover, to demonstrate the generalizability of LNC to other retrieval models, we also extend the well-known method SCAN (Lee et al., 2018) by LNC, denoted by LNC-SCAN. As shown in Fig. 8, LNC and LNC-SCAN achieve more stable performance than the baselines SGR and SCAN with increasing noise ratio from 0% to 60%, demonstrating the capacity of LNC for handling in various noise ratios. Moreover, LNC-SCAN remarkably outperforms SCAN in all cases, showing the generalizability of LNC to different models.

4.5.2 Ablation Study

In this section, we investigate the influence of different modules in LNC. First, to show the effectiveness of co-divide and co-rectify, we visualize the loss distribution and the model predictions of the training data in Flickr30K with 50% noise. For simplicity, here we only present the visualization result from LNC-SCAN and provide the visualization of LNC in the supplementary material. As shown in Fig. 5b, after a few training epochs, noisy samples have a larger loss value than the clean ones, verifying the memorization effect of DNNs. Moreover, the data are successfully divided into “clean” and “noisy” subsets by using the GMM. Fig. 6b visualizes the rectified soft correspondence obtained by the co-rectify module. As shown, the correspondence of clean samples are roughly rectified to [0.4, 1] and those of noisy ones are roughly rectified to [0, 0.6], showing the effectiveness of the co-rectify module in LNC. In addition, we also conduct the quantity experiment to investigate the performance of co-divide module. The experiment results are shown in Table 7. In the experiment, we report the noise recall, precision and F1-Score on the detected noisy correspondence by our method after warmup. Notably, the F1-scores for noise detection are about 88% $\sim$ 91%, indicating that our method effectively identifies nearly all noisy pairs, regardless of whether the noise constitutes 20% or 50% of the dataset.



(a) Noisy Pair Recall

(b) Clean Pair Recall

Fig. 6 Noisy and clean samples detection

Besides the visualization, we carry out the ablation study to further investigate the roles of different modules on the proposed method. We conduct experiments on the Flickr30K with 20% noise. Specifically, we provide the results by removing or replacing the modules in LNC including (1) Removing the warmup step; (2) Removing the co-divide module; (3) Removing the co-rectify module; (4) Using $\mathcal{L}_{tri}$ for warmup; (5) Using $\mathcal{L}_w$ for warmup; (6) Using RNA tech-

Table 7 Noisy correspondence identification

Dataset	Noise ratio (%)	Recall (%)	Precision (%)	F1-Score (%)
Flickr30K	20	83.04	94.22	88.28
	50	87.04	96.35	91.46
MS-COCO	20	99.20	80.57	88.92
	50	85.01	99.46	91.67

Table 8 Ablation studies on Flickr30K with 20% noise

Method					Image → Text			Text → Image		
Warmup			Co-divide	Co-rectify	R@1	R@5	R@10	R@1	R@5	R@10
Vanilla	MR	RNA								
			✓	✓	0.1	0.4	1.0	0.1	0.4	0.9
✓				✓	72.0	90.8	94.6	53.4	78.4	84.9
✓			✓		70.3	90.3	95.0	53.7	78.5	85.4
✓			✓	✓	75.0	**93.9**	97.5	58.3	83.0	89.0
	✓		✓	✓	75.4	93.4	**97.6**	58.0	83.6	88.9
		✓	✓	✓	75.7	93.6	96.9	58.3	82.5	89.0
	✓	✓	✓	✓	**76.3**	93.7	96.9	**58.4**	**83.8**	**89.8**

The best and second best results are highlight in bold and underline

Table 9 Ablation studies on four margin recasting functions on MS-COCO

Noise Ratio	Label $\hat{y}$ → Margin $\hat{\alpha}$	Image → Text			Text → Image		
		R@1	R@5	R@10	R@1	R@5	R@10
20%	Ψ_1—Linear	75.4	93.1	**97.1**	57.7	83.1	88.8
	Ψ_2—Exponential	76.3	**93.7**	96.9	**58.4**	**83.8**	**89.8**
	Ψ_3—Sin	75.8	93.5	96.8	57.3	82.7	88.9
	Ψ_4—Sigmoid	**76.5**	**93.7**	**97.2**	57.1	82.7	89.2
50%	Ψ_1 - Linear	72.8	91.7	95.6	53.7	79.5	87.0
	Ψ_2—Exponential	**73.0**	**92.5**	**96.1**	**54.1**	**80.5**	**87.2**
	Ψ_3—Sin	70.7	91.1	95.7	53.2	80.3	86.7
	Ψ_4—Sigmoid	71.2	91.3	95.3	53.7	79.8	86.8

The best results are highlight in bold

nique for warmup; (7) Using both $\mathcal{L}_w$ and RNA technique for warmup (i.e., LNC); As shown in Table 8, all the proposed modules are crucial for the proposed method LNC to obtain encouraging performance.

To investigate the noise detection power of LNC, we conduct experiments on Flickr30K with various noise ratios and report the noisy and clean detection performance in terms of the recall metric. As shown in Fig. 6, the proposed method LNC consistently identifies most of the clean samples in various noise ratios. For noise detection, LNC generally degrades with the increasing noise ratio. Moreover, with increasing epochs, the noise detection performance generally decreases as the neural network gradually fits the noisy samples, demonstrating the memorization effect of DNNs. Furthermore, with the powerful noisy/clean sample detection capacity, LNC could be a pluggable module to other methods in various tasks to enable robustness to NC.

Moreover, we also investigate the proposed four margin recasting strategies in the co-rectify module as discussed in Sect. 3.3. We conduct experiments on the Flickr30K data with 20% and 50% noise and report the results in Table 9. From the table, one could observe that the Ψ_2 (Exponential) achieves the best performance, demonstrating the effectiveness of restraining the probable noisy samples in Eq. 10. In addition, although Ψ_3 and Ψ_4 are designed for restraining the margins of pairs with small $\hat{y}$ and amplify that of the pairs with large $\hat{y}$, the performance is sub-optimal as the estimated noisy/clean boundary is unmatched to the real data distribution. Note that, the different margin recasting functions could be adaptively used for handling difference noise distributions in the real-world applications as shown in Fig. 4. In general, Ψ_1 has no specific action for margin recasting as it transforms it linearly; Ψ_2 and Ψ_3 both restrain the probable noisy sample and amplify the clean samples but with a slight difference

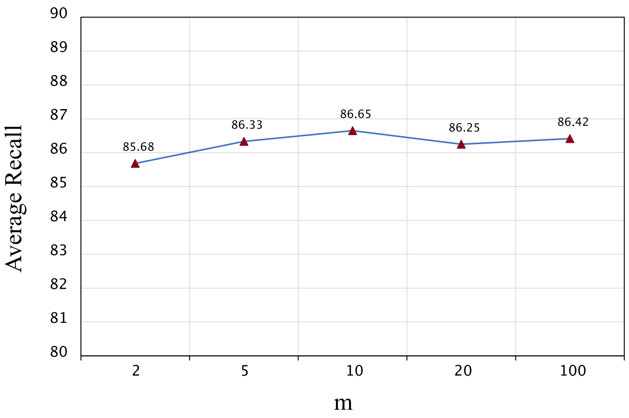


Fig. 7 Performance of Ψ_2 with various m on MS-COCO with 20% noise

in the noise/clean boundary; Ψ_4 aims to restrain almost all the possible noise samples which is suitable for indiscoverable noise by the co-divide module. Moreover, the different noise ratios exert minimal impact on the performance of the various Ψ functions. This can be attributed to the fact that all Ψ functions adhere to the same underlying principle of assigning an adaptive margin based on the rectified labels.

In addition, we have conducted additional sensitivity experiments on the selected Ψ_2 function to examine its performance and stability. Note that Ψ_2 can be written by:

$$\Psi_2 = \frac{m^{\hat{y}_i} - 1}{m - 1} \times \alpha, m > 1$$

So we varied the value of m in 2, 5, 10, 20, 100 and evaluated the results. The experiment outcomes are displayed in Figure 7. As illustrated, LNC exhibits consistent and stable performance across different values of m in Ψ_2.

4.5.3 Case Study

Fig. 8 provides some mismatched image-text pairs identified by LNC from CC152K. The detected image-text pairs are unrelated in semantics according to the EME score and are successfully identified by LNC. The EME is a quantitative metric designed to assess the level of correspondence between cross-modal pairs which will be introduced in Appendix A. Moreover, we conduct the experiment on MS-COCO to showcase the co-rectify ability to recall clean pairs from the “noisy” subset, as illustrated in Fig. 9. Our results demonstrate that our approach is capable of detecting clean pairs that are wrongly divided into noisy subset. Notably, even in cases where the original correspondence was negative (the last image-text pair framed in red), we were able to rectify the correspondence to a score of 0.88 by considering the related concepts shared between the image and caption, which can be considered a weakly matched pair.

5 Conclusion

This paper studies a new problem in multi-modal learning, i.e., noisy correspondence which is a new learning paradigm for noisy labels. To address this problem, we propose a novel method termed learning with noisy correspondence which first distinct the noisy pairs from the clean ones and then rectifies the false positive correspondence. With a novel triplet loss, LNC endows the cross-modal retrieval model with robustness against false positive pairs. Extensive experiments on six benchmark datasets demonstrate the effectiveness of the proposed method LNC. Furthermore, this study shows that the noisy correspondence expects an explicit solution, especially in the pre-training model era.

Acknowledgements The authors would like to thank the associate editor and reviewers for the constructive comments and valuable suggestions that remarkably improve this study. This work was supported in part by NSFC under Grant U21B2040, 62176171; and in part by the Fundamental Research Funds for the Central Universities under Grant CJ202303.

Appendix A: Exactly Matched Element (EME)

To quantify the noise in the correspondence, here we introduce the exactly matched element (EME) score which evaluates the similarity of image and text pairs based on how many elements they share in common. Formally,

$$\text{EME} = \frac{1}{2N_I} \sum_{e_i} E(e_i, T) + \frac{1}{2N_T} \sum_{e_t} E(e_t, I) \qquad \text{(A1)}$$

where e_i and e_t are meaningful elements extracted from the image I and the text T respectively, N_I and N_T are the number of elements in I and T respectively, the function $E(e_i, T)$ is an indicator function that outputs 1 if the element e_i is accurately described in T, and 0 otherwise. Similarly, $E(e_t, I)$ is an indicator function that outputs 1 if the element e_t is depicted in I, and 0 otherwise. EME could be considered as the correspondence label of cross modal pairs. To obtain EME score, we have two main approaches. Firstly, we can compute the EME score with human annotations to ensure accuracy. Alternatively, we can leverage advanced techniques such as semantic segmentation or object detection on images to extract all visual elements, and employ text segmentation methods on text to extract all textual elements. Subsequently, we can calculate EME by utilizing a visual-language model as the indicator (i.e., the function $E(e_i, T)$ and $E(e_t, I)$), such as CLIP. Such an indicator helps identify whether the extracted elements from one modality are described in the other modality.

The EME degree ranges from 0 to 1, where higher values indicate higher similarity between image and text pairs.

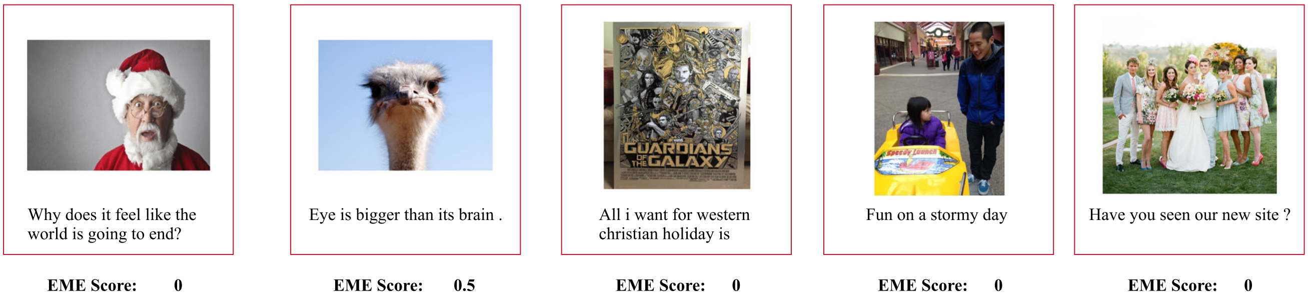

Why does it feel like the world is going to end?
EME Score: 0
Eye is bigger than its brain .
EME Score: 0.5
All i want for western christian holiday is
EME Score: 0
Fun on a stormy day
EME Score: 0
Have you seen our new site ?
EME Score: 0

Fig. 8 Some noisy examples correctly detected by LNC in the Conceptual Captions dataset

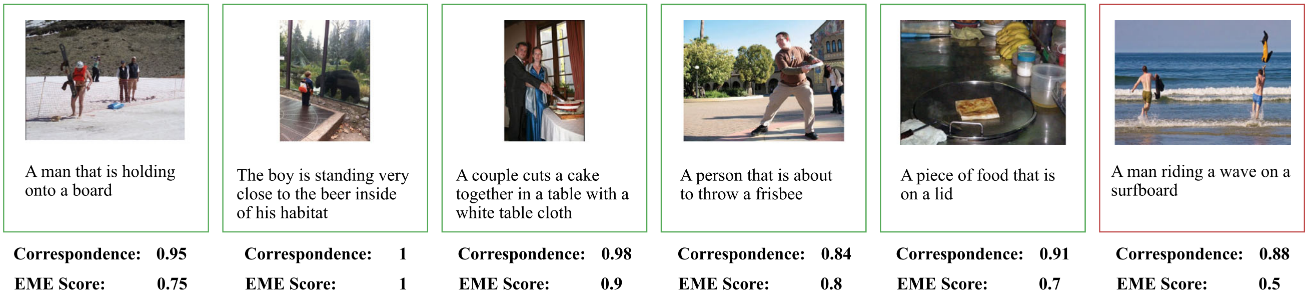

A man that is holding onto a board
Correspondence: 0.95
EME Score: 0.75
The boy is standing very close to the beer inside of his habitat
Correspondence: 1
EME Score: 1
A couple cuts a cake together in a table with a white table cloth
Correspondence: 0.98
EME Score: 0.9
A person that is about to throw a frisbee
Correspondence: 0.84
EME Score: 0.8
A piece of food that is on a lid
Correspondence: 0.91
EME Score: 0.7
A man riding a wave on a surfboard
Correspondence: 0.88
EME Score: 0.5

Fig. 9 Some clean pairs that are correctly detected by LNC from “noisy” subset divided by the Co-divide module

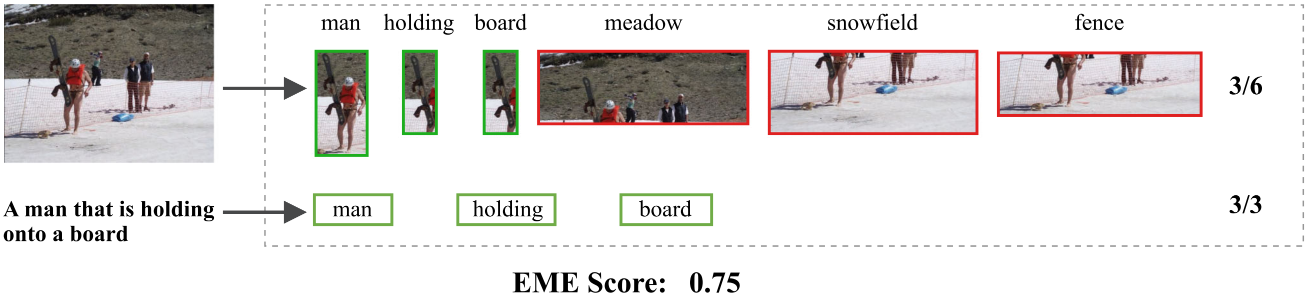


Fig. 10 An example of the process for calculating the EME score

For example, if an image and a text are completely unrelated, their EME degree will be 0. If they are partially related, their EME degree will be between 0 and 1. If they are fully related, their EME degree will be 1. A toy example is illustrated in Fig. 10 to show how to calculate the EME score of an image-text pair. Note that, the calculation of EME is contingent upon the quantity of shared content across various modalities. Thus EME score primarily estimates the level of cross-modal semantic completeness and lacks the capability to assess other intricate relationships, such as contradictions or complements.

Appendix B: Algorithm

Here we provide the detail algorithm of LNC in Algorithm 1.

Algorithm 1: The algorithm of LNC

```
Input: Training data $\mathcal{D}$ with datasize $N$, retrieval models $A = (f^A, g^A, S^A)$ and $B = (f^B, g^B, S^B)$
// Network Warmup
$\mathcal{D}_W = \mathcal{D}$;
for n=1:warmup_epochs do
    for k={A, B} do
        // for each network
        Train the network k on $\mathcal{D}_W$ by optimizing $L_w$;
    end
    // Clean confidence
    $\mathcal{W}^A = \{w_i^A\}_{i=0}^N \leftarrow GMM(B, \mathcal{D})$;
    $\mathcal{W}^B = \{w_i^B\}_{i=0}^N \leftarrow GMM(A, \mathcal{D})$;
    $\mathcal{D}_d = \{(\mathbf{v}_j, \mathbf{t}_j)|w_j^A < 0.5$, and $w_j^B < 0.5, \forall j \in N\}$; // Detected noisy data from A and B
    $\mathcal{D}_w = \mathcal{D} - RandomSelection(\mathcal{D}_d, \frac{1}{2}|\mathcal{D}_d|)$; // Randomly abandon half noisy data
end
// Network Training
for n=1:training_epochs do
    // Clean confidence
    $\mathcal{W}^A = \{w_i^A\}_{i=0}^N \leftarrow GMM(B, \mathcal{D})$;
    $\mathcal{W}^B = \{w_i^B\}_{i=0}^N \leftarrow GMM(A, \mathcal{D})$;
    for $k \in \{A, B\}$ do
        // Co-divide data
        $\mathcal{S}_c^k = \{(I_i, T_i, y_i, w_i)|w_i \geq 0.5, \forall(I_i, T_i, y_i, w_i) \in (\mathcal{D}, \mathcal{W}^k)\}$ ;
        $\mathcal{S}_n^k = \{(I_i, T_i)|w_i < 0.5, \forall(I_i, T_i) \in (\mathcal{D}, \mathcal{W}^k)\}$ ;
        for j=num_steps do
            Sampling Mini-batch: $(\mathcal{B}_j^c, \mathcal{B}_j^n) \leftarrow (\mathcal{S}_c^k, \mathcal{S}_n^k)$;
            Rectifying the correspondence using Eq. 8–9: $(\hat{\mathcal{B}}_j^c, \hat{\mathcal{B}}_j^n) \leftarrow (\mathcal{B}_j^c, \mathcal{B}_j^n)$;
            Optimizing Network k by minimizing $L_{soft}$ on the rectified data $(\hat{\mathcal{B}}_j^c, \hat{\mathcal{B}}_j^n)$.
        end
    end
end
Result: Retrieval models $(A, B)$
```

Appendix C: Additional Experiments

C.1 Evaluation on the MS-COCO 5K Testing Set

Here we provide the additional comparison results on the 5K testing set of MS-COCO. As shown in Table 10, LNC achieves SOTA results in the non-noise case. In the cases with noisy correspondence, LNC remarkably outperforms all the baselines. Specifically, in the noisy setting, LNC improves R@1 by 3.9%, 2.7%, 3.7%, and 3.1% in text and image retrieval compared to the best baseline SGR-C.

Table 10 Image-text retrieval on MS-COCO 5K

Noise	Methods	Image → Text			Text → Image		
		R@1	R@5	R@10	R@1	R@5	R@10
0%	SCAN	44.7	75.9	86.6	33.3	63.5	75.4
	VSRN	53.0	81.1	89.4	40.5	70.6	81.1
	IMRAM	53.7	83.2	91.0	39.7	69.1	79.8
	SAF	53.3	–	90.1	39.8	–	80.2
	SGR	56.9	–	90.5	40.2	–	79.8
	SGRAF	57.8	–	91.6	41.9	–	81.3
	NAAF	**58.9**	**85.2**	**92.0**	**42.5**	70.9	81.4
	NCR	58.2	84.2	91.5	41.7	**71.0**	81.3
	LNC	58.2	84.6	91.6	41.9	**71.0**	**81.6**
20%	SCAN	42.4	72.1	82.6	22.8	52.3	66.3
	VSRN	8.9	26.5	40.2	5.7	20.3	31.4
	IMRAM	44.3	75.5	85.7	34.1	63.1	74.5
	SAF	42.7	73.8	83.7	31.6	60.8	72.9
	SGR*	44.6	73.5	83.7	31.4	60.4	72.4
	NAAF	44.6	75.8	85.5	37.1	65.7	76.1
	SGR-C	53.4	81.5	89.3	38.4	67.8	78.8

Table 10 continued

Noise	Methods	Image → Text			Text → Image		
		R@1	R@5	R@10	R@1	R@5	R@10
	NCR	56.9	83.6	**91.0**	40.6	**69.8**	80.1
	LNC	**57.3**	**83.9**	90.9	**41.1**	**69.8**	**80.4**
50%	SCAN	18.5	44.5	58.9	2.2	6.2	9.6
	VSRN	8.3	25.4	37.7	4.8	18.1	29.2
	IMRAM	5.0	23.0	38.5	8.1	26.0	38.3
	SAF	10.4	32.8	48.2	15.2	38.3	51.9
	SGR*	36.4	64.8	77.1	26.0	52.9	64.3
	NAAF	26.8	56.2	68.9	18.2	44.1	57.5
	SGR-C	50.1	77.4	86.8	35.4	64.5	76.0
	NCR	53.1	80.7	88.5	37.9	66.6	77.8
	LNC	**53.8**	**81.5**	**89.4**	**38.5**	**67.0**	**78.0**

The best and second best results are highlight in bold and underline

C.2 Case Study

In this section, we show some qualitative results of LNC. The example image-text retrieval results are shown in Fig. 11 and Fig. 12. As shown in Fig. 11 (1)–(4) and Fig. 11 (1)–(5), LNC could successfully retrieve the corresponding samples with given queries. Moreover, we provide some failure cases from LNC in Fig. 11 (5)–(6) and Fig. 12 (6). Interestingly, the retrieved image from LNC is fit to the query compared to the ground truth in Fig. 12 (6).

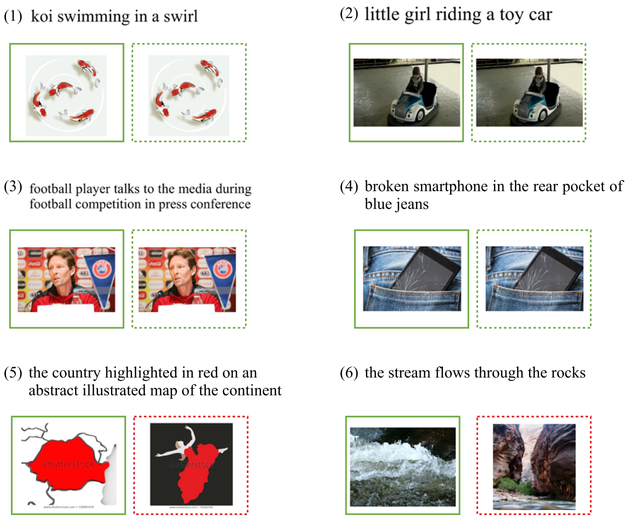

(1) koi swimming in a swirl
(2) little girl riding a toy car
(3) football player talks to the media during football competition in press conference
(4) broken smartphone in the rear pocket of blue jeans
(5) the country highlighted in red on an abstract illustrated map of the continent
(6) the stream flows through the rocks

Fig. 11 Some retrieved images by the LNC from CC152K. The left images are the ground truth while the right are the retrieved ones. The successfully retrieved images and the failure cases are highlighted by green dashed boxes and red dashed boxes, respectively (Color figure online)

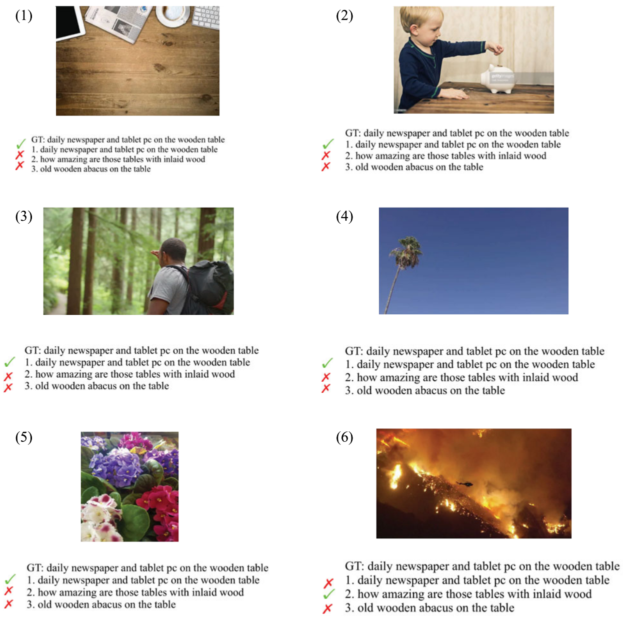
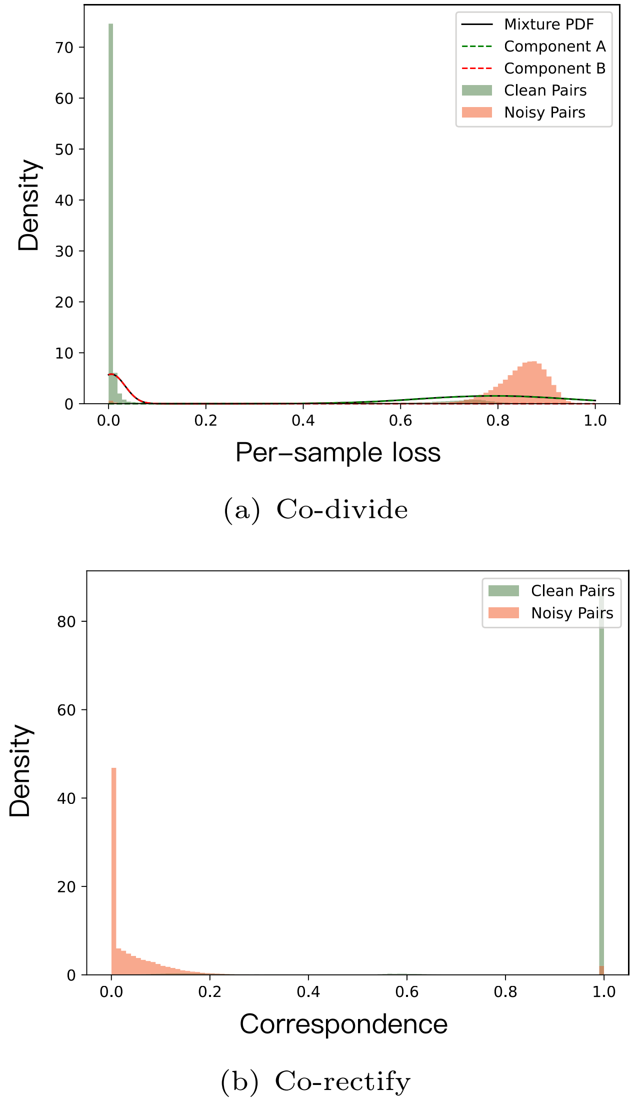

(1)
GT: daily newspaper and tablet pc on the wooden table
✓ 1. daily newspaper and tablet pc on the wooden table
✗ 2. how amazing are those tables with inlaid wood
✗ 3. old wooden abacus on the table
(2)
GT: daily newspaper and tablet pc on the wooden table
✓ 1. daily newspaper and tablet pc on the wooden table
✗ 2. how amazing are those tables with inlaid wood
✗ 3. old wooden abacus on the table
(3)
GT: daily newspaper and tablet pc on the wooden table
✓ 1. daily newspaper and tablet pc on the wooden table
✗ 2. how amazing are those tables with inlaid wood
✗ 3. old wooden abacus on the table
(4)
GT: daily newspaper and tablet pc on the wooden table
✓ 1. daily newspaper and tablet pc on the wooden table
✗ 2. how amazing are those tables with inlaid wood
✗ 3. old wooden abacus on the table
(5)
GT: daily newspaper and tablet pc on the wooden table
✓ 1. daily newspaper and tablet pc on the wooden table
✗ 2. how amazing are those tables with inlaid wood
✗ 3. old wooden abacus on the table
(6)
GT: daily newspaper and tablet pc on the wooden table
✗ 1. daily newspaper and tablet pc on the wooden table
✓ 2. how amazing are those tables with inlaid wood
✗ 3. old wooden abacus on the table

Fig. 12 Some retrieved captions by the LNC from CC152K. The top sentence is the ground truth while the rest are the retrieved top 3 captions. The successfully retrieved images and the failure cases are highlighted by ✓ and ✗, respectively

C.3 Co-divide and Co-rectify from LNC

In this section, we conduct analysis experiments to further study the influence of co-divide and co-rectify modules. First, we provide the visualization on co-divide and co-rectify from LNC in Fig. 13. As one could observe, the noisy and clean pairs are well divided and rectified by our method.

Besides, to evaluate the impact of our confidence estimation, we performed an ablation study by setting w_i to 1 for clean samples and 0 for noisy samples, based on the ground truth labels. We denote this method as LNC ($w_i^c = 1, w_i^n = 0$). The results are shown in Table 11. Interestingly, our LNC achieved better results than LNC ($w_i^c = 1, w_i^n = 0$), despite the latter having access to the ground truth labels. This indicates that our confidence estimation can effectively capture the uncertainty of data correspondence, including fully-matched, partially-matched, and unmatched image-text pairs, and thus improve the cross-modal matching performance.



(a) Co-divide

(b) Co-rectify

Fig. 13 **a** The visualization of loss distribution and GMM fitting results from LNC. **b** The visualization of the rectified correspondence from LNC

Appendix D: Implementation Detail

D.1 Image-text Retrieval

Here we detail how LNC adopts SGR for cross-modal retrieval.

Specifically, for images, the visual features of K local regions are extracted by the Faster R-CNN (Ren et al., 2015). Then we obtain the local embeddings $\{\mathbf{v}_1, \ldots, \mathbf{v}_K\}$ by embedding the above visual features by a fully connected layer f. The global embeddings are obtained via the self-attention mechanism (Vaswani et al., 2017). Moreover specifically, we aggregate all the local embeddings to obtain global embedding $\hat{\mathbf{v}}$ by treating the average local embeddings as query. For captions, the caption is spited into L words and are further represented by the 300-dimensional features with word embedding technique. Then the 1024-dimensional local embeddings $\{\mathbf{t}_1, \ldots, \mathbf{t}_L\}$ are obtained by a Bi-GRU (Schuster & Paliwal, 1997) $g(T)$. The global embeddings $\hat{\mathbf{t}}$ of captions are computed similar to the image.

With the extracted visual and textual embeddings, we compute the similarity vector for given pairs. In detail, the similarity vector is computed by:

$$s(\mathbf{v}_1, \mathbf{v}_2; \mathbf{W}) = \frac{\mathbf{W}|\mathbf{v}_1 - \mathbf{v}_2|^2}{\mathbf{W}\|\mathbf{v}_1 - \mathbf{v}_2\|^2} \tag{D2}$$

where $\mathbf{W}$ denotes a learnable matrix. Then we compute the similarity of global visual and textual embeddings as:

$$\mathbf{s}^g = \mathbf{s}(\hat{\mathbf{v}}, \hat{\mathbf{t}}; \mathbf{W}_g) \tag{D3}$$

and the similarity of local visual and textual embeddings:

$$\begin{aligned} \mathbf{s}_j^l &= \mathbf{s}(\mathbf{a}_j^v, \mathbf{t}_j; \mathbf{W}_l) \\ \mathbf{a}_j^v &= \sum_{i=1}^{K} \alpha_{ij} \mathbf{v}_i \end{aligned} \tag{D4}$$

where $\mathbf{a}_j^v$ denotes aggregated embeddings, α_{ij} denotes the attention coefficient:

$$\alpha_{ij} = \frac{exp(\lambda \hat{c}_{ij})}{\sum_{j=1}^{K} exp(\lambda \hat{c}_{ij})} \tag{D5}$$

Table 11 Ablation study on co-divide module by using MS-COCO

Noise Ratio	Method	Image → Text			Text → Image		
		R@1	R@5	R@10	R@1	R@5	R@10
20%	LNC ($w_i^c = 1, w_i^n = 0$)	77.2	95.4	98.2	61.6	89.2	95.1
	LNC	**78.2**	**95.8**	**98.5**	**62.6**	**89.4**	**95.4**
50%	LNC ($w_i^c = 1, w_i^n = 0$)	74.0	94.5	97.8	59.1	87.8	94.4
	LNC	**75.8**	**94.9**	**97.9**	**59.8**	**88.1**	**94.6**

Table 12 Experiment parameters

Dataset	Warmup Epoch	Epoch	LR Update	Batch Size
Flickr30K	3	40	30	128
MS-COCO	10	20	10	128
CC152K	5	30	10	128

where $\hat{c}_{ij}$ denotes the cosine similarity between the i-th image region and j-th word in a given image-text pair.

Once the similarity vectors $\mathcal{N} = \{\mathbf{s}_1^l, \mathbf{s}_2^l, \cdots, \mathbf{s}_K^l\}$ are obtained, we treat them as the similarity graph nodes and compute the graph edges as:

$$e(\mathbf{s}_p, \mathbf{s}_q; \mathbf{W}_{in}, \mathbf{W}_{out}) = \frac{exp((\mathbf{W}_{in}\mathbf{s}_p)(\mathbf{W}_{out}\mathbf{s}_q))}{\sum_q exp((\mathbf{W}_{in}\mathbf{s}_p)(\mathbf{W}_{out}\mathbf{s}_q))} \quad \text{(D6)}$$

where $\mathbf{W}_{in}$ and $\mathbf{W}_{out}$ are the learnable matrixes to transform the incoming and outgoing similarity. Finally, we aggregate all the similarities by updating the similarity of nodes and edges by

$$\begin{aligned} \hat{\mathbf{s}}_p^n &= \sum_q e(\mathbf{s}_p^n, \mathbf{s}_q^n; \mathbf{W}_{in}^n, \mathbf{W}_{out}^n) \cdot \mathbf{s}_q^n \\ \mathbf{s}_q^{n+1} &= ReLU(\mathbf{W}_r^n \hat{\mathbf{s}}_p^n) \end{aligned} \quad \text{(D7)}$$

where $\mathbf{W}_{in}^n$, $\mathbf{W}_{out}^n$ and $\mathbf{W}_r^n$ are learnable matrixes, $\mathbf{s}_p^0$ and $\mathbf{s}_q^0$ are the initial nodes from $\mathcal{N}$ at step $n = 0$. Specifically, it iteratively updates the similarity for N steps, and treats the global node as the reasoned similarity. Finally, we use a fully connected layer to compute the final similarity as $S(I, T)$ in LNC.

Here we provide the used parameters for training LNC in Table 12 including the number of epochs for warmup, the number of epochs for training, the number of learning rate update intervals (LR Update) and batch size.

D.2 Video-Text Retrieval

In the video-text retrieval experiment, we take the model proposed by Miech et al. (2019) as an example and extend it to be robust again noisy correspondence. Specifically, with the given video clip and caption $(\mathbf{v}, \mathbf{t})$, we adopt the class of non-linear embedding functions to obtain the visual and textual features, i.e.,

$$\begin{aligned} f(\mathbf{v}) &= \left(W_1^v \mathbf{v} + b_1^v\right) \circ \sigma\left(W_2^v\left(W_1^v \mathbf{v} + b_1^v\right) + b_2^v\right) \\ g(\mathbf{t}) &= \left(W_1^t \mathbf{t} + b_1^c\right) \circ \sigma\left(W_2^c\left(W_1^c \mathbf{t} + b_1^c\right) + b_2^c\right) \end{aligned} \quad \text{(D8)}$$

where W_1^v, W_1^t, W_2^v, and W_2^t are the learnable weight, b_1^v, b_1^t, b_2^v, and b_2^t are the learnable bias vectors, σ is an element-wise sigmoid activation and $\circ$ is the element-wise multiplication. In all experiments, we embed the clip and caption into 4096-dimensional space.

For all video-text experiments, we adopt the Adam optimizer with a learning rate of 0.0001 and set the batch size to 256. For the pre-training on HowTo100M data, we follow the default settings in Miech et al. (2019). The number of warmup epochs is fixed to 3 for all video datasets.

References

Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., & Zhang, L. (2018). Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6077–6086).

Arazo, E., Ortego, D., Albert, P., O'Connor, N., & McGuinness, K. (2019). Unsupervised label noise modeling and loss correction. In *International conference on machine learning, PMLR* (pp. 312–321).

Arpit, D., Jastrzebski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M. S., Maharaj, T., Fischer, A., Courville, A., Bengio, Y., & Lacoste-Julien, S. (2017). A closer look at memorization in deep networks. In *International Conference on Machine Learning, PMLR* (pp. 233–242).

Bai, Y., Yang, E., Han, B., Yang, Y., Li, J., Mao, Y., Niu, G., & Liu, T. (2021). Understanding and improving early stopping for learning with noisy labels. *Advances in Neural Information Processing Systems, 34*, 24392–24403.

Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A. & Raffel, C. A. (2019). MixMatch: A holistic approach to semi-supervised learning. arXiv preprint arXiv:1905.02249.

Carreira, J., & Zisserman, A. (2017). Quo vadis, action recognition? A new model and the kinetics dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6299–6308).

Chen, H., Ding, G., Liu, X., Liu J., & Han J. (2020). Imram: Iterative matching with recurrent attention memory for cross-modal image-text retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 12,655–12,663).

Deng, C., Chen, Z., Liu, X., Gao, X., & Tao, D. (2018). Triplet-based deep hashing network for cross-modal retrieval. *IEEE Transactions on Image Processing, 27*(8), 3893–3903.

Diao, H., Zhang, Y., Ma, L., & Lu, H. (2021). Similarity reasoning and filtration for image-text matching. In *AAAI*.

Faghri, F., Fleet, D. J., Kiros, J. R., & Fidler, S. (2017). VSE++: Improving visual-semantic embeddings with hard negatives. arXiv preprint arXiv:1707.05612.

Feng, L., & An, B. (2019). Partial label learning with self-guided retraining. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 3542–3549).

Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., & Sugiyama, M. (2018). Co-teaching: Robust training of deep neural networks with extremely noisy labels. arXiv preprint arXiv:1804.06872.

Han, J., Luo, P., & Wang, X. (2019). Deep self-learning from noisy labels. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 5138–5147).

Hara, K., Kataoka, H., & Satoh, Y. (2018). Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and imagenet? In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6546–6555).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Huang, Z., Niu, G., Liu, X., Ding, W., Xiao, X., Wu, H. & Peng, X. (2021). Learning with noisy correspondence for cross-modal matching. In *Thirty-Fifth conference on neural information processing systems*.

Jia, C., Yang, Y., Xia, Y., Chen, Y. T., Parekh, Z., Pham, H., Le, Q., Sung, Y. H., Li, Z., & Duerig, T. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. arXiv preprint arXiv:2102.05918.

Kaufman, D., Levi, G., Hassner, T., & Wolf, L. (2017). Temporal tessellation: A unified approach for video analysis. In *Proceedings of the IEEE international conference on computer vision* (pp. 94–104).

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Kiros, R., Salakhutdinov, R., & Zemel, R. S. (2014). Unifying visual-semantic embeddings with multimodal neural language models. arXiv preprint arXiv:1411.2539.

Kuang, Z., Gao, Y., Li, G., Luo, P., Chen, Y., Lin, L., & Zhang, W., (2019). Fashion retrieval via graph reasoning networks on a similarity pyramid. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 3066–3075).

Lee, K. H., Chen, X., Hua, G., Hu, H., & He, X., (2018). Stacked cross attention for image-text matching. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 201–216).

Li, J., Socher, R., & Hoi, S. C. (2020). DivideMix: Learning with noisy labels as semi-supervised learning. arXiv preprint arXiv:2002.07394.

Li, K., Zhang, Y., Li, K., & Fu, Y. (2019a). Visual semantic reasoning for image-text matching. In *ICCV*.

Li, S., Tao, Z., Li, K., & Fu, Y. (2019). Visual to text: Survey of image and video captioning. *IEEE Transactions on Emerging Topics in Computational Intelligence, 3*(4), 297–312.

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In *European conference on computer vision* (pp. 740–755). Springer.

Liu, T., & Tao, D. (2015). Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 38*(3), 447–461.

Miech, A., Laptev, I., & Sivic, J. (2018). Learning a text-video embedding from incomplete and heterogeneous data. arXiv preprint arXiv:1804.02516.

Miech, A., Zhukov, D., Alayrac, J. B., Tapaswi, M., Laptev, I., & Sivic, J. (2019). Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 2630–2640).

Mikolov, T., Chen, K., Corrado, G., & Dean J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Patrini, G., Rozza, A., Krishna Menon, A., Nock, R., & Qu, L. (2017). Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1944–1952).

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., & Krueger, G. (2021). Learning transferable visual models from natural language supervision. arXiv preprint arXiv:2103.00020.

Reed, S., Lee, H., Anguelov, D., Szegedy, C., Erhan, D., & Rabinovich, A. (2014). Training deep neural networks on noisy labels with bootstrapping. arXiv preprint arXiv:1412.6596.

Ren, S., He, K., Girshick, R. B., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*.

Rohrbach, A., Torabi, A., Rohrbach, M., Tandon, N., Pal, C., Larochelle, H., Courville, A., & Schiele, B. (2017). Movie description. *International Journal of Computer Vision, 123*(1), 94–120.

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing, 45*(11), 2673–2681.

Sharma, P., Ding, N., Goodman, S., & Soricut, R. (2018). Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th annual meeting of the association for computational linguistics* (Volume 1: Long Papers, pp. 2556–2565).

Shekhar, R., Pezzelle, S., Klimovich, Y., Herbelot, A., Nabi, M., Sangineto, E., & Bernardi, R. (2017). Foil it! find one mismatch between image and language caption. arXiv preprint arXiv:1705.01359.

Song, H., Kim, M., Park, D., Shin, Y., & Lee, J. G. (2020). Learning from noisy labels with deep neural networks: A survey. arXiv preprint arXiv:2007.08199.

Torabi, A., Tandon, N., & Sigal, L. (2016). Learning language-visual embedding for movie understanding with natural-language. arXiv preprint arXiv:1609.08124.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., & Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. arXiv preprint arXiv:1706.03762.

Wang, L., Li, Y., & Lazebnik, S. (2016). Learning deep structure-preserving image-text embeddings. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5005–5013).

Wu, Q., Shen, C., Wang, P., Dick, A., & Van Den Hengel, A. (2017). Image captioning and visual question answering based on attributes and external knowledge. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 40*(6), 1367–1381.

Wu, Q., Teney, D., Wang, P., Shen, C., Dick, A., & Van Den Hengel, A. (2017). Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding, 163*, 21–40.

Xu, J., Mei, T., Yao, T., & Rui, Y. (2016). MSR-VTT: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5288–5296).

Xu, X., Shen, F., Yang, Y., Shen, H. T., & Li, X. (2017). Learning discriminative binary codes for large-scale cross-modal retrieval. *IEEE Transactions on Image Processing, 26*(5), 2494–2507.

Yang, E., Deng, C., Liu, W., Tao, D., & Gao, X. (2017). Pairwise relationship guided deep hashing for cross-modal retrieval. In *Proceedings of the AAAI conference on artificial intelligence*.

Ye, M., Shen, J., Lin, G., Xiang, T., Shao, L., & Hoi, S. C. (2021). Deep learning for person re-identification: A survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 44*, 2872–2893.

Young, P., Lai, A., Hodosh, M., & Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics, 2*, 67–78.

Yu, X., Han, B., Yao, J., Niu, G., Tsang, I., & Sugiyama, M. (2019). How does disagreement help generalization against label corruption? In *International conference on machine learning, PMLR* (pp. 7164–7173).

Yu, Y., Ko, H., Choi, J., & Kim, G. (2017). End-to-end concept word detection for video captioning, retrieval, and question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3165–3173).

Yu, Y., Kim, J., & Kim, G. (2018). A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 471–487).

Zhao, Z., Yang, Q., Cai, D., He, X., & Zhuang, Y. (2017). Video question answering via hierarchical spatio-temporal attention networks. In *IJCAI* (pp. 3518–3524).

Zheng, W. S., Gong, S., & Xiang, T. (2012). Reidentification by relative distance comparison. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(3), 653–668.

Zhou, L., Xu, C., & Corso, J. J. (2018). Towards automatic learning of procedures from web instructional videos. In *Thirty-second AAAI conference on artificial intelligence*.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.