IDC: Boost Text-to-Image Retrieval via Indirect and Direct Connections

Guowei Ge, Kuangrong Hao[*], Lingguang Hao
College of Information Science and Technology, Ministry of Education, Donghua University
Shanghai 201620, P. R. China
ggw@mail.dhu.edu.cn, krhao@dhu.edu.cn, haolingguang@mail.dhu.edu.cn

Abstract

The Dual Encoders (DE) framework maps image and text inputs into a coordinated representation space, and calculates their similarity directly. On the other hand, the Cross Attention (CA) framework performs modalities interactions after completing the feature embedding of images and text, and then outputs a similarity score. For scenarios with bulk query requests or large query sets, the latter is more accurate, but the former is faster. Therefore, this work finds a new way to improve the retrieval accuracy of the DE framework by borrowing the advantages of the CA framework. Drawing inspiration from image captioning, we introduce a text decoder in the model training stage to simulate the cross-modal interaction function, like the CA framework. The text decoder is eventually discarded, aligning our model with the DE framework. Finally, to ensure training stability and prevent overfitting, we modify the Self-Distillation from Last Mini-Batch and apply it to the retrieval areas. Extensive experiments conducted on the MSCOCO and Flickr30K datasets validate the effectiveness of our proposed methods. Notably, our model achieves competitive results compared to state-of-the-art approaches on the Flickr30K dataset.

Keywords: Text-to-Image retrieval, Dual Encoders, Cross Attention, Indirect Connection, Regularization

1. Introduction

With the decreasing cost of acquiring multimodal information, there is a growing enthusiasm for exploring the relationships between data from different modalities. This trend has sparked research in cross-modal retrieval (Bogolin et al., 2022; Bain et al., 2021), with text-to-image retrieval being a particularly popular area of focus. Text-to-image retrieval involves the development of intelligent methods to establish a similarity function between cross-modal data.

Dual Encoders (DE) and *Cross Attention* (CA) are two prominent frameworks for processing image and text (Miech et al., 2021). Figure 1(a) illustrates the DE framework (Wang et al., 2022; Zhang et al., 2020), where visual and textual inputs are embedded into a coordinated representation space by independent encoders. The similarity between features is then determined using cosine distance or Euclidean distance. Such approaches require less computation and are particularly time-effective for retrieval tasks involving large query sets. However, their accuracy is limited due to feature fusion being performed only during the final similarity calculation. Figure 1(b) showcases the CA framework (Li et al., 2020b; Zhang et al., 2021) , which incorporates early fusion of textual and visual features. In addition to the respective feature embedding of the two modalities, it establishes interaction between them using cross-attention or self-attention, to obtain similarity scores through regression. The application of cross-modal fusion is doubtless a significant improvement in retrieval performance, but it also introduces an explosion in computational complexity, since this approach needs to exhaustively consider all image-text combinations in the query set. Therefore, how to improve speed whilst ensuring retrieval accuracy is an urgent issue to be solved in this field.

Inspired by image captioning, we intend to introduce a text decoder in the training stage to simulate the modality interaction function in the CA framework. This decoder takes features of the image and text as input and then generates the corresponding caption description of the image. We aim to simultaneously bring the embedded features in the text decoder closer to those in the image and text Encoders, which is beneficial for reducing the distance between image-text pairs. As shown in Figure 1(c), such an approach is called Indirect Connection, while the similarity calculation between the original text and the image is called Direct Connection. In the inference stage, the text decoder is eventually discarded, ensuring that our model is consistent with the DE framework.

Maintaining training stability and preventing overfitting has always been an important part of model training. Self-Distillation from Last Mini-Batch (DLB regularization), proposed in the paper (Shen et al., 2022), has demonstrated its effectiveness in image classification tasks. We improve the DLB and adapt it for application in the domain of multimodal retrieval, which helps to increase the stability and consistency of the training and improve the generalisation ability of our model. To the best of our

[*] Corresponding Author.

LREC-COLING 2024, pages 8546–8555
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

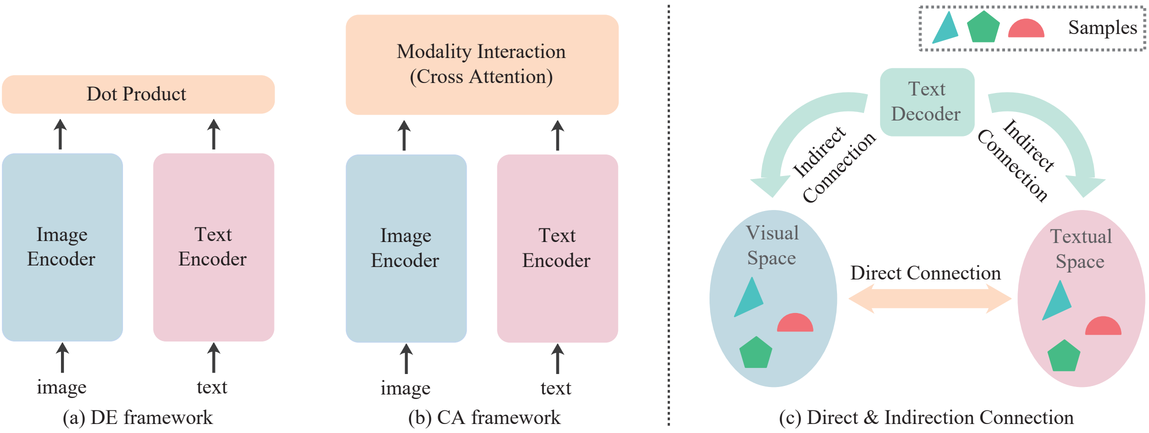

Figure 1: (a) Illustration of the DE framework; (b) Illustration of the CA framework; (c) Visualization of Direct and Indirect Connection. The visual (resp. textual) space refers to the feature space where the image (resp. text) samples are located after being encoded by the image (resp. text) encoder.

knowledge, this is the first instance of DLB being applied to the field of multimodal retrieval.

In summary, the contributions can be summarized as follows:

- Inspired by image captioning, we employ a text decoder to imitate the modal fusion functionality of CA to avoid training the CA retrieval model, significantly saving training and inference time.
- Using the text decoder as a node, we established the Indirect Connection to minimize the distance between the embedded features in the text decoder and those in the image or text encoder, making it easier to match the corresponding image and text.
- To ensure the stability and consistency of the training process, DLB regularization is modified and applied specifically to the field of multimodal retrieval, which helps prevent overfitting and enhances the reliability of the training process.
- Extensive experiments on the MSCOCO Lin et al. (2014) and Flickr30K Plummer et al. (2015) benchmark datasets confirm the effectiveness of our methods. Remarkably, our model yields new state-of-the-art results on the widely-used Flickr30K dataset.

2. Related Work

2.1. Vision and Language Representations.

Because of the natural semantic gap between different modalities, pre-training plays a crucial role in feature extraction for various multimodal tasks, such as image Captioning (IC) (Cornia et al., 2020; Pan et al., 2020), Visual Question Answering (VQA) (Kim et al., 2020), Cross-modal Retrieval (CR) (Lu et al., 2022; Gorti et al., 2022), etc. Prior to the advent of multimodal learning (Baltrušaitis et al., 2019), visual models were typically pre-trained through an image classification or object detection tasks, whilst text models underwent self-supervised training.

Multimodal learning refers to the process of training visual and textual models together within a unified task or framework, which helps to further narrow the semantic gap between different modalities. The methods we call CA (Li et al., 2020b; Zhang et al., 2021) fuse visual and textual features at an early stage and perform joint reasoning, which clearly facilitates models to learn multimodal joint representations. Recently, the methods we call DE (Lu et al., 2022; Radford et al., 2021) process visual and textual features individually and then learn a coordinated representation space for matching relevant image-text pairs by contrastive loss.

2.2. Text-to-image Retrieval

Text-to-image retrieval enables users to locate specific images based on their descriptive text. Suppose the query set has a size of Q, roughly Q model calculations are necessary for the CA model when a query request ensues. The DE model only requires just about one model calculation. This is because the DE model can pre-extract the features of all samples of the query set and subsequently obtain the similarity matrix by matrix multiplication. (Miech et al., 2021) utilizes knowledge distillation to transfer the knowledge learned by the CA model to the DE model in the training phase, which is used to enhance the capability of the DE model. In the inference phase, the DE model first obtains the top K relevant samples. Then the CA model sorts these K samples to obtain the final results. Different from that paper, we do not train a real CA model but employ a text decoder to simulate its ability to learn fused features. And the text decoder can be discarded in the inference phase, so it does not add extra parameters.

3. Methods

Our proposed approaches are presented in detail in this section. As shown in Figure 2, our model comprises three modules: image encoder, text encoder, and text decoder. The image encoder incorporates a pre-trained visual encoder (e.g., CLIP (Radford et al., 2021)) along with self-attention layers. The text encoder consists of a pre-trained language model (e.g., a CLIP textual encoder) combined with masked self-attention layers. And the text decoder solely consists of cross-attention layers. Due to the inherent semantic gap between different modalities, training a multimodal model from scratch often requires a substantial amount of data support, which places significant demands on the computational resources and training time. Consequently, fine-tuning on pre-trained models emerges as a simpler and more efficient alternative.

Given an image x and its corresponding text description y, our model maps them to distinct feature subspaces and computes the similarity score between the image feature f_x and the text feature f_y. In other words, the model creates a connection between the image and text modality, discriminating the paired image-text from negative ones. In this work, we introduce two types of connections: the Direct Connection and the Indirect Connection.

3.1. Direct Connection

Given images x, the image encoder ϕ_x will map x to the image features f_x, as follow:

$$f_x = \phi_x\left(x\right), \tag{1}$$

where $x \in \mathbb{R}^{C\times W\times H}$, C is the number of channels of the RGB image, W and H are the width and height respectively, $f_x \in \mathbb{R}^{N_x\times d_{model}}$, N_x is related to the downsampling rate of the image encoder, and d_{model} is the feature dimension.

Since we employ a text decoder and an additional image captioning task to implement the function of feature fusion in CA. The masked self-attention layer (Vaswani et al., 2017) is necessary to prevent information leakage during training. However, the forward mask results in a partial loss of contextual information. To solve this problem, we have inverted the mask instead of the text input $y = \left[y^1, y^2, ..., y^T\right]$ to compensate for the text features $f_y \in \mathbb{R}^{N_y\times d_{model}}$, where $T\ \left(= N_y\right)$ is the length of the input sentence. Formally, the operation can be written as:

$$f_y = f_y^{fwd} + f_y^{bwd} = \phi_y^{fwd}\left(y\right) + \phi_y^{bwd}\left(y\right), \tag{2}$$

where $\phi_y^{fwd}\left(y\right)$ and $\phi_y^{bwd}\left(y\right)$ both indicate the text encoder, the only difference between them is that the former uses a forward mask and the latter uses a backward mask.

A vast majority of cross-modal retrieval models (Chen et al., 2020; Wang et al., 2022; Zhang et al., 2020; Li et al., 2020a) only utilize the Direct Connection, by directly reducing the contrastive loss between positive image-text pairs relative to negative ones:

$$L_{IT}^{I2T} = -\frac{1}{N_b}\sum_{i=1}^{N}\log\left(1/\left(1+\frac{\sum_{y\in\mathcal{D}_{x_i}^-} e\left(f_{x_i}, f_y\right)}{\sum_{y\in\mathcal{D}_{x_i}^+} e\left(f_{x_i}, f_y\right)}\right)\right), \tag{3}$$

$$s\left(f_x, f_y\right) = \frac{\bar{f}_x \cdot \bar{f}_y}{\left\|\bar{f}_x\right\| \cdot \left\|\bar{f}_y\right\|}, \tag{4}$$

$$e\left(f_x, f_y\right) = \exp\left(s\left(f_x, f_y\right)/\tau\right) \tag{5}$$

where $\bar{f}_x$ (resp. $\bar{f}_y$) is the mean value of f_x (resp. f_y), N_b is the batch size, τ is the temperature factor, $\mathcal{D}_{x_i}^+$ (resp. $\mathcal{D}_{x_i}^-$) represents the set of positive (resp. negative) samples for x_i, and $s\left(\cdot,\cdot\right)$ is applied to calculate the cosine similarity between samples. To facilitate the calculation, we consider the matched samples within a batch as positive samples and the unmatched samples as negative samples:

$$\mathcal{D}_{x_i}^+ = \left\{\left(x_i, y_i\right)\right\}, \mathcal{D}_{x_i}^- = \left\{\left(x_i, y_j\right)|j \neq i\right\}_{j=1,2,...,N_b}. \tag{6}$$

Eq. (3) only calculates the image-to-text contrastive loss. To make the network also have the ability of text-to-image retrieval, we define the text-to-image contrastive loss as:

$$L_{IT}^{T2I} = -\frac{1}{N_b}\sum_{i=1}^{N}\log\left(1/\left(1+\frac{\sum_{x\in\mathcal{D}_{y_i}^-} e\left(f_{y_i}, f_x\right)}{\sum_{x\in\mathcal{D}_{y_i}^+} e\left(f_{y_i}, f_x\right)}\right)\right), \tag{7}$$

where $\mathcal{D}_{y_i}^+$ (resp. $\mathcal{D}_{y_i}^-$) represents the set of positive (resp. negative) samples for y_i. The final loss of the Direct Connection is noted as:

$$L_{IT} = L_{IT}^{I2T} + L_{IT}^{T2I}. \tag{8}$$

Although our objective is only text-to-image retrieval, bi-directional contrastive loss enables better exploitation of the correlation between images and text, thus further improving retrieval performance.

3.2. Indirect Connection

In the Direct Connection, feature fusion is limited to the computation of similarity $s\left(\cdot,\cdot\right)$. Such a simplistic calculation does not fully exploit model's ability. Although the CA model can effectively utilize

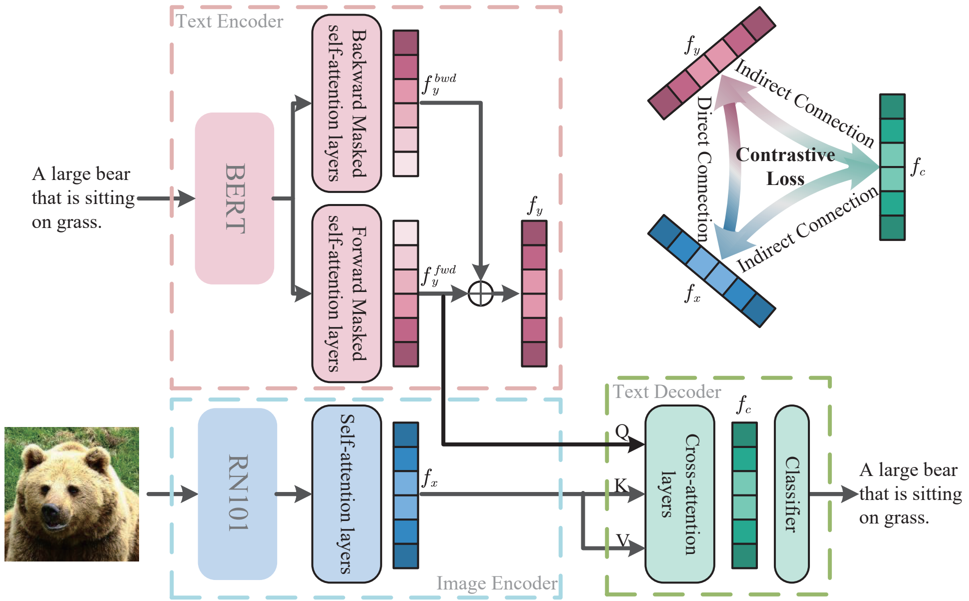

Figure 2: Illustration of the work flow in train phase. The image encoder and text encoder process the image and text inputs respectively, and then their outputs are fed into the text decoder to generate captions. At the same time, utilizing contrastive loss to narrow the distance between image features f_x, text features f_y and caption features f_c. The contrastive loss between f_x and f_y is called Direct Connection. The contrastive loss between f_c and f_x/f_y is called Indirect Connection.

the fused features, it entails lengthy training and inference time, which is not suitable for real-time retrieval with large query sets. Consequently, we introduce the Indirect Connection to bridge the gap between the embedded features in the text decoder and those in the image or text encoder, making it easier to match the corresponding image and text.

The embedded features in text decoder, denoted as $f_c \in \mathbb{R}^{N_c \times d_{model}}$, are calculated as follows:

$$f_c = \phi_c\left(\phi_x(x), \phi_y^{fwd}(y)\right), \quad (9)$$

where ϕ_c is the text decoder.

In purpose to enable the text decoder to output words, it is necessary to map f_c to the vocabulary space. We use cross entropy to train the image captioning task with the following loss function:

$$L_{Cap} = -\frac{1}{N_b}\sum_{i=1}^{N_b}\sum_{t=1}^{T}\log\left(P_\theta\left(y_i^t | y_i^{t-1}, ..., y_i^1, x_i\right)\right), \quad (10)$$

where $P_\theta\left(y_i^t | y_i^{t-1}, ..., y_i^1, x_i\right)$ denotes the output probability of the text decoder for the token y_i^t at the position t, taking into account the previous tokens $y_i^{t-1}, ..., y_i^1$ and the image x_i. Therefore in Eq. (9) we only use ϕ_y^{fwd} to encode y_i to prevent the information of y_i^t and subsequent tokens from being leaked.

The Indirect Connection is achieved by calculating the similarity score between f_x and f_y through the intermediary features f_c. The formula can be expressed as follows:

$$s'\left(f_x, f_y\right) = s\left(f_x, f_c\right) s\left(f_c, f_y\right). \quad (11)$$

Therefore four contrastive losses are needed to constitute the Indirect Connection loss:

$$L_{IC}^{I2C} = -\frac{1}{N_b}\sum_{i=1}^{N}\log\left(1/\left(1+\frac{\sum_{c\in\mathcal{D}_{x_i}^-} e\left(f_{x_i}, f_c\right)}{\sum_{c\in\mathcal{D}_{x_i}^+} e\left(f_{x_i}, f_c\right)}\right)\right), \quad (12)$$

$$L_{IC}^{C2I} = -\frac{1}{N_b}\sum_{i=1}^{N}\log\left(1/\left(1+\frac{\sum_{x\in\mathcal{D}_{c_i}^-} e\left(f_{c_i}, f_x\right)}{\sum_{x\in\mathcal{D}_{c_i}^+} e\left(f_{c_i}, f_x\right)}\right)\right), \quad (13)$$

$$L_{CT}^{C2T} = -\frac{1}{N_b}\sum_{i=1}^{N}\log\left(1/\left(1+\frac{\sum_{y\in\mathcal{D}_{c_i}^-} e\left(f_{c_i}, f_y\right)}{\sum_{y\in\mathcal{D}_{c_i}^+} e\left(f_{c_i}, f_y\right)}\right)\right), \quad (14)$$

$$L_{CT}^{T2C} = -\frac{1}{N_b}\sum_{i=1}^{N}\log\left(1/\left(1+\frac{\sum_{c\in\mathcal{D}_{y_i}^-} e\left(f_{y_i}, f_c\right)}{\sum_{c\in\mathcal{D}_{y_i}^+} e\left(f_{y_i}, f_c\right)}\right)\right), \quad (15)$$

$$L_{IC} = L_{IC}^{I2C} + L_{IC}^{C2I}, L_{CT} = L_{CT}^{C2T} + L_{CT}^{T2C}, \quad (16)$$

where L_{IC} is used to calculate the loss between f_x and f_c, while L_{IC} is used to calculate the loss between f_c and f_y. The operational principle of these two loss functions aligns with L_{IT} in Eq. (8).

3.3. DLB Regularization

Without the advantage of pre-training, the model's capacity to generalize from limited data can be compromised. Therefore, the inclusion of an effective regularization is vital to enhance the model's performance and ensure robust training. We improve DLB (Shen et al., 2022), which reorders sampling by restricting half of each training batch to overlap with the previous iteration. In other words, the soft target generated in the previous iteration performs knowledge distillation on the current first half batch.

Concretely, let t denote the iteration step. $\mathcal{S}^t$ represents the sampled data. And we combine $\mathcal{S}^{t-1}$ and $\mathcal{S}^t$ to form the data of the training batch: $\mathcal{B}^t = \left[\mathcal{S}^{t-1}, \mathcal{S}^t\right]$. The soft target P^{t-1} is yielded by the following equation:

$$P^{t-1} = F_y^{t-1}\left[\frac{N_b}{2}:, \frac{N_b}{2}:\right] \times \left(F_x^{t-1}\left[\frac{N_b}{2}:, \frac{N_b}{2}:\right]\right)^T, \tag{17}$$

$$F_x^{t-1} = \mathrm{norm}\left(\mathrm{mean}\left(\phi_x\left(\mathcal{B}_x^{t-1}\right)\right)\right), \tag{18}$$

$$F_y^{t-1} = \mathrm{norm}\left(\mathrm{mean}\left(\phi_{\mathrm{y}}\left(\mathcal{B}_y^{t-1}\right)\right)\right), \tag{19}$$

where $\mathrm{mean}(\cdot)$ and $\mathrm{norm}(\cdot)$ denote the mean and l_2 normalization operations for the independent samples in $\mathcal{B}^{t-1}$, respectively. Similarly, P^t can be obtained as

$$P^t = F_y^t\left[:\frac{N_b}{2}, :\frac{N_b}{2}\right] \times \left(F_x^t\left[:\frac{N_b}{2}, :\frac{N_b}{2}\right]\right)^T. \tag{20}$$

Consequently, we introduce the regularization loss as follows:

$$L_{DLB} = \frac{2}{N_b} D_{KL}\left(P^{\tau,t-1} || P^{\tau,t}\right), \tag{21}$$

where $P^{\tau,t-1} = \mathrm{softmax}\left(P^{t-1}/\tau\right)$, $P^{\tau,t} = \mathrm{softmax}\left(P^t/\tau\right)$ and $D_{KL}\left(\cdot||\cdot\right)$ is the KL divergence.

Since $\mathcal{S}^t$ is computed within both adjacent batches, it is equivalent to expanding the number of negative samples in disguise. Compared to storing a momentum model and a memory bank in previous work (He et al., 2020), the memory consumed by the DLB regularization is negligible. Conclusively, the general loss is expressed as:

$$L = L_{IT} + L_{Cap} + L_{IC} + L_{CT} + \alpha L_{DLB}, \tag{22}$$

where α is the hyperparameter to control the strength of the regularization term.

4. Experiments

4.1. Experimental Settings

Dataset. We use two datasets to train and evaluate our approach: (1) MSCOCO is composed of 123,287 images, each with 5 captions. Following the Karpathy split Karpathy and Fei-Fei (2015), we use 5,000 images for testing, 5,000 images for validation, and the rest for training. (2) Flickr30K consists of approximately 30K images with 5 captions per image. We follow the paper Karpathy and Fei-Fei (2015), where 1K images are used for testing, 1K for validation and the remaining for training. R@1 (resp. R@5 and R@10) represents the recall of the top 1 (resp. 5 and 10) text-to-image results on the test set.

Models. We adopted the visual coder (RN101) of CLIP (Radford et al., 2021) and FasterRCNN (Ren et al., 2015) as the backbones of the image encoder. For the text encoder, we use BERT (Devlin et al., 2018), GPT2 (Radford et al., 2019) and the text encoder of CLIP as backbones. In order not to diminish the capability of the backbone of image encoder, we fixed its parameters. Also to adapt the text encoder to the language style of the new dataset, we fine-tune it. If not specified, RN101 and BERT are used by default in the following experiments.

Implementation Details. We set the number of attention layers to 3 for the image encoder, text encoder and text decoder. d_{model} is set to 768. The temperature factor τ in Eq. (3) is set to 0.07. Hyperparameter α in Eq. (22) is set to 20. The Multi-head attention is applied to our model, whose number of heads is set to 12.

To alleviate overfitting during training, we set the initial learning rate to 4e-5, and adopt the cosine annealing (Loshchilov and Hutter, 2016) learning rate scheduler. Adam optimizer is applied to the training process. The deep learning library Pytorch 1.8.1 and relevant third-party libraries are used to develop our model. The experiments are all implemented in Python on a personal computer with 16GB memory and one NVIDIA QUADRO RTX 8000 GPU.

4.2. Quantitative Results and Analysis

Comparing state of the art result on Flickr30K 1K test set. We compare our model with state of the art on Flickr30K 1K test set. As shown in Table 1, our model outperforms text-to-image retrieval metrics even when compared to models pre-trained on larger multimodal datasets such as COT (Lu et al., 2022) and ViLEM (Chen et al., 2023). Specifically, compared with the most rescent DE model ViLEM (Chen et al., 2023), our model gets higher results on R@1 (from 78.1 to 82.5), R@5 (from 94.6 to 97.1) and R@10 (from 97.0 to 98.6). It is worth noting that our training set contains only Flickr30K training data. Besides, RN101 (from CLIP (Radford et al., 2021)) for the image encoder and BERT (Devlin et al., 2018) for the text encoder are publicly available resources. Therefore, the training cost of our method is very low and the training time is very short.

Methods	Type	R@1	R@5	R@10
VILBERT (3.1M) (Lu et al., 2019)	CA	58.2	84.9	91.5
PixelBERT (5.6M) (Huang et al., 2020)	CA	59.8	85.5	91.6
Unicoder-VL (3.8M) (Li et al., 2020a)	CA	71.5	90.9	94.9
UNITER (9.6M) (Chen et al., 2020)	CA	75.6	94.1	96.8
OSCAR (6.50M) (Li et al., 2020b)	CA	75.9	93.3	96.6
Fast and Slow (5.5M) (Miech et al., 2021)	DE+CA	72.1	91.5	95.2
Frozen in time (2.7M) (Bain et al., 2021)	DE	61.0	87.5	92.7
LightningDOT (9.5M) (Sun et al., 2021)	DE	69.9	91.1	95.2
COOKIE (5.9M) (Wen et al., 2021)	DE	68.3	91.1	95.2
COTS (15.3M) (Lu et al., 2022)	DE	76.5	93.9	96.6
ViLEM (14.1M) (Chen et al., 2023)	DE	78.1	94.6	97.0
ours	DE	**82.5**	**97.1**	**98.6**

Table 1: Comparison to state of the art on Flickr30K 1K test set for text-to-image retrieval.

Methods	FT	R@1	R@5	R@10
CLIP (Radford et al., 2021)	w/o	37.8	62.4	72.2
CLIP (RN101) (Radford et al., 2021)	w/o	30.7	55.5	66.0
ALIGN (Jia et al., 2021)	w/o	45.6	69.8	78.6
COTS (Lu et al., 2022)	w/o	43.8	71.6	81.3
ViLEM (Chen et al., 2023)	w/o	47.7	75.2	84.5
LightningDOT (Sun et al., 2021)	w/	45.8	74.6	83.8
COOKIE (Wen et al., 2021)	w/	46.6	75.2	84.1
COTS (Lu et al., 2022)	w/	50.5	77.6	86.1
ViLEM (Chen et al., 2023)	w/	54.5	80.6	88.2
ours	w/	43.6	73.0	83.0

Table 2: Comparisons of the text-to-image retrieval results (without fine-tuning) on the MSCOCO full 5K test set. FT: fine-tuning.

CLIP (RN101)	L_{IT}	L_{Cap}	L_{IC}	L_{CT}	R@1	R@5	R@10
w/o FT					30.70	55.50	66.03
CLIPLike	✓				36.50	66.28	77.63
	✓	✓	✓	✓	37.38	67.22	78.14
Mean pooling	✓				39.28	69.48	78.00
	✓	✓	✓	✓	**40.76**	**70.44**	**80.59**

Table 3: Comparing the results of CLIP model under different finetune approaches on MSCOCO 5k test set. FT: fine-tuning; CLIPLike: Using cls tokens as proxy for images and text; Mean pooling: Extracting features of images and text by mean pooling

Comparing state of the art results on the MSCOCO 5K test set. We report the comparative results on MSCOCO 5K test set in Table 2. We can observe that: (1) The results in the first row of the table are from the paper (Lu et al., 2022), and the results in the second row are obtained from our own tests using RN101 as the backbone. It is obvious that the results of our test are lower, because of the different backbones chosen. (2) Our results are significantly worse than those of finetune on MSCOCO for such methods as ViLEM(Chen et al., 2023). This is because, first of all, these models have been pre-trained on large datasets. Although our approach makes use of CLIP's visual coder, its training set is expanded from the network, so its text data is noisy. For visual language learning, noisy image-text pairs are not optimal. As can be seen from the data in Table 2 and Table 3, when the text encoder of CLIP is replaced, the results are instead better. The second reason is that, in order not to increase the number of parameters and the inference time of the network too much, we do not employ the best performing visual coder (much larger and slower than we use) in CLIP. (3) The impact of DLB regularization on the metrics of models trained on small datasets is much larger than models trained on large datasets. Subsequent ablation experiments (Table 4 and 5) can also verify this conclusion. Large datasets such as MSCOCO are less dependent on regularization due to their rich data diversity. While small datasets such as

Backbone	L_{IT}	L_{Cap}	L_{IC}	L_{CT}	L_{DLB}	R@1	R@5	R@10
RN101 + BERT	✓					39.28	69.48	78.00
	✓	✓				38.43	68.89	79.60
	✓	✓	✓			39.36	69.02	80.19
	✓	✓	✓	✓		40.76	70.44	80.59
	✓	✓	✓	✓	✓	**43.60**	**73.01**	**83.01**
RN101 + GPT2	✓					37.37	68.06	79.12
	✓	✓				37.79	68.36	37.37
	✓	✓	✓			38.25	68.63	79.68
	✓	✓	✓	✓		39.53	69.56	80.00
	✓	✓	✓	✓	✓	**42.04**	**71.83**	**82.00**
FasterRCNN + BERT	✓					35.03	66.81	78.46
	✓	✓				34.60	66.09	78.13
	✓	✓	✓			35.16	66.74	78.28
	✓	✓	✓	✓		36.37	67.53	79.16
	✓	✓	✓	✓	✓	**39.70**	**70.54**	**82.00**

Table 4: Ablation study on different backbones on MSCOCO 5k test set. RN101: One of the CLIP image encoder (ResNet 101).

Backbone	L_{IT}	L_{Cap}	L_{IC}	L_{CT}	L_{DLB}	R@1	R@5	R@10
RN101+BERT $(\tau = 0.07)$	✓					57.80	85.46	92.76
	✓	✓				56.98	84.82	92.08
	✓	✓	✓			56.64	85.34	92.40
	✓	✓	✓	✓		58.82	86.14	93.22
	✓				✓	81.50	96.80	98.50
	✓	✓	✓	✓	✓	**82.50**	**97.10**	**98.60**

Table 5: Ablation study on Flickr30K 1K test set

Flickr30K may be prone to overfitting and training instability due to poor data diversity, and thus have a large reliance on regularization.

4.3. Ablation Study

Ablation results of different fine-tuning methods. We compare the impact of different fine-tuning strategies on the performance with our model in Table 3. We found that employing cls tokens as representatives of image and text features on the MSCOCO dataset is not as simple and efficient as directly using mean pooling (R@1: from 36.5 to 39.28). Because the approach of using cls token requires more training data, while mean pooling is more suitable for some small datasets for fine-tuning. It can also be found that, regardless of the fine-tuning method, the Indirect Connection is able to improve the text-to-image retrieval capacity of the model (R@1 in CLIPLike: from 36.50 to 37.38, R@1 in Mean pooling: from 39.28 to 40.76).

Ablation study of different backbones of the image and text encoder. As shown in Table 4, we perform ablation experiments with three different combinations of backbones ("RN101+BERT", "RN101+GPT2" and "FasterRCNN+BERT"). It can be observed that: (1) our Indirect Connection and modified DLB regularization are generalizable for different backbones. In particular, the combination "RN101+BERT" improves R@1 from 39.28 to 43.60, R@5 from 69.48 to 73.01 and R@10 from 78.00 to 83.01. (2) BERT is not suitable for the image captioning task. Focus on the "RN101+BERT" and "FasterRCNN+BERT" combinations. The addition of L_{cap} instead makes the evaluations lower (R@1: from 39.28 (resp. 35.03) to 38.43 (resp. 34.6)). Because the output features of BERT are naturally bi-directional, which is not friendly for training image captioning task. Nevertheless, the Indirect Connection makes up for the for the drop in performance.

Ablation results on Flickr30K 1K test set. In Table 5, we present the ablation results on the Flickr30K 1K test set. The Indirect Connection raises R@1 by 1.02 (from 57.80 to 58.82) without DLB regularization and by 1 (from 81.50 to 82.50) with DLB regularization. This proves that there is no mutual exclusivity between the Indirect Connection and the DLB regularization. Surprisingly, with DLB regularization alone, the model can be improved by 22.80 on the R@1 metric, 11.46 on the R@5 metric and 5.84 on the R@10 metric. However it does not show such a significant advantage on the MSCOCO dataset. This demonstrates the clear superiority of DLB for training and fine-tuning on small datasets.

4.4. Hyperparameter Adjustments

Backbone	α	R@1	R@5	R@10
RN101 +BERT	0	40.76	70.44	80.59
	1	40.54	70.65	80.68
	10	43.04	72.74	82.38
	20	**43.60**	**73.01**	**83.01**
	30	43.29	72.64	82.94

Table 6: Comparing the results of different values of α in Eq. (22) on MSCOCO 5K test set.

Experiments with the hyperparameters α in Eq. (22). As shown in Table 6, we present the experimental results regarding the hyperparameter α taking different values. We can observe that the best results are obtained when alpha is set to 20 (R@1 reached 43.60, R@5 reached 73.01 and R@10 reached 83.01). When the α is too small, the DLB regularization is not strong and does not fully stabilize the training. When α is too large, excessive regularization hinders the model from learning knowledge.

Backbone	τ	L_{DLB}	R@1	R@5	R@10
RN101 +BERT	1	w/o	6.59	19.79	30.00
	0.7	w/o	9.22	25.89	37.59
	0.1	w/o	39.04	69.07	79.76
	0.07	w/o	40.76	70.44	80.59
	0.01	w/o	**41.84**	**71.34**	**81.49**
	0.07	w	**43.6**	73.01	83.01
	0.01	w	43.34	**73.14**	**83.17**

Table 7: Comparing the results of different values of τ in Eq. (3) on MSCOCO 5K test set.

Experiments with the hyperparameters τ in Eq. (3). In Table 7, we display the experimental results with different values of the hyperparameter τ and with/without DLB regularization. In the absence of DLB, the best results are obtained with a value of 0.01 for τ (R@1 reached 41.84, R@5 reached 71.341 and R@10 reached 81.49). And the model has particularly poor learning ability when τ is greater than 0.1. In the presence of DLB, τ does not demonstrate a trend toward better results for smaller values, but rather the highest sum of results at 0.07. This is because the DLB regularization limits the learning speed of the model. After reaching a certain value, even if τ continues to decrease, it will not continue to improve the learning ability of the model and may even make the model lose performance. When the τ is too small, the losses are too large and the computer has an upper limit on the number of values it can store, making it impossible for the model to be trainable.

5. Conclusions

In this article, we aim to improve the precision of text-to-image retrieval while maintaining its speed. Specifically, we employed a text decoder to simulate the interaction function between modalities like the CA framework. Taking the text decoder as a node, we established the Indirect Connection to minimize the distance between the caption features and image/text features, which helps to match the corresponding images and text. Besides, to maintain the stability and consistency in the training phase, we improved the DLB regularization to make it suitable for the text-to-image retrieval domain. Extensive ablation studies were conducted and the experimental results on MSCOCO and Flickr30K datasets demonstrate the effectiveness of the proposed methods. Especially, our model achieved state-of-the-art results on the Flickr30K benchmark dataset. The code to reproduce our results is available at `https://github.com/moment-ggw/IDC/tree/main`. In the near future, we plan to design a more direct way that allows distillation of knowledge from Indirect Connection into Direct Connection.

6. Acknowledgements

This work was supported in part by the Fundamental Research Funds for the Central Universities (2232021A-10), National Natural Science Foundation of China (61903078), Shanghai Sailing Program (no. 22YF1401300), and Natural Science Foundation of Shanghai (20ZR1400400).

7. Bibliographical References

Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. 2021. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1728–1738.

Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. 2019. Multimodal Machine Learning: A Survey and Taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):423–443.

Simion-Vlad Bogolin, Ioana Croitoru, Hailin Jin, Yang Liu, and Samuel Albanie. 2022. Cross Modal Retrieval with Querybank Normalisation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5184–5195.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. UNITER: UNiversal Image-TExt Representation Learning. In *Computer Vision – ECCV 2020*, Lecture Notes in Computer Science, pages 104–120. Springer International Publishing.

Yuxin Chen, Zongyang Ma, Ziqi Zhang, Zhongang Qi, Chunfeng Yuan, Ying Shan, Bing Li, Weiming Hu, Xiaohu Qie, and Jianping Wu. 2023. Vilem: Visual-language error modeling for image-text retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11018–11027.

Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. 2020. Meshed-Memory Transformer for Image Captioning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10575–10584.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Satya Krishna Gorti, Noël Vouitsis, Junwei Ma, Keyvan Golestan, Maksims Volkovs, Animesh Garg, and Guangwei Yu. 2022. X-Pool: Cross-Modal Language-Video Attention for Text-Video Retrieval. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4996–5005.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum Contrast for Unsupervised Visual Representation Learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9726–9735.

Zhicheng Huang, Zhaoyang Zeng, Bei Liu, Dongmei Fu, and Jianlong Fu. 2020. Pixel-bert: Aligning image pixels with text by deep multi-modal transformers. *arXiv preprint arXiv:2004.00849*.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR.

Junyeong Kim, Minuk Ma, Trung Pham, Kyungsu Kim, and Chang D. Yoo. 2020. Modality Shifting Attention Network for Multi-Modal Video Question Answering. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10103–10112.

Gen Li, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang. 2020a. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11336–11344.

Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. 2020b. Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks. In *Computer Vision – ECCV 2020*, Lecture Notes in Computer Science, pages 121–137. Springer International Publishing.

Ilya Loshchilov and Frank Hutter. 2016. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

Haoyu Lu, Nanyi Fei, Yuqi Huo, Yizhao Gao, Zhiwu Lu, and Ji-Rong Wen. 2022. COTS: Collaborative Two-Stream Vision-Language Pre-Training Model for Cross-Modal Retrieval. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15671–15680.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Antoine Miech, Jean-Baptiste Alayrac, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 2021. Thinking Fast and Slow: Efficient Text-to-Visual Retrieval with Transformers. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9821–9831.

Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. 2020. X-Linear Attention Networks for Image Captioning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10968–10977.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings* *of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Yiqing Shen, Liwu Xu, Yuzhe Yang, Yaqian Li, and Yandong Guo. 2022. Self-Distillation from the Last Mini-Batch for Consistency Regularization. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11933–11942.

Siqi Sun, Yen-Chun Chen, Linjie Li, Shuohang Wang, Yuwei Fang, and Jingjing Liu. 2021. Lightningdot: Pre-training visual-semantic embeddings for real-time image-text retrieval. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 982–997.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Haoran Wang, Dongliang He, Wenhao Wu, Boyang Xia, Min Yang, Fu Li, Yunlong Yu, Zhong Ji, Errui Ding, and Jingdong Wang. 2022. CODER: Coupled Diversity-Sensitive Momentum Contrastive Learning for Image-Text Retrieval. In *Computer Vision – ECCV 2022*, Lecture Notes in Computer Science, pages 700–716. Springer Nature Switzerland.

Keyu Wen, Jin Xia, Yuanyuan Huang, Linyang Li, Jiayan Xu, and Jie Shao. 2021. COOKIE: Contrastive Cross-Modal Knowledge Sharing Pre-training for Vision-Language Representation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2188–2197.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. 2021. VinVL: Revisiting Visual Representations in Vision-Language Models. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5575–5584.

Qi Zhang, Zhen Lei, Zhaoxiang Zhang, and Stan Z. Li. 2020. Context-Aware Attention Network for Image-Text Retrieval. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3533–3542.

8. Language Resource References

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3128–3137.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. In *Computer Vision – ECCV 2014*, Lecture Notes in Computer Science, pages 740–755. Springer International Publishing.

Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2641–2649.