Hindawi
Complexity
Volume 2022, Article ID 2343707, 14 pages
https://doi.org/10.1155/2022/2343707

Research Article

PTF-SimCM: A Simple Contrastive Model with Polysemous Text Fusion for Visual Similarity Metric

Xinpan Yuan,[1] **Xinxin Mao**,[1] **Wei Xia**,[1] **Zhiqi Zhang**,[1] **Shaojun Xie**,[1] **and Chengyuan Zhang**[2]

[1]*School of Computer Science, Hunan University of Technology, Zhuzhou 412007, China*
[2]*College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China*

Correspondence should be addressed to Xinxin Mao; maoxinxin126@126.com

Received 23 February 2022; Revised 30 July 2022; Accepted 5 August 2022; Published 16 September 2022

Academic Editor: Zhen Zhang

Image similarity metric, also known as metric learning (ML) in computer vision, is a significant step in various advanced image tasks. Nevertheless, existing well-performing approaches for image similarity measurement only focus on the image itself without utilizing the information of other modalities, while pictures always appear with the described text. Furthermore, those methods need human supervision, yet most images are unlabeled in the real world. Considering the above problems comprehensively, we present a novel visual similarity metric model named PTF-SimCM. It adopts a self-supervised contrastive structure like SimSiam and incorporates a multimodal fusion module to utilize textual modality correlated to the image. We apply a cross-modal model for text modality rather than a standard unimodal text encoder to improve late fusion productivity. In addition, the proposed model employs Sentence PIE-Net to solve the issue caused by polysemous sentences. For simplicity and efficiency, our model learns a specific embedding space where distances directly correspond to the similarity. Experimental results on MSCOCO, Flickr 30k, and Pascal Sentence datasets show that our model overall outperforms all the compared methods in this work, which illustrates that the model can effectively address the issues faced and enhance the performances on unsupervised visual similarity measuring relatively.

1. Introduction

During the past decades, metric learning (ML) in computer vision (CV), also known as image similarity measurement, has been a fundamental problem in a variety of image applications, including image retrieval [1–3], face recognition [4–7], and visual search [8–11]. The goal of metric learning is to learn an embedding space, where the mapped feature vectors of similar instances are encouraged to be closer. At the same time, samples of different categories are pushed apart from each other [12–14]. In other words, the image similarity metric aims to estimate whether a given pair of images are similar or not.

Deep metric learning (DML) is a novel measure technique that combines deep learning (DL) with metric learning. With the recent rapid development of deep neural networks (DNN) in computer vision, DML has drawn growing attention and has become a mainstream metric learning method. Most previous deep metric approaches rely on supervised learning, meaning labels must be provided to the model along with input data. However, most data have not been labeled in the real world, and annotating a large-scale dataset is time-consuming and expensive. Consequently, learning effective visual-metric representations without human supervision is a crucial problem. The research on this problem is also known as Unsupervised Learning (UL). Self-Supervised Learning (SSL) can be regarded as a particular type of UL with a supervised form, where supervision is induced by self-supervised tasks rather than preset prior knowledge. SSL has delivered promising results in recent years, and a majority of mainstream approaches of it fall into one of two classes: generative or discriminative [15]. Generative techniques learn to reconstruct the original input [16, 17], and discriminative

(a) (b) (c)

Figure 1: Image examples. (a) and (b) are randomly selected from Tumblr and Twitter, respectively. (c) is randomly picked from Amazon Marketplace.

methods learn representations using objective functions. Discriminative ones are similar to the supervised learning approaches but require specific pretext tasks [18, 19] in which both inputs and labels are derived from an unlabeled dataset. Recently, discriminative approaches based on contrastive learning have achieved outstanding results in unsupervised visual representation. Those approaches [20–22] show significant potential to close the performance gap with supervised methods or even achieve the same results in specific downstream tasks and datasets. Therefore, it is of great research value to introduce SSL into image similarity measurement.

In addition, the current image similarity measurement techniques based on deep metric learning frequently focus on the image itself merely without further utilizing the information of other modalities correlated to the image. Nevertheless, pictures usually do not appear independently in the real world, yet with other modal information such as text and sound, especially in social media. For example, when people browse images in social media applications like Twitter and Tumblr, the pictures are always accompanied by text descriptions. Figures 1(a) and 1(b) are images selected randomly from Tumblr and Twitter, respectively, and both images have related sentences. Thus, when searching for relevant images in social apps, the results would be better if we make good use of the text information that appeared with the pictures. Besides, online shopping malls like Amazon and eBay often provide a text description of the product when listing it. Figure 1(c) shows an image of a sweater jacket randomly picked from Amazon Marketplace, and a detailed product description can be found below the image. In this way, we can use relevant description information to obtain more accurate product candidates when exploring similar products in online shopping. From the above two application scenarios, it can be concluded that the textual information that can make the image more specific will be lost if we only focus on the image itself in image similarity measuring.

However, a text sometimes appears ambiguous in real life. For example, "he left the bank" has two interpretations. One is that he is now far away from the bank, like "he left the bank five minutes ago." Another is that he is no longer working at the bank, like "he left the bank five years ago." Namely, each of those ambiguous sentences can map to more than one point in the embedding space. Nevertheless, an injective model [23], a one-to-one mapper, can embed merely one point from an ambiguous text input sample. That is to say, the semantics resulting from the injective model will likely deviate from the ones we expect. Therefore, if these ambiguous sentences are used in the image similarity metric directly, this will not only hardly improve the accuracy of the algorithm but also have a negative impact on the results.

To address the above issues, we propose a Simple Contrastive Model with Polysemous Text Fusion (PTF-SimCM) for visual similarity metric in this work. The architecture of PTF-SimCM consists of two branches and a middle cross-modal encoder. Of all the current best contrastive learning models, SimSiam [24] is the simplest without negative samples and no need for large batch size. Hence, the two branches in our model adopt the same asymmetric contrast structure as SimSiam for solving the trouble of visual-metric representations without human supervision. It contains upper and lower subnets with shared weights. Besides, we use a cross-modal model for encoding input sentences rather than a unimodal text model to reduce the computation of late features fusion and improve training productivity. For the problem in polysemy text fusion, we employ the pretrained Sentence PIE-Net [23] as our cross-modal encoder and fuse the output embeddings with image features. Moreover, to make the image similarity measurement simpler and more efficient, PTF-SimCM straight learns an embedding space where distances directly correspond to a measure of image similarity.

In short, our contributions are summarized as follows:

(i) We propose a novel method based on the contrastive learning structure for unsupervised visual similarity measuring.

(ii) For better model performance, we further pay attention to information on the textual modality correlated to the image by effectively combining multimodal fusion with the contrastive structure.

(iii) To address the problem caused by polysemous sentences and improve the efficiency of late feature fusion, we exploit a specific cross-modal net to process input text modality rather than a standard unimodal text sequence model.

(iv) We obtain competitive results compared with the methods which do not consider textual information in image similarity measuring or cannot process the unlabeled training data on MSCOCO [25], Flickr 30k [26], and Pascal Sentence datasets [27].

The rest of this paper is organized as follows. Section 2 reviews and discusses the related works in this area. Section 3 characterizes the proposed PTF-SimCM model in detail, followed by the implementation details, performance comparison experimental results, and ablation study in Section 4. Conclusions are in Section 5.

2. Related Works

This section will briefly review some of the most relevant research to our work. The structure of this section is as follows. Section 2.1 will review the recent research on deep metric learning, followed by the recent study on contrastive learning in Section 2.2. The SimSiam model [24], as same asymmetric contrastive structure as our approach, is described in Section 2.3. The recent examinations on cross-modal representation are in Section 2.4.

2.1. Deep Metric Learning. Deep metric learning (DML) is a specific type of metric learning that aims to measure the similarity between input data samples by mapping data to an embedding space in which similar instances are close together, and dissimilar data are far apart. One of the fundamental ideas where explicit metric learning is performed is the Siamese network model [28, 29]. Siamese network, with a contrastive loss [30], is asymmetric neural network architecture consisting of two identical subnetworks that share the same parameters. The contrastive loss, as shown in equation (1), is a classic loss function and is one of the most straightforward and intuitive training objectives for metric learning.

$$\mathscr{L}_{\text{contrast}} = \mathbf{1}_{y_1=y_2}\mathscr{D}^2(x_1, x_2) + \mathbf{1}_{y_1\neq y_2}\max\left(0, \alpha - \mathscr{D}^2(x_1, x_2)\right), \quad (1)$$

where x_1, x_2 are input samples and y_1, y_2 are their corresponding labels. $\mathbf{1}_{\text{condition}}$ denotes the identical function that is equal to 1 indicating that the data pairs are similar or positive and $y_1 \neq y_2$ shows that the pairs are dissimilar or negative. The parameter α is the margin threshold between positive and negative.

Triplet loss [5] is another popular and widely used loss function for metric learning. Let x_a, x_p, x_n be samples from the dataset and y_a, y_p, y_n are their corresponding labels and $y_a = y_p, y_a \neq y_n$. Usually, x_a is called anchor, x_p is called positive sample, and x_n is called negative sample. The loss function is defined as

$$\mathscr{L}_{\text{triplet}} = \max\left(0, \mathscr{D}^2\left(x_a, x_p\right) - \mathscr{D}^2\left(x_a, x_n\right) + \alpha\right), \quad (2)$$

where α represents the margin threshold between $\mathscr{D}^2(x_a, x_p)$ and $\mathscr{D}^2(x_a, x_n)$. It needs to be pointed out that negative samples mining, on which we sample such triplets x_a, x_p, x_n satisfying $\mathscr{D}(x_a, x_n) < \mathscr{D}(x_a, x_p) + \alpha$, plays a key role in the effect of this loss function.

Based on contrastive loss and triplet loss, many improved methods have emerged. N-Pair Loss [31] generalizes triplet loss by allowing joint comparison among more than one negative example and reduces the computation by an efficient batch construction strategy. MS Loss [32] can fully consider three similarities for pair weighting.

Other commonly used DML methods are non-contrastive. The main idea of SphereFace [4] is enforcing the class centers to be at the same distance from the center by mapping them to a hypersphere. Furthermore, it employs angular distance with angular margin to measure distance. CosFace [7] proposes a more straightforward yet more effective method to define the margin. ArcFace [6] is very similar to CosFace. However, instead of defining the margin in the cosine space, it defines the margin directly in the angle space. The SoftTriple Loss [33] takes a different strategy by expanding the weight matrix to have multiple columns per class, providing more flexibility for modeling class variances.

2.2. Contrastive Learning. Self-Supervised Learning (SSL) is a technique to derive information from unlabeled data itself directly by formulating specific predictive tasks, and contrastive learning, based on contrastive constraints, is one of the most popular methods of self-supervision. Deep InfoMax [34] was presented to learn representations of images by leveraging the local structure present in an image. Structurally, SimCLR [15] is a simple contrastive learning method that utilizes negative samples and a sizeable mini-batch up to 4096 to work better. To get around the problem of an enormous number of negative examples, MoCo [20] maintained a large queue of them instead of updating the negative encoder by backpropagation. BYOL [35] assumed that it is still possible to train an SSL model with outstanding results without negative samples. It adopted an asymmetric network structure with momentum updates and a stop-gradient operation to avoid collapse. SimSiam [24] removed the target encoder updated by the momentum based on BYOL. Barlow Twins [36] started from the perspective of embedding rather than from samples and proposed improving the representation ability of features by allowing features of various dimensions to represent different information as much as possible.

2.3. SimSiam. SimSiam [24] has been one of the most popular contrastive learning methods recently, and the architecture, as shown in Figure 2, uses two views x_1 and x_2 randomly augmented from the same image x as input. Two encoder modules, consisting of a backbone and a projection [15], extract image features from two views, respectively, and the encoder shares weights between the two views. Then only the left branch of the model takes the output and feeds it to a predictor module h . The output in the right branch does not

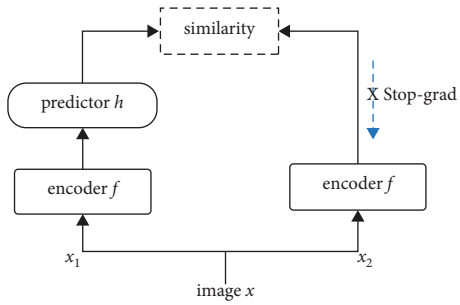


Figure 2: The architecture of SimSiam [24].

require any treatment. Then the cosine similarity of results of two branches is calculated.

"Stop-grad" denotes that the right branch does not update gradients during training. The strategy to stop updating parameters of the right branch and the asymmetric structure of the model are both for preventing model collapse, and the experiments of the paper [24] show that the above methods, especially stop-gradient, can effectively avoid the issue of model collapsing.

2.4. Cross-Modal Representation. Cross-modal representation has played an indispensable role in representation learning in recent years, and the goal is to build embeddings using information from multiple modalities. Fusing information from the different domains into unified embeddings is one of the wide implementations of the cross-modal representation approaches. In this field, [37] presented that through unifying local document context and global visual context to learn better visually grounded word embeddings. Based on *word2vec* [38], Kottur et al. [39] proposed adding images as inputs to the training of it so that the resulting word vectors have visual semantic information.

Building embeddings for different modalities in a common semantic space has been another popular way over the past few years. This method allows the model to compute cross-modal similarity, which can be further used for downstream tasks, such as cross-media retrieval [40–42]. Ba et al. [43] presented a model that can classify unseen categories from their textual description by cross-modal similarity in Zero-Shot Learning (ZSL). Huang et al. [44] presented the cross-media model to transfer valuable knowledge in existing data to new data. On the basis of cross-modal representation, Song et al. [23] further considered the ambiguity of instances, thereby improving the accuracy of visual semantics.

3. Proposed Method

In this section, we propose a Simple Contrastive Model with Polysemous Text Fusion (PTF-SimCM) for visual similarity metric. The structure of this section is organized as follows. In Section 3.1, we describe the problems. We provide an overall description of the model in Section 3.2. Section 3.3 characterizes the cross-modal encoder, followed by the multimodal fusion and projector in Section 3.4. Predictor and inference are in Section 3.5 and Section 3.6, respectively. Section 3.7 presents the algorithm details.

Table 1: Notation list of the proposed method.

Symbol	Description
x	Input image
c	Textual description of image
t, t'	Image augmentation
f_θ	Image encoder
$\mathscr{P}$	Cross-modal encoder
K	The number of cross-modal embeddings
g_θ	Multimodal projector module
q_θ	Predictor module
u	Fused feature
z	Metric embedding
p	Vector output from predictor
sg	Stop-gradient operation

3.1. Problem Formulation. The image similarity metric aims to estimate whether a given pair of images are similar or not, and the problem to be addressed in this study is how to measure image similarity and improve the performance of the algorithm in scenarios where there are only a large number of unlabeled images. Firstly, learning effective visual-metric representations without human supervision is a crucial problem as plenty of images that are not human-labeled in our daily lives exist. Moreover, images in our daily social media always appear with correlated text descriptions. Therefore, it is a meaningful study to further consider and utilize textual modalities among image similarity metrics. Unfortunately, the textual description is sometimes ambiguous, so solving this polysemy is another critical problem.

The proposed model combines self-supervised contrastive learning, cross-modal representation, and modalities fusion to address the above issues and advance the performance. Furthermore, our model learns the metric embeddings directly, the distance of which corresponds to image similarity. The notation list of the proposed method is shown in Table 1.

3.2. Overview of PTF-SimCM. PTF-SimCM employs the same contrastive learning structure as SimSiam. It aims to learn an embedding space where the cosine similarity of embeddings directly corresponds to image similarity: images with similar content have small distances, and images with distinct content have large distances. In contrast, the previous approach [15, 24, 35] of contrast learning was to learn a representation that can then be used for downstream tasks. The architecture of PTF-SimCM, composed of two branches of neural networks and a middle cross-modal encoder (CME), is presented in Figure 3. The upper branch is defined by a set of weights θ and is composed of three stages: an encoder f_θ, a multimodal projector g_θ, and a predictor q_θ. The lower branch is the same as the upper branch, except that there is no final predictor. The CME $\mathscr{P}$ is a pretrained cross-modal network with frozen parameters for reducing the computation. Alternatively, as if computing resources

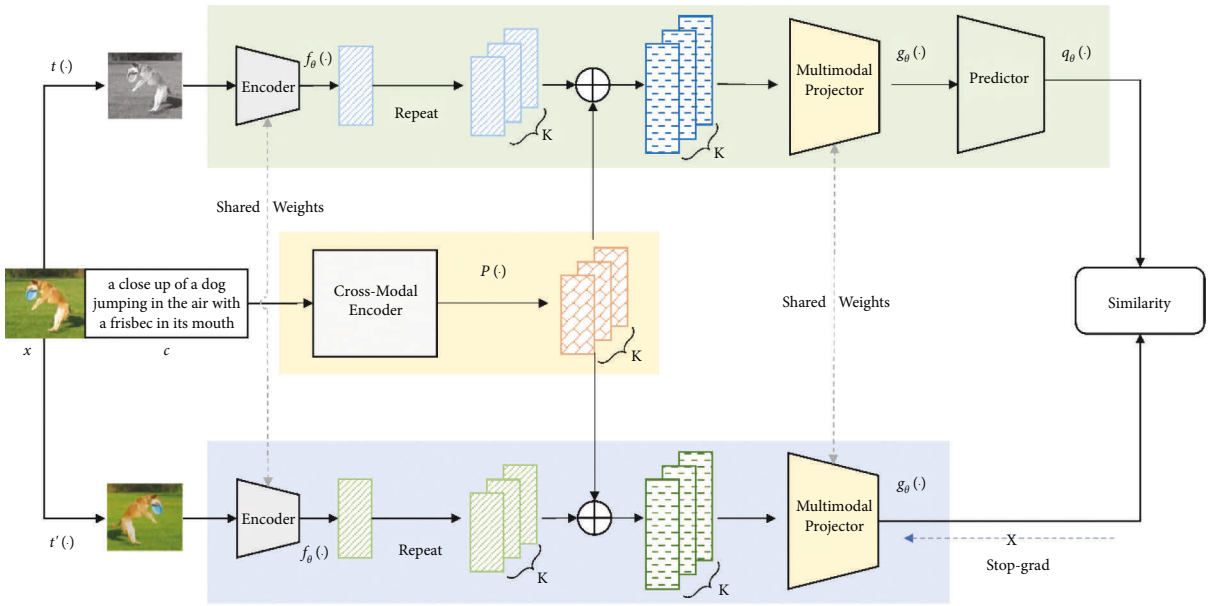


Figure 3: The architecture of PTF-SimCM.

are not a concern, fine-tuning the parameters in training may be a better option.

Given a set of image-text pairs $\mathscr{D}$, a pair $(x, c) \sim \mathscr{D}$ sampled uniformly from $\mathscr{D}$, and two image augmentations t and t', image augmentation is a technique that generates similar but distinct views after a series of random changes to the input images. The model generates two augmented views $x_1 \triangleq t(x)$ and $x_2 \triangleq t'(x)$ to image x as the upper branch and lower branch input, respectively, and takes description c of the image as input of CME. The encoder network f is actually a backbone (e.g., ResNet [45]) used to extract image features. CME is used to process text description c. To make the text learn the semantic information of the image before fusing with it so that the subsequent fusion and projection get more uncomplicated, and to improve the efficiency of model training, we directly use a pretrained cross-modal representation model instead of a unimodal text sequence net as our CME module, which can embed the sentence to shared visual-textual space. In addition, considering the ambiguity of the sentence, CME further transforms the input into K embeddings with different semantics. The embedding, output from the middle tube, is a visually grounded word representation, and we call it a cross-modal embedding. The image feature output from the encoder f_θ and each of the K embeddings are fused through simple concatenating, as seen in the following equation:

$$u^i = f_\theta(x) \oplus (\mathscr{P}(c))_i, \tag{3}$$

where u^i denotes the ith fusion feature and $(\mathscr{P}(c))_i$ is the ith cross-modal embedding. The multimodal projector g_θ is a four-layer multilayer perceptron (MLP) and it maps fused representations to the specific metric space. Then, g_θ is followed by an MLP predictor in the upper side path. The predictor transforms the output of the upper side multimodal projector and matches it to the lower branch output. Stop-gradient as same as [24] is adopted in the lower branch; that is, parameters of network on lower side path will not be updated by backpropagation. Various experiments in [35] have shown that predictor and stop-gradient are the keys to preventing the model from falling into a collapsed solution. Negative cosine similarity, shown in (4), is used as the loss between output embeddings of the upper and lower branch networks.

$$\mathscr{S}\left(p_1^i, z_2^i\right) = -\frac{p_1^i}{\left\|p_1^i\right\|_2} \cdot \frac{z_2^i}{\left\|z_2^i\right\|_2}, i \in [1, 2, \ldots, K], \tag{4}$$

where $\|\cdot\|_2$ is ℓ_2-norm, $p_1^i \triangleq q(g(u_1^i))$ and $z_2^i \triangleq g(u_2^i)$, and u^i is one of the mixed features. Following [24], we define a symmetrized loss as

$$\begin{aligned}\mathscr{L} = &\frac{1}{2K}\sum_{i=1}^{K}\mathscr{S}\left(p_1^i, sg\left(z_2^i\right)\right) \\ &+\frac{1}{2K}\sum_{j=1}^{K}\mathscr{S}\left(p_2^i, sg\left(z_1^j\right)\right) \quad i, j \in [1, 2, \ldots, K],\end{aligned} \tag{5}$$

where sg denotes stop-gradient operation and then the final output z of the lower branch network is denoted as $sg(z)$. Loss function $\mathscr{L}$ is used to measure the similarities between x_1 and x_2, which are two augmentation views with textual content description. The function is defined for each view-text pair, and total loss is averaged value over all pairs.

3.3. Cross-Model Encoder Module. For the model to perform feature fusion and projection more efficiently, we use a cross model to handle the sentence input rather than a unimodal text model. Moreover, a sentence sometimes appears ambiguous in real life. That is to say, each of those captions can map to more than one point in the embedding space. Unfortunately, a one-to-one mapper can merely find one

single point, and there is no doubt that it has a negative impact on subsequent applications.

PVSE (Polysemous Visual-Semantic Embedding) [23] is a visual-semantic embedding model which could more effectively address the partial cross-domain association issue and ambiguous instances issue compared with other approaches. The architecture of PVSE is composed of two submodels called PIE-Net (Polysemous Instance Embedding Network), which can extract K embeddings of each sentence instance based on the different meanings. To address the issue of ambiguity in input sentences, we adopt Sentence PIE-Net pretrained through PVSE architecture as the cross-modal encoder.

The sentence encoder is composed by GloVe [46] and a bidirectional GRU (Bi-GRU) [47]. A sentence with L words is converted by the pretrained GloVe into L 300-dim vectors as local features $\Psi(x) \in \mathbb{R}^{L\times 300}$. Then they are taken into a Bi-GRU with H hidden units. The output of the final hidden layer as global features $\phi(x) \in \mathbb{R}^{H}$. The local feature transformer transforms $\Psi(x)$ into K locally guided representations. More specifically, $\Psi(x)$ is first fed into a multihead self-attention [48, 49] module implemented by a two-layer perceptron and K attention maps are obtained. Given local features $\Psi(x) \in \mathbb{R}^{B\times D}$, K attention maps $\alpha \in \mathbb{R}^{D\times B}$ are computed as follows:

$$\alpha = \text{softmax}\left(w_2 tanh\left(w_1 \Psi(x)^T\right)\right), \tag{6}$$

where $w_2 \in \mathbb{R}^{K\times A}$, $w_1 \in \mathbb{R}^{A\times D}$, and $A = D/2$. Then, local features are multiplied by the attention map and the result is taken into a nonlinear module to obtain K locally guided features $\Gamma \in \mathbb{R}^{K\times H}$:

$$\Gamma(x) = \sigma\left((\alpha\Psi(x))w_3 + b_3\right), \tag{7}$$

where $w_3 \in \mathbb{R}^{D\times H}$, $b_3 \in \mathbb{R}^{H}$, and σ denotes sigmoid function.

Finally, the fusion block obtains the final K embeddings by combining global features and locally guided features. The K embedding vectors $z \in \mathbb{R}^{K\times H}$ are computed as

$$z = LN\left(\Phi(x) + \Gamma(x)\right), \tag{8}$$

where $\Phi(X) \subset \mathbb{R}^{K\times H}$ represents K repetitions of $\phi(x)$ and LN denotes layer normalization [50].

3.4. Multimodal Fusion and Projector. As the cross-modal module is used to process text modality in the previous step, the obtained embeddings have learned a certain amount of image semantic information. Consequently, the fusion method "$\oplus$" adopts a simple concatenate operation. Given feature $y \in \mathbb{R}^{M}$ of view augmented from input image, embedding $m \in \mathbb{R}^{N}$ output from cross-modal block, it computes fused vector u:

$$u = y \oplus m = [y, m] \in \mathbb{R}^{M+N}. \tag{9}$$

This simple concatenation operation does not establish a special semantic connection between different features, so it relies on subsequent network layers to adapt it. That is to say, the multimodal projector is not only a mapper but also a fusion adaptor.

The multimodal projector is a four-layer perceptron with batch normalization [51] applied to each fully connected layer, including the final output layer. The previous three-layer perceptron has no biases and takes the ReLU as the activation function. One of the multimodal metric embeddings z is computed as

$$\begin{aligned} h^{(l)} &= ReLU\left(BN\left(W^{(l)}h^{(l-1)}\right)\right) \\ z &= BN\left(W^{(l+1)}h^{(l)} + b^{(l)}\right), \end{aligned} \tag{10}$$

where $h^{(l)}$, $W^{(l)}$, and $b^{(l)}$ represent the hidden state victor, weight matrix, and bias, respectively, of l-layer and BN denotes batch normalization.

3.5. Predictor. The predictor module is only applied to the upper branch to make the architecture asymmetric, as in [24, 35], and this structure can prevent collapse to a certain extent. Like the multimodal projector, the predictor block is also a multilayer perception yet with two layers. The batch normalization is applied to hidden layer which has no bias. The output fully connected layer does not have BN and activation function. One of the final outputs p is given:

$$p = W_2\left(ReLU\left(BN\left(W_1 z\right)\right)\right) + b_2. \tag{11}$$

In the above formula, W_1 is weight matrix of hidden layer, and W_2 and b_2 are parameters of output layer.

3.6. Inference. At test time, we use the inference model obtained by removing the upper branch and the loss function from the PTF-SimCM to measure similarity. Two image-text pairs A' and B' are given to measure the similarity. They are fed into the inference model and two K embedding vectors $A^i = [a_1^i, a_2^i, \ldots, a_n^i], i \in [1, \ldots, K]$ and $B^j = [b_1^j, b_2^j, \ldots, b_n^j], j \in [1, \ldots, K]$, respectively, are got. Then we find the best matching pair by comparing the cosine distances between all K^2 combinations of embeddings. At this point, the distance is as the similarity between A' and B'. The architecture and processing process in inference is shown in Figure 4.

3.7. Optimization Algorithm. The model adopted mini-batch gradient descent to update parameters, and the specific steps and details of model training and optimization are shown in Algorithm 1.

4. Experiment

4.1. Datasets. We first introduce the three image-text retrieval datasets, that is, MSCOCO [25], Flickr 30k [26], and Pascal Sentence [27]. Figure 5 shows part data samples from the three datasets.

MSCOCO is one of the most prevalent cross-modal retrieval datasets in recent years. It contains 123,287 images and 616,767 descriptions. Every image contains roughly 5 text descriptions on average. This dataset is officially split into 113,287 images as training data, 5,000 images as validation data, and 5,000 images as testing data. As our

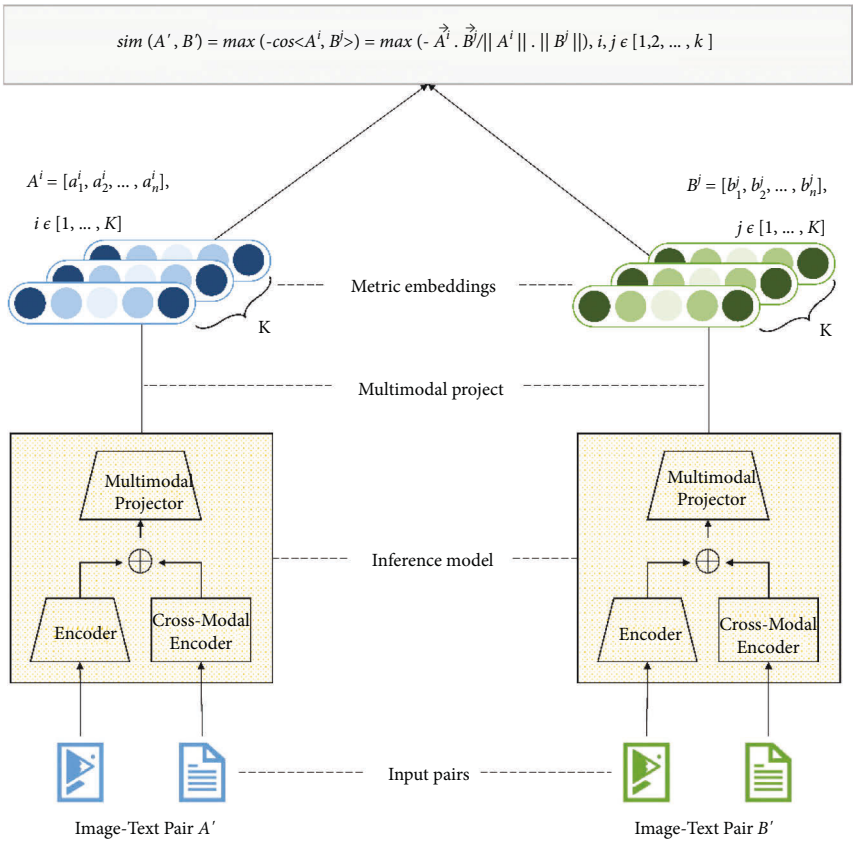
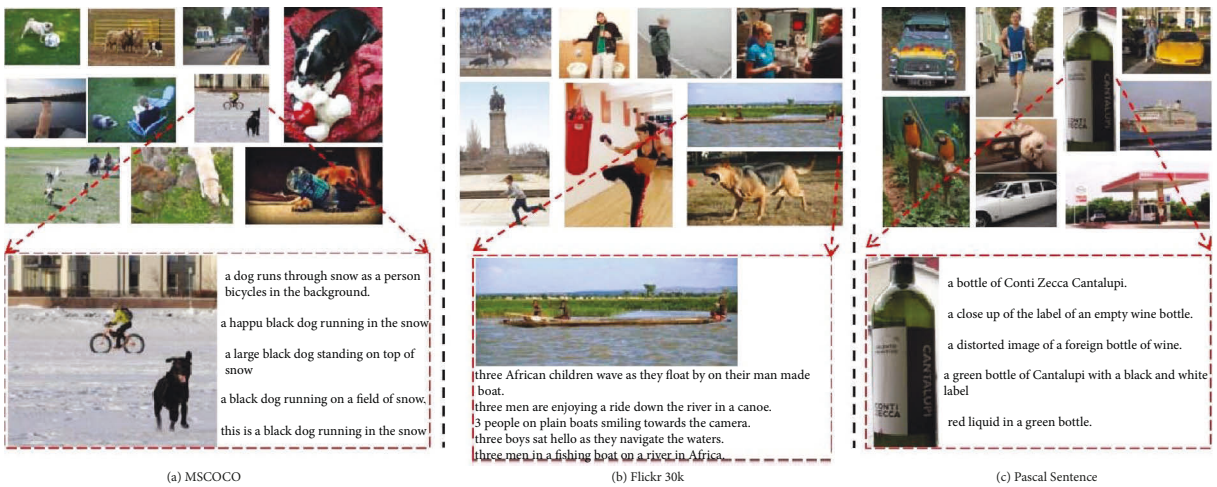


Figure 4: The architecture and processing process in inference.

Input:
$\mathcal{D}, \mathcal{T}, \mathcal{T}'$ set of images with description and distributions of transformations;
$\theta, f_\theta, g_\theta$ and q_θ initial parameters, encoder, multimodal projector, predictor;
$\mathcal{P}$ cross-modal encoder;
K the number of cross-modal embeddings;
opt optimizer, updates parameter using the loss gradient;
M and N total number of optimization steps and batch size;
$\{\eta_m\}_{m=1}^{M}$ learning rate schedule;

(1) **for** $m = 1$ to M **do**
(2) $\quad \mathcal{B} \leftarrow \{\mathcal{D}\}_{i=1}^{N}$; //sample a batch of N image-text pairs
(3) $\quad$ **for** $x_i, c_i \in \mathcal{B}$ **do**
(4) $\quad\quad t \sim \mathcal{T}$ and $t' \sim \mathcal{T}'$; //sample image transformations
(5) $\quad\quad y_1 \leftarrow f_\theta(t(x_i))$ and $y_2 \leftarrow f_\theta(t'(x_i))$;
(6) $\quad\quad$ **for** $k = 1$ to K **do**
(7) $\quad\quad\quad z_1^k \leftarrow (g_\theta(y_1) \oplus \mathcal{P}(c_i)^k)$ and $z_2^k \leftarrow (g_\theta(y_2) \oplus \mathcal{P}(c_i)^k)$;
(8) $\quad\quad\quad p_1^k \leftarrow q_\theta(z_1^k)$ and $p_2^k \leftarrow q_\theta(z_2^k)$;
(9) $\quad\quad\quad l^k \leftarrow -1/2\,(p_1^k/\|p_1^k\|_2 \cdot z_2^k/\|z_2^k\|_2 + p_2^k/\|p_2^k\|_2 \cdot z_1^k/\|z_1^k\|_2)$;
(10) $\quad\quad$ **end**
(11) $\quad\quad l_i \leftarrow 1/K \sum_{k=1}^{K} l^k$; //compute the total loss
(12) $\quad$ **end**
(13) $\quad \delta\theta \leftarrow 1/N \sum_{i=1}^{N} \partial_\theta l_i$; //compute the total loss gradient
(14) $\quad \theta \leftarrow opt(\theta, \delta\theta, \eta_m)$; //update parameters
(15) **end**
(16) **return** $f(\cdot), g(\cdot)$

Algorithm 1: PTF-SimCM's main learning algorithm.



(a) MSCOCO (b) Flickr 30k (c) Pascal Sentence

Figure 5: Sample images in the three datasets.

proposed model requires input that includes both images and sentences, we disuse the testing set. In order to be able to test usually, we divided the validation set into 100 classes by the K-means algorithm [52].

Flickr 30k is another prevalent and large-scale image captioning dataset. It contains 31,783 images that are collected from Flickr. Every image is annotated with five textual descriptions. The average sentence length reaches 10.5 words after removing rare words. We adopt the protocol in [53] to split the dataset into 1,000 test images, 1,000 validation images, and 29,783 training images. In order to have more training data, we merged the validation set into the training set and got the 30783 images final training set. Again, we split the test set into 20 classes using the K-means algorithm.

Pascal Sentence dataset also is a widely used dataset for cross-media retrieval. It is a subset of Pascal Voc and contains 1,000 image-text pairs from 20 categories. So each category includes 50 pairs, and each pair has an image and several sentences. The number of samples in this dataset is not large enough to be used as training data for our model. Consequently, we only use this dataset as our test set.

4.2. Evaluation Metrics. We use three metrics, Recall at k (Recall@k), R-Precision, and Mean Average Precision at R (MAP@R), to evaluate the performance of models. According to [54], R-Precision and MAP@R are the most suitable metrics to measure model performance in metric learning. As Recall has been the most widely used measure in the past few years in metric learning, we also used it to evaluate metrics for comprehensive comparisons to other methods.

Recall@k in metric learning is defined as follows: first, for each sample in the dataset, get k-nearest neighbors. If at least 1 of those nearest neighbors is a match, then the sample gets a score of 1. Otherwise, it scores 0. Recall@k is the average of score of all query sample:

$$\text{Recall@}k = \frac{1}{n}\sum_{i=1}^{n}(\text{score}). \tag{12}$$

Given a query, let R be the total number of references that are the same class as the query. **R-Precision** is computed as

$$\text{R} - \text{Precision} = \frac{r}{R}, \tag{13}$$

where r is the number of R-nearest references that are the same class as the query.

MAP@R is Mean Average Precision with the number of nearest neighbors for each sample set to R. For a single query,

$$\begin{aligned}\text{MAP@}R &= \frac{1}{R}\sum_{i=1}^{R}P(i)\\ P(i) &= \begin{cases}\text{precision at i} & \text{if the ith retrieval is correct}\\ 0 & \text{otherwise}\end{cases}.\end{aligned} \tag{14}$$

4.3. Implementation Details. The model proposed in this study employs the same contrastive learning structure as SimSiam [24]. We used momentum SGD as the optimizer and used initial learning rate is $lr = 0.05$. Learning rate is computed as $lr \times \text{BatchSize}/256$, which followed a cosine decay schedule [55], and the specific parameter settings are shown in Table 2. In addition, to acquire significant computational speedup and less GPU memory consumption, we utilize mixed-precision training [56] in the model training.

4.4. Baseline Methods. To verify the performance and validity of the model proposed in this study, we select four models that are quite popular in the relevant fields and show good performance in past studies as the baselines. In terms of supervised metric learning, we compare the following models: Siamese Network [29], Triplet Network [5], SoftTriple [33], MemVir [57], and Label Relaxation [58]. The name of the selected Siamese Network is SimNet [29], and it is trained on pairs of positive and negative images like the conventional Siamese model. The improvement of this model is that it uses a novel online pair mining strategy

Table 2: Hyperparameters set in experiments.

Parameters	Description	Performance comparison	Ablation study
lr	Initial learning rate	0.05	0.05
wd	Weight decay	$1e-4$	1e-4
η	Momentum	0.9	0.9
d_{view}	Dimension of image view features	2048	2048
d_{cross}	Dimension of cross-modal embedding	1024	{512, 1024, 2048}
d_{metric}	Dimension of metric embedding	{64, 128}	{64, 128, 256, 512}
K	Number of cross-model embedding	2	{1, 2, 3, 4}
l_{proj}	Layers of multimodal projector	4	4
l_{pred}	Layers of predictor	2	2

based on curriculum learning and adopts a multiscale convolutional neural network (CNN) as the feature extractor. FaceNet [5] was initially used to distinguish face images and directly learn a mapping from input face images to a metric space where distances directly correspond to a measure of face similarity. It used a triplet architecture with triplet loss and also has a good performance in the similarity detection of other nonface images. SimNet and FaceNet are contrastive-based DML approaches that use the specific loss function that directly pulls together the embeddings of samples with the same label and pushes away the embeddings of dissimilar samples. SoftTriple is representative of noncontrastive approaches and can learn the embeddings without the sampling stage by lightly increasing the size of the last fully connected layer. MemVir and Label Relaxation are two state-of-the-art methods in metric learning. MemVir exploits the virtual classes by maintaining memory queues to utilize augmented information for training and alleviates a strong focus on seen classes for better generalization. To improve metric learning performance, Label Relaxation employs a relaxed contrastive loss based on embedding transfer to enable more crucial pairs to contribute more to training. All of those five methods have intensely outstanding performance in the presence of a large number of labeled datasets. With regard to unsupervised visual-metric representation, we compare our method to SimSiam [24], the simplest SSL model without negative samples and no need for a large batch size. It is one of the best SSL models, and the contrastive structure of our model refers to it.

4.5. Performance Comparison. In this section, we conduct a series of performance comparison experiments. In order to simulate the unsupervised scene where we cannot obtain relevant labeled datasets, we use an additional dataset Mini ImageNet (a subset of ImageNet [59]) to train the above models. For unsupervised metric learning, as the contrastive structure of our model is based on SimSiam, we employ it as the baseline. SimSiam generally uses the feature extraction module without the projector block for downstream tasks. As our model has an additional mapping module in structure, in order to make the comparison fairer, we add a model "SimSiam + proj" that retains the projector for comparison. The performances of the proposed model and baselines on MSCOCO dataset are illustrated in Table 3.

From Table 3, we can observe that our model achieves positive results in experiments. The proposed model with setting of $K = 2$ and $d = 64$ (dimensions of metric embedding = 64) achieves significant improvement in terms of MAP@R with at least 0.2, and PTF-SimCM with $d = 128$ obtains the best overall performance in recall rate. SoftTriple has the best R-Precision of 0.307 and the highest value of Recall@1. MemVir gets the second-highest MAP@R value. It is noteworthy that the performance of Label Relaxation is inferior to that of SoftTriple. In our judgment, the primary cause is that Label Relaxation is an embedding transfer model which relies on a well-knowledged source embedding model. However, it can hardly obtain a well-trained source model in the unsupervised scenario. We can only train on a labeled dataset from another domain and then test on data from the target domain. Consequently, the Label Relaxation lost its advantage in this particular scenario and achieved a nonremarkably good performance in the above experiments. In the same way, it is also tricky for MemVir to exert its advantages when the training and test data are not in the same domain. Compared to SimSiam and "SimSiam + proj," our model achieves better performance in all three metrics. Furthermore, it is worth noting that the overall results of the experiments are not very superior. One of the key factors is that MSCOCO is not a dataset for classification or image retrieval. Consequently, there is a relatively large intraclass variation between the positive samples of this dataset. Besides, it also depends on the effect of the cluster method.

With the purpose of making the experimental comparison results more comprehensive, we conducted experiments on the Flickr 30k dataset. The results are summarized in Table 4. We can see that our proposed method with setting of $K = 2$ and $d = 64$ got the best MAP@R scores, and it is 0.3 percentage points higher than the second place. The proposed model with $k = 2$ and $d = 128$ achieved the best Recall@8 value. SoftTriple obtained the best R-Precision of 0.296, the highest value of Recall@1, the highest value of Recall@4, and the second-highest MAP@R value. MemVir got 69.1% on Recall@2, the highest value. In general, the performance of our method on Flickr 30k is close to those of supervised methods SoftTriple and MemVir without exceeding them by much. The main reason is that the training data is not enough. Moreover, the classes of test sets were still divided by K-means, and this played an important role in the accuracy of the experiment.

Table 3: Accuracy on MSCOCO. K and d in PTF-SimCM denote the number of embeddings and dimensions of metric embedding, respectively.

Methods	Recall@k				R-Precision	MAP@R
	$k=1$	$k=2$	$k=4$	$k=8$		
Siamese	0.571	0.679	0.776	0.834	0.296	0.187
Triplet	0.557	0.656	0.762	0.823	0.286	0.172
SoftTriple	**0.583**	0.686	0.789	0.843	**0.307**	0.190
Label Relaxation	0.532	0.657	0.745	0.829	0.265	0.164
MemVir	0.579	0.693	0.794	0.840	0.296	0.192
SimSiam	0.416	0.564	0.692	0.791	0.216	0.117
SimSiam + proj	0.461	0.585	0.681	0.775	0.175	0.112
Ours ($K=2, d=64$)	0.573	0.689	0.793	0.832	0.298	**0.202**
Ours ($K=2, d=128$)	**0.583**	0.702	0.801	0.846	0.261	0.182

Table 4: Accuracy on Flickr 30k. K and d denote the number of embeddings and dimensions of metric embedding, respectively.

Methods	Recall@k				R-Precision	MAP@R
	$k=1$	$k=2$	$k=4$	$k=8$		
Siamese	0.551	0.689	0.780	0.847	0.290	0.186
Triplet	0.530	0.664	0.753	0.823	0.281	0.164
SoftTriple	**0.559**	0.683	**0.781**	0.842	**0.296**	0.190
Label Relaxation	0.514	0.663	0.748	0.831	0.257	0.162
MemVir	0.543	**0.691**	0.769	0.849	0.287	0.189
SimSiam	0.403	0.502	0.692	0.819	0.203	0.113
SimSiam + proj	0.433	0.549	0.714	0.791	0.164	0.108
Ours ($K=2, d=64$)	0.532	0.667	0.758	0.833	0.290	**0.193**
Ours ($K=2, d=128$)	0.546	0.669	0.772	**0.854**	0.263	0.181

Table 5: Accuracy on Pascal Sentence. K and d denote the number of embeddings and dimensions of metric embedding, respectively.

Methods	Recall@k				R-Precision	MAP@R
	$k=1$	$k=2$	$k=4$	$k=8$		
Siamese	0.581	0.709	0.784	0.860	0.326	0.218
Triplet	0.560	0.671	0.773	0.842	0.318	0.202
SoftTriple	0.598	0.712	0.802	0.864	**0.343**	0.228
Label Relaxation	0.579	0.687	0.785	0.849	0.308	0.198
MemVir	0.593	0.709	0.796	0.876	0.337	0.232
SimSiam	0.443	0.585	0.716	0.821	0.259	0.147
SimSiam + proj	0.491	0.606	0.702	0.789	0.217	0.138
Ours ($K=2, d=64$)	0.592	0.714	0.806	0.864	0.336	**0.240**
Ours ($K=2, d=128$)	**0.616**	0.719	0.801	0.870	0.299	0.206

In order to evaluate the universality of the proposed model, we also conducted experiments on the Pascal Sentence dataset. In this set of experiments, MSCOCO is still used as training data, and Pascal Sentence is merely used as the test data for models. The results are shown in Table 5.

The above results, in which the proposed model has better overall performance, are similar to those shown in Table 3 and 4. The validation values from Table 5 have improved, yet they are still not excellent. We believe the reason behind that is that there is not enough training data for a generic image metric model. Accordingly, it did not learn the distribution closer to the ground truth. Overall, our model performs favorably in the experimental results, outperforming recent popular supervised learning methods under certain conditions, and, especially compared to SimSiam with the same contrast structure, our model improves significantly.

4.6. Ablation Study. To demonstrate the effectiveness of some settings in the proposed model, we execute a series of ablation experiments on the Pascal Sentence dataset. In those experiments, we denote the number and the dimensions of cross-modal embeddings as K, d_{cross}, respectively, and the dimensions of metric embeddings as d_{metric}.

The Number of EmbeddingsK. Figure 6 shows the effect of the PTF-SimCM model with different K, which is the number of the cross-modal embeddings, on the Pascal Sentence dataset. According to the "Proposed Method" section, K represents k different sentence semantics. We set $d_{\text{cross}} = 1024$ and $d_{\text{metric}} = 128$. To better comprehend the effect of the number K, we vary it from 1 to 4 and compare the experimental results of metric Recall@1 (also known as "Precision@1") under different epochs. When $K = 1$, it means that the cross-modal encoder is a typical one-to-one model. From Figure 6, we can see a significant improvement

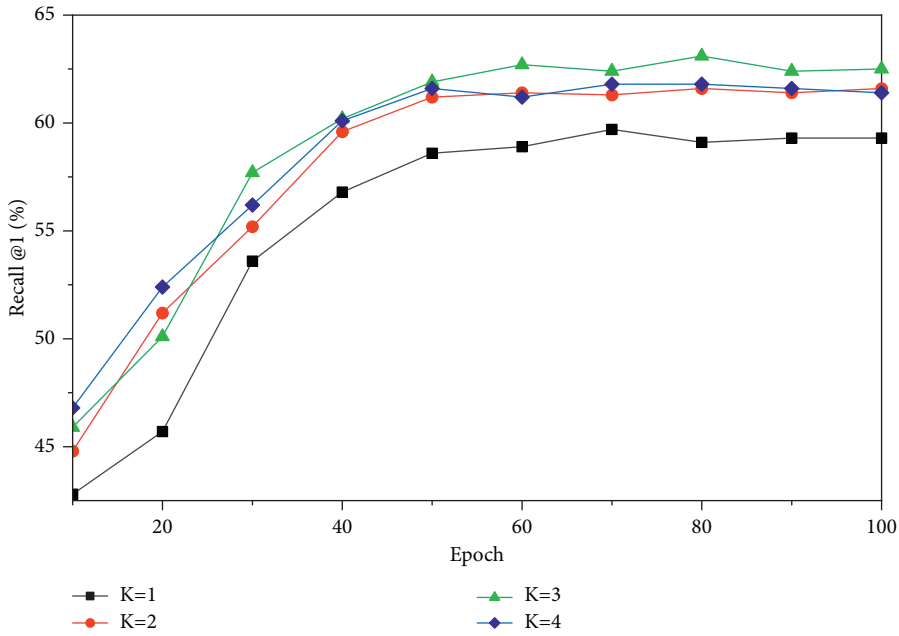


Figure 6: Performance with different numbers of K.

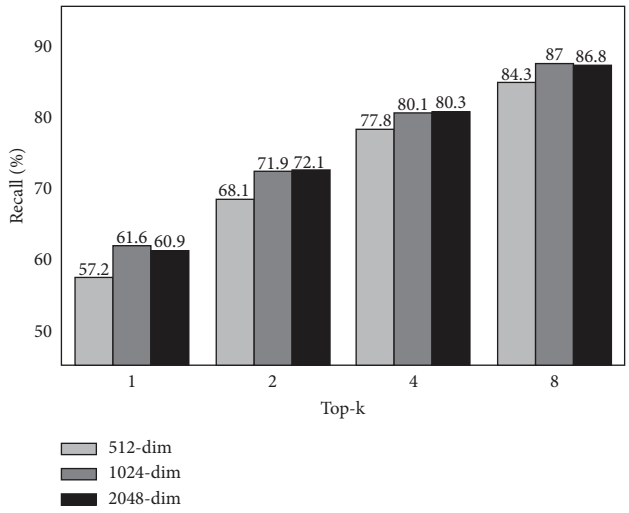


Figure 7: Experimental results on different dimensions of cross-modal embedding.

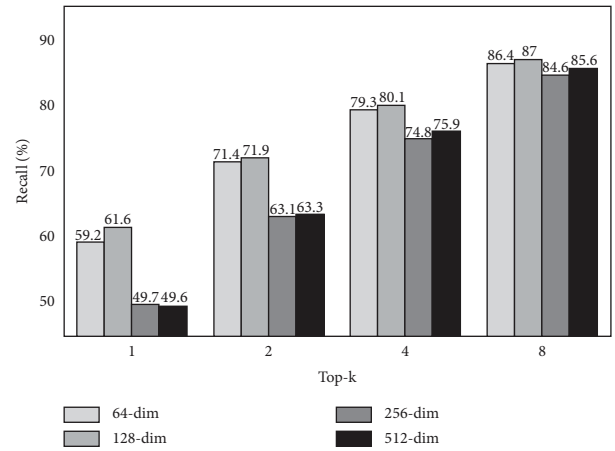


Figure 8: Experimental results on different dimensions of metric embedding.

from $K = 1$ to $K = 2$; this shows the necessity to consider ambiguity. When $K = 3$, the performance of almost all epochs reaches its peak.

The Dimensions of Cross-Modal Embeddings. We employ experiments on $d_{\text{cross}} \in \{512, 1024, 2048\}$ on the Pascal Sentence dataset with $K = 2$ and $d_{\text{metric}} = 128$. Figure 7 shows the result on different dim based on the Recall@k metric. From the result, we can find that as the parameter K increases gradually, the Recall will increase, and when $d_{\text{cross}} = 1024$ and $d_{\text{cross}} = 2048$, the performances are significantly better than those when $d_{\text{cross}} = 512$. Considering all Recall values, $d_{\text{cross}} = 1024$ is the best. This conclusion shows that the dimensions of the cross-modal embedding cannot be too small, which may lose important information and not be too large, which may cause redundancy.

The Dimensions of Metric Embedding. Figure 8 shows the Recall values on the Pascal Sentence dataset when we vary d_{metric} while $K = 2$ and $d_{\text{cross}} = 1024$ and $d_{\text{metric}} \in \{64, 128, 256, 512\}$. The results show that there is noticeable difference between the performances of $d_{\text{metric}} = 64, 128$ and $d_{\text{metric}} = 256, 512$. When $d_{\text{metric}} = 128$, the experiment can get the best Recall, and when $d_{\text{metric}} = 256$, it obtains the

worst. It shows that the dimensions of metric embedding play an important role in our model.

5. Conclusions

In this paper, we present a Simple Contrastive Model with Polysemous Text Fusion (PTF-SimCM) for visual similarity metric. PTF-SimCM is composed of two branches of neural networks and a middle cross-modal encoder. Two branches adopt an asymmetric contrastive structure and shared weights to address the unsupervised visual representation issue. The pretrained cross-modal encoder is used for polysemous expression embedding, and a multimodal fusion operation is designed for feature fusion. To become a simpler and more efficient image similarity measurement, PTF-SimCM straight learns an embedding space where distances directly correspond to a measure of image similarity.

Experimental results on MSCOCO, Flickr 30k, and Pascal Sentence datasets show that PTF-SimCM utilized the information of text modalities more comprehensively, fully considered sentence polysemous, and had better results compared to the baselines. Following the observational evidence, despite the fact that the Recall@k is not the best in general, the MAP@R value reaches the highest value, surpassing the supervised learning model. Moreover, we conducted ablation studies on the numbers and dimensions of cross-modal embeddings, and the results advise us that when $K = 3, d_{\text{cross}} = 1024$, the performances of the proposed model on the overall Recall@k reach the best. For the effect of metric embedding dimensions, even though Recall@k is better when $d_{\text{metric}} = 128$, MAP@R touches the peak at $d_{\text{metric}} = 64$.

Nevertheless, a limitation of our study is that the proposed method only works if there exists a correlation between the two modalities of image and text. When the two modalities are uncorrelated or less correlated, our approach will be ineffective and even negatively affect the similarity metric of images. Accordingly, in future work, we will explore a new strategy of multicomplicated modalities fusion, such as considering both temporal information and spatial information [60], thereby further improving the robustness and performance of the model. Furthermore, we can also explore the use of semisupervised approaches [61, 62] to address the problem of model training without labeled data.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was funded by the Hunan Provincial Natural Science Foundation of China (Grant nos. 2022JJ30231 and 2022JJ50051), the National Natural Science Foundation of China (Grant no. 62072166), and the Innovation Foundation for Postgraduate of Hunan University of Technology (Grant no. CX2213).

References

[1] P. Wohlhart and V. Lepetit, "Learning descriptors for object recognition and 3D pose estimation," in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3109–3118, Boston, MA, USA, June. 2015.

[2] X. He, Y. Zhou, Z. Zhou, S. Bai, and X. Bai, "Triplet-Center Loss for Multi-View 3D Object Retrieval," 2018, https://arxiv.org/abs/1803.06189.

[3] A. Grabner, P. M. Roth, and V. Lepetit, "3D pose estimation and 3D model retrieval for objects in the wild," in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3022–3031, Salt Lake City, UT, USA, June. 2018.

[4] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and Le Song, "Sphereface: deep hypersphere embedding for face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 212–220, Seattle, WA, USA, June, 2017.

[5] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: a unified embedding for face recognition and clustering," in *Proceedings of the 2015 IEEE Conference On Computer Vision And Pattern Recognition (CVPR)*, pp. 815–823, Boston, MA, USA, June. 2015.

[6] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: additive angular margin loss for deep face recognition," in *Proceedings of the 2019 IEEE/CVF Conference On Computer Vision And Pattern Recognition (CVPR)*, pp. 4685–4694, Long Beach, CA, USA, June. 2019.

[7] H. Wan, "CosFace: large margin cosine loss for deep face recognition," in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5265–5274, Salt Lake City, UT, USA, June 2018.

[8] Y. Jin, "Visual search at pinterest," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1889–1898, Association for Computing Machinery, New York, NY, USA, August, 2015.

[9] H. Hu, "Web-scale responsive visual search at bing," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 359–367, New York, NY, USA, July. 2018.

[10] A. Zhai, "Visual discovery at pinterest," in *Proceedings of the 26th International Conference On World Wide Web Companion, Republic and Canton of Geneva*, pp. 515–524, CHE, Switzerland, April. 2017.

[11] Y. Zhang, "Visual search at alibaba," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 993–1001, New York, NY, USA, July 2018.

[12] D. G. Lowe, "Similarity metric learning for a variable-kernel classifier," *Neural Computation*, vol. 7, no. 1, pp. 72–85, 1995.

[13] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K. R. Mullers, "Fisher discriminant analysis with kernels," in *Proceedings of the Workshop Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society (Cat. No.98TH8468)*, pp. 41–48, Madison, WI, USA, August. 1999.

[14] E. P. Xing, A. Y. Ng, M. Jordan, and S. Russell, "Distance metric learning with application to clustering with side-information," in *Proceedings of the 15th Advances in Neural*

Information Processing Systems, pp. 521–528, NIPS, Cambridge, MA, United States, January. 2002.

[15] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proceedings of the International Conference on Machine Learning*, pp. 1597–1607, Chennai, India, November. 2020.

[16] D. Kingma and M. Welling, "Auto-Encoding Variational Bayes," 2014, https://arxiv.org/abs/1312.6114.

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[18] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *Proceedings of the European Conference on Computer Vision*, pp. 69–84, Glasgow, UK, August, 2016.

[19] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," 2018, https://arxiv.org/abs/1803.07728.

[20] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9726–9735, Seattle, WA, USA, June. 2020.

[21] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," 2020, https://arxiv.org/abs/2003.04297.

[22] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and J. Armand, "Unsupervised learning of visual features by contrasting cluster assignments," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9912–9924, 2020.

[23] Y. Song and M. Soleymani, "Polysemous visual-semantic embedding for cross-modal retrieval," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1979–1988, Seattle, WA, USA, June, 2019.

[24] X. Chen and K. He, "Exploring simple siamese representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15750–15758, Nashville, TN, USA, June, 2021.

[25] T.-Y. Lin, "Microsoft COCO: common objects in context," pp. 740–755, 2014, https://arxiv.org/abs/1405.0312.

[26] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, "From image descriptions to visual denotations: new similarity metrics for semantic inference over event descriptions," *Transactions of the Association for Computational Linguistics*, vol. 2, pp. 67–78, 2014.

[27] C. Rashtchian, P. Young, M. Hodosh, and J. Hockenmaier, "Collecting image annotations using Amazon's Mechanical Turk," in *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pp. 139–147, USA, June. 2010.

[28] I. Melekhov, J. Kannala, and E. Rahtu, "Siamese network features for image matching," in *Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 378–383, Cancun, Mexico, December, 2016.

[29] S. Appalaraju and V. Chaoji, "Image Similarity Using Deep CNN and Curriculum Learning," 2018, https://arxiv.org/abs/1709.08761?context=cs.

[30] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, pp. 539–546, San Diego, California, June. 2005.

[31] K. Sohn, "Improved deep metric learning with multi-class N-pair loss objective," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 1857–1865, Red Hook, NY, USA, June, 2016.

[32] X. Wang, X. Han, W. Huang, D. Dong, and M. R. Scott, "Multi-Similarity Loss with General Pair Weighting for Deep Metric Learning," pp. 5017–5025, 2019, https://arxiv.org/abs/1904.06627.

[33] Q. Qian, L. Shang, B. Sun, J. Hu, H. Li, and R. Jin, "SoftTriple Loss: Deep Metric Learning without Triplet Sampling," pp. 6450–6458, 2019, https://arxiv.org/abs/1909.05235.

[34] R. Hjelm, "Devon, alex fedorov, samuel lavoie-marchildon, karan grewal, phil bachman, adam trischler, and yoshua bengio. "Learning deep representations by mutual information estimation and maximization," 2018, https://arxiv.org/abs/1808.06670?fbclid=IwAR3ejzOI7iJ-tzSK91lJZ8EkvF0jntUluaEIxi5leoxmmgM3KhA9TGpWi4g.

[35] J.-B. Grill, "Bootstrap your own latent - a new approach to self-supervised learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 21271–21284, 2020.

[36] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow twins: self-supervised learning via redundancy reduction," in *Proceedings of the 38th International Conference on Machine Learning*, pp. 12310–12320, Virtual Event, July. 2021.

[37] R. Xu, J. Lu, C. Xiong, Z. Yang, and J. J. Corso, "Improving word representations via global visual context," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, NIPS Workshop on Learning Semantics*, p. 9, Republic of Korea, July, 2014.

[38] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," 2013, https://arxiv.org/abs/1301.3781.

[39] S. Kottur, R. Vedantam, J. M. F. Moura, and D. Parikh, "VisualWord2Vec (vis-W2V): learning visually grounded word embeddings using abstract scenes," in *Proceedings of the 2016 IEEE Conference On Computer Vision And Pattern Recognition (CVPR)*, pp. 4985–4994, Las Vegas, Nevada, USA, June. 2016.

[40] L. Zhu, J. Song, X. Zhu, C. Zhang, S. Zhang, and X. Yuan, "Adversarial learning-based semantic correlation representation for cross-modal retrieval," *IEEE MultiMedia*, vol. 27, no. 4, pp. 79–90, 2020.

[41] L. Zhu, C. Zhang, J. Song, L. Liu, S. Zhang, and Y. Li, "Multi-graph based hierarchical semantic fusion for cross-modal representation," in *Proceedings of the 2021 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, Shenzhen, China, July. 2021.

[42] L. Zhu, C. Zhang, J. Song, S. Zhang, C. Tian, and X. Zhu, "Deep multi-graph hierarchical enhanced semantic representation for cross-modal retrieval," *IEEE Multimed*, 2022.

[43] J. L. Ba, K. Swersky, S. Fidler, and R. Salakhutdinov, "Predicting Deep Zero-Shot Convolutional Neural Networks Using Textual Descriptions," pp. 4247–4255, 2015, https://arxiv.org/abs/1506.00511?context=cs.

[44] X. Huang and Y. Peng, "Deep cross-media knowledge transfer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8837–8846, Seattle, WA, USA, August, 2018.

[45] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, San Francisco, CA, USA, June, 2016.

[46] J. Pennington, R. Socher, and C. Manning, "GloVe: global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, Doha, Qatar, July, 2014.

[47] K. Cho, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, Doha, Qatar, July, 2014.

[48] Z. Lin, M. Feng, C. Nogueira dos Santos et al., "A structured self-attentive sentence embedding," 2017, https://arxiv.org/abs/1703.03130.

[49] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[50] J. Lei, J. Ryan Kiros, and E. Geoffrey, "Hinton. "Layer normalization," 2016, https://arxiv.org/abs/1607.06450.

[51] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, pp. 448–456, Lille, France, August, 2015.

[52] S. Lloyd, "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, Mar. 1982.

[53] J. Johnson, A. Karpathy, and L. Fei-Fei, "DenseCap: fully convolutional localization networks for dense captioning," in *Proceedings of the 2016 IEEE Conference On Computer Vision And Pattern Recognition (CVPR)*, pp. 4565–4574, Las Vegas, Nevada, USA, June. 2016.

[54] K. Musgrave, S. Belongie, and S.-N. Lim, "A Metric Learning Reality Check," 2020, https://arxiv.org/abs/2003.08505.

[55] I. Loshchilov and H. Frank, "Sgdr: stochastic gradient descent with warm restarts," 2016, https://arxiv.org/abs/1608.03983.

[56] P. Micikevicius, S. Narang, J. Alben et al., "Mixed precision training," 2017, https://arxiv.org/abs/1710.03740.

[57] B. Ko, G. Gu, and H.-G. Kim, "Learning with memory-based virtual classes for deep metric learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11792–11801, Nashville, TN, USA, June, 2021.

[58] S. Kim, D. Kim, M. Cho, and S. Kwak, "Embedding transfer with label relaxation for improved metric learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3967–3976, Nashville, TN, USA, June 2021.

[59] O. Russakovsky, J. Deng, H. Su et al., "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[60] D. Zhang, L. Yao, K. Chen, S. Wang, X. Chang, and Y. Liu, "Making sense of spatio-temporal preserving representations for EEG-based human intention recognition," *IEEE Transactions on Cybernetics*, vol. 50, no. 7, pp. 3033–3044, 2020.

[61] K. Chen, L. Yao, D. Zhang, X. Wang, X. Chang, and F. Nie, "A semisupervised recurrent convolutional attention model for human activity recognition," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 5, pp. 1747–1756, 2020.

[62] M. Luo, X. Chang, L. Nie, Y. Yang, A. G. Hauptmann, and Q. Zheng, "An adaptive semisupervised feature analysis for video semantic recognition," *IEEE Transactions on Cybernetics*, vol. 48, no. 2, pp. 648–660, 2018.