MCAD: Multi-teacher Cross-modal Alignment Distillation for efficient image-text retrieval

Anonymous ACL submission

Abstract

Due to the success of large-scale visual-language pretraining (VLP) models and the widespread use of image-text retrieval in industry areas, it is now critically necessary to reduce the model size and streamline their mobile-device deployment. Single- and dual-stream model structures are commonly used in image-text retrieval with the goal of closing the semantic gap between textual and visual modalities. While single-stream models use deep feature fusion to achieve more accurate cross-model alignment, dual-stream models are better at offline indexing and fast inference. We propose a Multi-teacher Cross-modality Alignment Distillation (MCAD) technique to integrate the advantages of single- and dual-stream models. By incorporating the fused single-stream features into the image and text features of the dual-stream model, we formulate new modified teacher similarity distributions and features. Then, we conduct both distribution and feature distillation to boost the capability of the student dual-stream model, achieving high retrieval performance without increasing inference complexity. Extensive experiments demonstrate the remarkable performance and high efficiency of MCAD on image-text retrieval tasks. Furthermore, we implement a lightweight CLIP model on Snapdragon/Dimensity chips with only ∼100M running memory and ∼8.0ms search latency, achieving the mobile-device application of VLP models.

1 Introduction

Image-text mutual retrieval is a fundamental problem of multimodal learning, whose primary objective is to bridge the semantic gap between visual and textual modalities, enabling accurate match of image (text) based on the given text (image). However, aligning and matching visual and textual information is non-trivial due to the differences in their representations and structures. In recent years, the rapid growth of large-scale paired vision-language datasets (Schuhmann et al., 2021, 2022) has paved the way for the development of powerful models that can bridge the gap between visual and textual information. These models, known as vision-language pretraining (VLP) models, have shown remarkable capabilities in understanding both vision and language (Radford et al., 2021; Jia et al., 2021; Li et al., 2021).

Typically, the dual-stream architecture, *e.g.*, CLIP (Radford et al., 2021), ALIGN (Jia et al., 2021), facilitates autonomous processing of individual modalities through segregated streams, exhibiting inferior retrieval performance due to the lack of effective cross-modal feature fusion. Nevertheless, the disentanglement of image and text encoder enable fast retrieval speed. On the contrary, single-stream models integrate information from multiple modalities during encoding through a deep interaction module, *e.g.*, transformer block (Vaswani et al., 2017), commonly leading to superior retrieval performance but sacrificing flexibility and resulting in extremely low retrieval speed. Therefore, in industrial applications, dual-stream models are still the first choice. However, their significant size hinders practical deployment in lightweight scenarios, especially for mobile devices.

In recent years, several works endeavor to transfer knowledge of large models into small models through distillation technology (Fang et al., 2021a; Wang et al., 2022a; Rao et al., 2023; Ren and Zhu, 2022; Wang et al., 2022b; Miech et al., 2021; Lei et al., 2022; Wu et al., 2023; Vasu et al., 2023). But they just consider soft-label, feature, or attention map distillation from one teacher or homogeneous teachers. The strategy of homogeneous, multi-teacher distillation has not yet been explored. Among these works, a critical question is how to distill the knowledge of the single-stream models into efficient dual-stream models. Although DIDE (Wang et al., 2022b) proposes to employ

cross-modal attention distillation to transfer the knowledge of the ViLT (Kim et al., 2021) teacher to a CLIP student, this is not a universal method since other single-stream structures, *e.g.*, ALBEF (Li et al., 2021) cannot discriminate explicit image and text features after cross-attention fusion so that the image-text attention maps are unavailable. Another work LoopITR (Lei et al., 2022) considers only employ the single-stream output scores of top k hard examples chosen by the dual-stream model to enhance the dual-stream model itself, which cannot excite the whole ability of the single-stream model. So the integration of single-stream and dual-stream teachers is a non-trivial challenge. In this paper, we are motivated to propose a Multi-teacher Cross-modal Alignment Distillation (MCAD) method to make full use of the information fusion ability of the single-stream model and the large-scale parallel training advantage of the dual-stream model. Specifically, after extracting features through the frozen single- and dual-stream teacher models, we apply different learnable projection layers to align image or text features from different latent spaces, as shown in Fig. 2. Finally, we employ similarity distribution and feature distillation based on the newly-formulated fused features to boost the performance of the dual-stream student model, as shown in Fig. 1. In summary, our main contributions are as follows:

- We propose a single- and dual-stream multi-teacher distillation algorithm to enhance the cross-modal retrieval ability of a light-weight CLIP-like dual-stream model.
- Comprehensive experiments on different datasets and networks demonstrate that our method is a model-agnostic general framework that can achieve superior performance both in zero-shot and fine-tuning settings.
- By using MobileViTv2 (Mehta and Rastegari, 2022) and TinyBERT (Jiao et al., 2020) as the image and text encoder, respectively, we compress a 400M large CLIP model onto Snapdragon/Dimensity chips, achieving merely 25.9M model size, ∼100M running memory, and ∼8.0ms retrieval latency.

2 Related Work

2.1 Image-Text Retrieval with VLP

Image-text retrieval (ITR) has attracted increasing attention in recent years. In recent years, cross-modal pre-training has been extensively studied and applied to ITR (Liu et al., 2019; Lu et al., 2019; Chen et al., 2020; Wang et al., 2022c). The model structure can be roughly classified into two categories: single-stream and dual-stream. Single-stream models jointly encode images and text through a deep interaction module and output a fused feature. Early algorithms (Lu et al., 2019) employ object detectors (Girshick, 2015; Ren et al., 2015) to extract image features, which usually ignore important background information. Then, ViLT (Kim et al., 2021; Diao et al., 2021) unifies image and text extractor as Transformer (Vaswani et al., 2017) to make full use of all information. The models, however, depend on a cross-modal Transformer encoder to fuse visual and textual signals at the same time across layers, which necessitates a large compute budget and slows down inference speed. Even though some trade-off approaches, *e.g.*, ALBEF (Li et al., 2021), employ separate image and text encoders prior to hard example fusion, their top k re-ranking strategy is still far from being implemented in real time.

On the contrary, the dual-stream model mainly focuses on learning how to align visual and textual features obtained from independent encoders. Since only a light-weight interaction module (usually a MLP or dot product) is applied to image and text features, dual-stream structure allows for contrastive learning on billions of examples, including CLIP (Radford et al., 2021) and ALIGN (Jia et al., 2021). Thanks to the shadow interaction module, all visual or textual features can be pre-calculated and stored offline, leading to a fast retrieval speed. Nevertheless, due to a lack of deep cross-model fusion, the visual-language understanding ability of dual-stream models is inferior to that of single-stream models, resulting in lower retrieval accuracy. Hence, we are inspired to transfer the advantages of single- and dual-stream models into a compressed, lightweight model through our proposed innovative distillation technique.

2.2 Knowledge Distillation for VLP

Knowledge Distillation (Hinton et al., 2015) is a method of transferring knowledge from a teacher model to a student model, which can effectively improve the performance of the student model (Lan et al., 2020; Yalniz et al., 2019; Touvron et al., 2021; Fang et al., 2021b). In the multimodal distillation area, a group of approaches considers transferring knowledge from large models into small models with the same architecture, either both

single-stream models (Fang et al., 2021a; Wang et al., 2022a; Rao et al., 2023) or dual-stream models (Ren and Zhu, 2022; Wu et al., 2023) by using logit, feature, or attention map distillation. DistillVLM and EfficientVLM (Wang et al., 2022a) propose attention map distillation and hidden feature distillation for object-detection-based and ALBEF-like (Li et al., 2021) single-stream VLP model compression, respectively. TinyCLIP (Wu et al., 2023) trains lightweight CLIP models via cross-modal affinity mimicking (similarity distribution distillation) and weight inheritance.

Another group of methods employ single-stream models to improve performance of dual-stream models (Wang et al., 2022b; Miech et al., 2021; Lei et al., 2022). DIDE (Wang et al., 2022b) applies cross-model attention distillation to transfer knowledge of a single-stream ViLT (Kim et al., 2021) teacher model into a CLIP-like dual-stream student model. LoopITR (Lei et al., 2022) proposes a mutual-loop enhancement strategy to distill dual-stream models by top hard samples of single-stream models. Thinking fast and slow (Miech et al., 2021) improves dual-stream model performance by single-stream model via logit distillation.

As multi-teacher distillation (Yang et al., 2020; Gou et al., 2021; Zhao et al., 2022; Zhang et al., 2023) has been generally regarded as an effective approach to improving student models, MobileCLIP (Vasu et al., 2023) proposes to employ ensemble of K CLIP models as a strong teacher. To the best of our knowledge, our MCAD is the first work that uses heterogeneous multi-teachers to distill the advantages of single- and dual-stream models into a lightweight student VLP model.

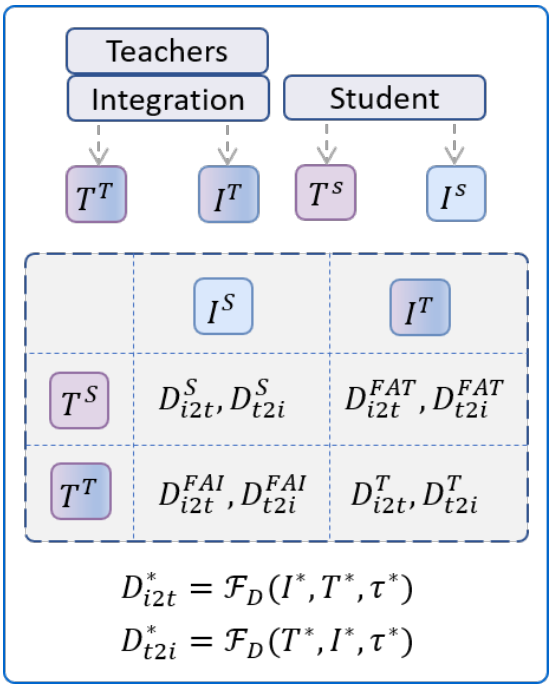

Figure 1: An overview of our MCAD framework. (I^T, T^T) and (I^S, T^S) represent the (image, text) feature pair output by teachers and the student, respectively. D^*_{i2t} represents the similarity distribution of image-to-text, while D^*_{t2i} denotes that of text-to-image. D^S_* indicates the distribution matrix produced by the student, while D^T_* depicts that derived from the aggregated teachers. Additionally, D^{FAI}_* denotes the softmax output after cross-feature alignment between the student's image feature and the teachers' text feature, while D^{FAT}_* represents the corresponding operation after aligning the student's text feature to the teachers' image feature.

3 Method

3.1 Preliminary

We first define the general form for calculating the similarity distribution matrix and the KL divergence loss, then we will introduce the distribution matrices shown in Fig. 1.

The general form to calculate the image-text similarity distribution matrix can be denoted as:

$$\mathcal{F}_D(I, T, \tau) = \mathrm{softmax}[(IT^\top)/\tau], \qquad (1)$$

where $\mathrm{softmax}(\cdot)$ represents the softmax function that operates in the last dimension, I, T denote the normalized image and text representations, respectively, with shape $[n, d]$, where n is the batch size and d is the output dimension, and τ is a temperature parameter. Moreover, the row-wise KL divergence between two distribution matrices D and $\hat{D}$ can be denoted as:

$$\mathcal{F}_{KL}(D, \hat{D}) = \sum_l \mathrm{KL}(D_l || \hat{D}_l), \qquad (2)$$

where l indicates the row index.

Fig. 1 shows an overview of the MCAD framework, which combines single- and dual-stream models at the token level. Given n image-text pair inputs in a batch, $\{(i_j, t_j)\}_1^n$, we will get image representations $I^T \in \mathbb{R}^{n\times d}$ and text representations $T^T \in \mathbb{R}^{n\times d}$ after feeding the output of multiple teachers to the integration module, which will be detailedly discussed in Sec. 3.3. Moreover, the student's image encoder and text encoder output the image representation $I^S \in \mathbb{R}^{n\times d}$ and text representation $T^S \in \mathbb{R}^{n\times d}$, respectively. After that, several distribution matrices shown in Fig. 1 can

be expressed as:

$$
\begin{aligned}
D_{i2t}^{S} &= \mathcal{F}_D(I^S, T^S, \tau^S), \\
D_{i2t}^{T} &= \mathcal{F}_D(I^T, T^T, \tau^T), \\
D_{i2t}^{FAI} &= \mathcal{F}_D(I^S, T^T, (\tau^T + \tau^S)/2), \\
D_{i2t}^{FAT} &= \mathcal{F}_D(I^T, T^S, (\tau^T + \tau^S)/2),
\end{aligned} \tag{3}
$$

where D_{i2t}^{S} denotes the similarity distribution of image-to-text output by student, while that of text-to-image simply involves swapping the input positions, *i.e.*, $D_{t2i}^{S} = \mathcal{F}_D(T^S, I^S, \tau^S)$. The detailed description of D_*^* can be found in the caption of Fig. 1.

3.2 Multi-teacher Cross-modal Alignment Distillation

Dual-stream target. Assuming a collection of n image-text pairs $\{(i_j, t_j)\}_1^n$ in a batch, the text and image features of the dual-stream teacher model are denoted as $I^{DS} \in \mathbb{R}^{n\times d}$ and $T^{DS} \in \mathbb{R}^{n\times d}$, respectively, and the target similarity distribution matrix can be expressed as:

$$
\begin{aligned}
D_{i2t}^{DS} &= \mathcal{F}_D(I^{DS}, T^{DS}, \tau^{DS}), \\
D_{t2i}^{DS} &= \mathcal{F}_D(T^{DS}, I^{DS}, \tau^{DS}),
\end{aligned} \tag{4}
$$

where τ^{DS} denotes the temperature of the dual-stream model.

Single-stream target. Besides the straightforward format of dual-stream target distributions, we also need to calculate the single-stream target. Subsequently, the indices of the top k similarity scores are first computed based on Eq. (4), which can be represented as:

$$
\begin{aligned}
P_{i2t} &= \text{topK\_indices}(D_{i2t}^{DS}), \\
P_{t2i} &= \text{topK\_indices}(D_{t2i}^{DS}),
\end{aligned} \tag{5}
$$

where P_{i2t} denotes the indices of each image and the top k texts that are similar to it, while P_{t2i} represents the indices of each text and its k most similar images. Then, we recalculate the scores of the top k image-text pairs by the single-stream model, *e.g.,* ALBEF (Li et al., 2021). We assume that the score matrices output by the single-stream model are $D_{i2t}^{SS} \in \mathbb{R}^{n\times n}$, $D_{t2i}^{SS} \in \mathbb{R}^{n\times n}$, which are calculated as:

$$
\begin{aligned}
(D_{i2t}^{SS})_{l,m} &= f_{SS}(i_l, t_m), \quad (l,m) \in P_{i2t}, \\
(D_{t2i}^{SS})_{l,m} &= f_{SS}(i_m, t_l), \quad (l,m) \in P_{t2i}.
\end{aligned} \tag{6}
$$

In general, a single-stream model will usually output a similarity score for the current image-text pair. It should be noted that in matrix D_{i2t}^{SS} and D_{t2i}^{SS}, only $D_{i2t}^{SS}[P_{i2t}] \in \mathbb{R}^{n\times k}$, $D_{t2i}^{SS}[P_{t2i}] \in \mathbb{R}^{n\times k}$ are computed, and we only care about this part.

Loss function. The objective of this paper is to introduce the MCAD technique for effectively merging single- and dual-stream models. The ultimate goal is to enable effective knowledge transfer from multiple teachers to the student network. We adopt a dual-stream architecture for the student network, which results in improved retrieval speed for image-text tasks. In doing so, the proposed method can be more conveniently deployed on mobile devices. In this study, the uniform loss function is denoted as:

$$
\mathcal{L}_{total} = \mathcal{L}_{\mathcal{TDD}} + \mathcal{L}_{\mathcal{TFD}}, \tag{7}
$$

where $\mathcal{L}_{\mathcal{TDD}}$ denotes the loss function of target distribution distillation (TDD), and $\mathcal{L}_{\mathcal{TFD}}$ denotes the target feature distillation (TFD).

First, the $\mathcal{L}_{\mathcal{TDD}}$ of multi-teachers can be expressed as:

$$
\begin{aligned}
\mathcal{L}_{\mathcal{TDD}} : \mathcal{L}_{MT} = f_{MT}(D_{i2t}^{S}, D_{t2i}^{S}) \\
+ f_{MT}(D_{i2t}^{T}, D_{t2i}^{T}),
\end{aligned} \tag{8}
$$

where f_{MT} is a loss function that measures the KL divergence between the output and the target distribution, including dual-stream and single-stream targets as mentioned before. Importantly, the second term brings the output similarity distribution of the integration module (discussed in Sec. 3.3) close to the target distribution, which can be viewed as a regularization of the integration module. Second, the $\mathcal{L}_{\mathcal{TFD}}$ of multi-teachers is denoted as:

$$
\begin{aligned}
\mathcal{L}_{\mathcal{TFD}} : \mathcal{L}_{MT\_FA} = f_{MT}(D_{i2t}^{FAI}, D_{t2i}^{FAI}) \\
+ f_{MT}(D_{i2t}^{FAT}, D_{t2i}^{FAT}),
\end{aligned} \tag{9}
$$

where the two terms bring the representation of the student output close to the fused feature.

Finally, the core loss function f_{MT} is defined as:

$$
\begin{aligned}
f_{MT}(D_{i2t}^{*}, D_{t2i}^{*}) &= \mathcal{F}_{KL}(D_{i2t}^{*}, D_{i2t}^{DS}) \\
&+ \mathcal{F}_{KL}((D_{t2i}^{*}, D_{t2i}^{DS}) \\
&+ \mathcal{F}_{KL}(\sigma(D_{i2t}^{*}[P_{i2t}]), \sigma(D_{i2t}^{SS}[P_{i2t}])) \\
&+ \mathcal{F}_{KL}(\sigma(D_{t2i}^{*}[P_{t2i}]), \sigma(D_{t2i}^{SS}[P_{t2i}])),
\end{aligned} \tag{10}
$$

where $* \in \{S, T, FAI, FAT\}$ and $\sigma(\cdot)$ is a normalization method. When given a matrix $D^* \in \mathbb{R}^{n\times k}$, the normalization method can be expressed as:

$$
\begin{aligned}
\sigma(D_{l,m}^{*}) &= \frac{D_{l,m}^{*}}{\sum_{v=1}^{k} D_{l,v}^{*}}, \\
l \in [1, .., n], m &\in [1, ..., k]
\end{aligned} \tag{11}
$$

Here, we don't directly align the student output with the integrated-teacher output. Instead, we align the output of the student and the integrated teacher with the dual- and single-stream teacher simultaneously, as Eq. (8)–(10) show. Since the integrated teacher also contains learnable parameters (will be introduced in Sec. 3.3), we regard the dual- and single-stream as pivots to align the student and the integrated teacher.

3.3 Multi-teacher Integration

To better utilize the features of multi-models, we propose a framework to integrate the output of different models, which is shown in Fig. 2. When giving an image-text pair $(i_l, t_m), l, m \in [1, ..., n]$, suppose that I_l^{DS}, T_m^{DS} represent the "CLS" token output by dual-stream's image encoder and text encoder, respectively, and H_{m-l}^{SS} represents the "CLS" token output by the single-stream model. The "CLS" token outputs by multiple teachers are first projected into other vectors by a different function, g_*. Especially, although the single-stream model has only one "CLS" token, it still has to be projected to different spaces using two different functions, *i.e.,* g_1, g_2. And in this study, the g_1, g_2 function can be denoted as follows:

$$g_{1/2}(\cdot) = \begin{cases} f_P(H_{m-l}^{SS}), & if\ (l,m) \in P_{i2t} \\ & or\ (m,l) \in P_{t2i} \\ 0, & if\ (l,m) \notin P_{i2t} \\ & and\ (m,l) \notin P_{t2i}, \end{cases} \quad (12)$$

where f_P represents a projection layer. Moreover, g_1 and g_2 play the role of a gate. Because we fuse single- and dual-stream models to adjust the distribution of the top k, all we need to do is to fuse two teachers' features only on the top k.

Finally, the output of the text representation T_m^T and image representation I_l^T can be expressed as:

$$\begin{aligned} T_m^T &= \text{norm}(g_3(T_m^{DS-T}) + \alpha \cdot g_1(H_{m-l}^{SS})) \\ I_l^T &= \text{norm}(g_4(I_l^{DS-I}) + \alpha \cdot g_2(H_{m-l}^{SS})), \end{aligned} \quad (13)$$

where α is a learnable parameter, and norm represents the ℓ_2 normalization operator.

4 Experiments

4.1 Datasets

We utilize existing image-text pair datasets to verify our method, including MSCOCO (Lin et al., 2014), Conceptual Captions (CC) (Sharma et al., 2018), SBU captions (Ordonez et al., 2011), and Flickr30K (Plummer et al., 2015). To test the zero-shot capability of our method, we only combine CC and SUB as training datasets, while in fine-tuning experiments, we use all four data datasets during training. For validation and testing, we utilize the standard split (Karpathy and Fei-Fei, 2015) of COCO and Flickr. More details of the datasets and training hyper-parameters are presented in Appendix A and B, respectively.

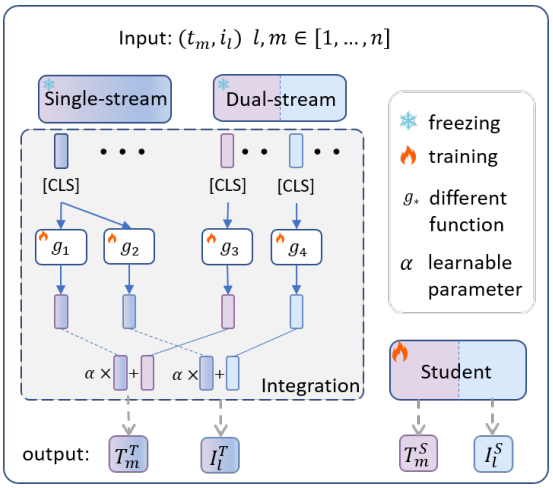

Figure 2: Details of the integration module

4.2 Baselines and Components

In this paper, as shown in Eq. (7), we propose a general loss function by dividing it into two parts: target distribution distillation ($\mathcal{L}_{\mathcal{TDD}}$) and target feature distillation ($\mathcal{L}_{\mathcal{TFD}}$) losses. We consider the first component to be the process of allowing the student's image-text similarity output to approximate a desired distribution. In terms of the second component, we can align the student's feature with the teacher's feature by following different constraints. Several prior works can be viewed as special cases of the general form proposed in Eq. (7). For the target distribution distillation, the categories can be summarized as follows:

Ground truth. Given n image-text pairs $\{(i_l, t_l)\}_1^n$, the student model outputs two matrix $D_{i2t}^S \in \mathbb{R}^{n \times n}, D_{t2i}^S \in \mathbb{R}^{n \times n}$, and the ground truth can be denoted as D^{GT}, which is an identity matrix. Then, the loss function using the ground truth as the target distribution can be expressed as:

$$\begin{aligned} \mathcal{L}_{TDD} : \mathcal{L}_{GT} = &\mathcal{F}_{KL}(D_{i2t}^S, D^{GT}) \\ &+ \mathcal{F}_{KL}(D_{t2i}^S, D^{GT}) \end{aligned} \quad (14)$$

Moreover, $\mathcal{L}_{GT}$ is also called a hard target in the study (Hinton et al., 2015). This can be viewed as

a baseline for training dual-stream models without teacher distillation.

Dual-stream distribution distillation. In this form, CLIP is often used as the dual-stream teacher, and the distribution of CLIP output is denoted as $D^{DS}_{i2t}, D^{DS}_{t2i}$. Then, a uniform loss function for dual-stream distillation is introduced as follows:

$$\begin{aligned} &\mathcal{L}_{TDD} : \mathcal{L}_{CLIP} = f_{DS}(D^{S}_{i2t}, D^{S}_{t2i}) \\ &= \mathcal{F}_{KL}(D^{S}_{i2t}, D^{DS}_{i2t}) + \mathcal{F}_{KL}(D^{S}_{t2i}, D^{DS}_{t2i}). \end{aligned} \tag{15}$$

This loss has been widely used in pioneering studies, including Leaner and Faster (Ren and Zhu, 2022) which combin $\mathcal{L}_{GT}$ and $\mathcal{L}_{CLIP}$ as $\mathcal{L}_{\mathcal{TDD}}$ and TinyCLIP (Wu et al., 2023) which solely uses $\mathcal{L}_{CLIP}$ as the distillation loss. We reimplement these works based on our model structures and datasets for fair comparisons.

Single-stream distribution distillation. Similarly, ALBEF (Li et al., 2021) is also widely employed as the single-stream teacher. Then we construct the distribution of the ALBEF output by Eq. (6). Due to the limitation of computing resources, we only calculate the top k text similarities that are most similar to each image. Similarly, each text is treated in the same way. It should be noted that the information of top k is provided by a dual-stream model, *i.e.*, CLIP. Then, the loss function using only the single-stream model, *i.e.*, ALBEF, can be denoted as follows:

$$\begin{aligned} &\mathcal{L}_{TDD} : \mathcal{L}_{ALBEF} = \\ &\mathcal{F}_{KL}(\sigma(D^{S}_{i2t}[P_{i2t}]), \sigma(D^{SS}_{i2t}[P_{i2t}])) \\ &+\mathcal{F}_{KL}(\sigma(D^{S}_{t2i}[P_{t2i}]), \sigma(D^{SS}_{t2i}[P_{t2i}])). \end{aligned} \tag{16}$$

In LoopITR (Lei et al., 2022) and Thinking Fast and Slow (Miech et al., 2021), the authors employ $\mathcal{L}_{ALBEF} + \mathcal{L}_{GT}$ as $\mathcal{L}_{\mathcal{TDD}}$ and we also implement them as comparison methods.

Dual-stream feature distillation. In terms of target feature distillation, the dual-stream model outputs the image representation I^{DS} and text presentations T^{DS} separately. Then, we can align the student's feature with the teacher's feature by constructing the following equation:

$$\begin{aligned} D^{FAI'}_{i2t} &= \mathcal{F}_D(I^S, T^{DS}, (\tau^{DS} + \tau^S)/2) \\ D^{FAI'}_{t2i} &= \mathcal{F}_D(T^{DS}, I^S, (\tau^{DS} + \tau^S)/2) \\ D^{FAT'}_{i2t} &= \mathcal{F}_D(I^{DS}, T^S, (\tau^{DS} + \tau^S)/2) \\ D^{FAT'}_{t2i} &= \mathcal{F}_D(T^S, I^{DS}, (\tau^{DS} + \tau^S)/2) \\ \mathcal{L}_{TFD} &: \mathcal{L}_{CLIP\_FA} = f_{DS}(D^{FAI'}_{i2t}, D^{FAI'}_{t2i}) \\ &+ f_{DS}(D^{FAT'}_{i2t}, D^{FAT'}_{t2i}) \end{aligned} \tag{17}$$

where $\mathcal{L}_{CLIP\_FA}$ indicates that we align the student features with the dual-stream teacher features, *i.e.*, CLIP, and f_{DS} is defined in Eq. (15).

Multi-teacher distillation. Our motivation for integrating the multi-teachers' output distributions is to gain a better distribution to distill the student model. Since single-stream models tend to perform better than dual-stream models, we argue that single-stream models can better distinguish difficult samples that cannot be discriminated against by dual-stream models. So, the final loss for measuring the distribution gap between student and multi-teacher is expressed as Eq. (8). Furthermore, to align the student output features with the multi-teacher fused features, the loss function for feature alignment is expressed as Eq. (9).

4.3 Zero-shot Experiments and Ablations

For the student's image and text encoder, we utilize MobileViTv2 (Mehta and Rastegari, 2022) and TinyBERT (Jiao et al., 2020), with 11.19 M and 14.71 M parameters, respectively. We also employ CLIP (ViT-L/14) and ALBEF as the teacher models, containing approximately 427.62M and 419.12M parameters, respectively. In order to evaluate the generalizability of the student model, we train it on both the CC3M and SBU datasets and subsequently assess its performance on the COCO and Flickr30k testsets. Moreover, the default value of k is 11. All zero-shot results for comparisons with the aforementioned baselines and ablation studies are obtained using the checkpoint associated with the highest validation performance and presented in Table 1.

As mentioned in Sec. 4.2, we propose a uniform loss paradigm for image-text retrieval distillation approaches. For fair comparisons, we reimplement several baseline methods based on the same model structures and datasets. Specifically, for TinyCLIP (Wu et al., 2023), we adopt its *affinity mimicking* loss (equivalent to $\mathcal{L}_{CLIP}$) and *uniformly manual inheritance* for the TinyBERT text encoder. For Leaner and Faster (Ren and Zhu, 2022), we adopt its $\mathcal{L}_{CLIP}+\mathcal{L}_{GT}$ loss while eliminating the $\mathcal{L}_{HN}$ item since it's orthogonal to distillation approaches. For LoopITR (Lei et al., 2022), we employ its $\mathcal{L}_{ALBEF}+\mathcal{L}_{GT}$ as the loss function. In addition, we also conduct comprehensive ablation studies based on our loss components, as illustrated in Table 1.

Several observations can be drawn from the statistics. 1) Compared to training without teach-

Table 1: Zero-shot comparisons and ablations on MSCOCO and Flickr30K testsets of the student model that uses the mobileViTv2 and TinyBERT as backbones. * indicates that we reimplement the comparable approaches based on our model structures and datasets. # denotes the value of ALBEF are copied from the original paper.

methods	Loss $\mathcal{L}_{total} = \mathcal{L}_{\mathcal{TDD}} + \mathcal{L}_{\mathcal{TFD}}$		Flickr30K						MSCOCO					
			Image Retrieval			Text Retrieval			Image Retrieval			Text Retrieval		
	$\mathcal{L}_{\mathcal{TDD}}$	$\mathcal{L}_{\mathcal{TFD}}$	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP	$\mathcal{L}_{GT}$		85.2	97.5	99.1	64.9	87.2	92.1	56.3	79.4	86.7	36.7	61.4	71.5
ALBEF #	loss of ALBEF		94.1	99.5	99.7	82.8	96.3	98.1	-	-	-	-	-	-
no teachers	$\mathcal{L}_{GT}$	-	43.5	70.5	78.9	31.0	58.1	69.2	21.9	44.4	56.9	16.4	37.0	58.6
(Lei et al., 2022)*	$\mathcal{L}_{ALBEF} + \mathcal{L}_{GT}$	-	54.3	81.1	88.2	40.5	70.2	79.3	30.5	56.5	67.2	21.5	45.6	57.5
(Ren and Zhu, 2022)*	$\mathcal{L}_{CLIP} + \mathcal{L}_{GT}$	-	56.2	80.2	87.9	40.5	67.8	77.2	29.7	55.9	67.5	20.7	43.4	55.0
(Wu et al., 2023)*	$\mathcal{L}_{CLIP}$	-	60.1	82.4	89.2	41.3	69.2	78.5	31.9	58.1	67.9	21.1	44.3	56.0
Ablations of ours	$\mathcal{L}_{CLIP}$	-	61.3	84.6	90.9	43.6	71.0	80.5	33.8	59.8	70.8	21.8	45.2	57.1
	$\mathcal{L}_{ALBEF}$	-	39.7	69.2	77.8	29.9	58.0	69.6	22.5	46.9	60.1	15.5	36.4	48.2
	$\mathcal{L}_{MT}$	-	63.5	85.9	91.7	47.9	76.7	84.3	37.6	63.3	74.6	25.7	51.2	62.8
	-	$\mathcal{L}_{CLIP\_FA}$	60.5	83.8	89.5	41.9	69.9	79.3	32.6	57.7	68.7	21.1	43.6	55.2
	-	$\mathcal{L}_{MT\_FA}$	61.3	85.0	90.9	46.0	74.7	83.6	37.1	63.4	73.6	25.4	50.3	62.1
	$\mathcal{L}_{CLIP}$	$\mathcal{L}_{CLIP\_FA}$	64.3	87.0	92.2	46.2	74.3	82.7	35.7	62.4	72.6	23.5	46.9	58.7
	$\mathcal{L}_{CLIP}$	$\mathcal{L}_{MT\_FA}$	65.2	87.3	92.5	50.3	77.6	85.1	37.2	64.0	73.9	26.0	51.9	62.9
	$\mathcal{L}_{MT}$	$\mathcal{L}_{CLIP\_FA}$	66.0	**88.1**	**92.7**	51.1	78.3	85.7	37.7	64.4	74.6	26.6	52.1	63.3
Ours	$\mathcal{L}_{MT}$	$\mathcal{L}_{MT\_FA}$	**66.6**	87.4	**92.7**	**52.1**	**78.9**	**86.6**	**38.6**	**65.0**	**75.2**	**27.3**	**52.7**	**64.0**

Table 2: Finetuing ablations on MSCOCO and Flickr30K testsets of the student model that uses the MobileViTv2 and TinyBERT as backbones. # denotes the value of ALBEF are copied from the original paper.

methods	Loss $\mathcal{L}_{total} = \mathcal{L}_{\mathcal{TDD}} + \mathcal{L}_{\mathcal{TFD}}$		Flickr30K						MSCOCO					
			Image Retrieval			Text Retrieval			Image Retrieval			Text Retrieval		
	$\mathcal{L}_{\mathcal{TDD}}$	$\mathcal{L}_{\mathcal{TFD}}$	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP	$\mathcal{L}_{GT}$	-	95.3	99.7	100.0	84.0	97.0	98.7	74.2	92.3	96.0	57.3	81.8	88.7
ALBEF #	loss of ALBEF		95.9	99.8	100.0	85.6	97.5	98.9	77.6	94.3	97.2	60.7	84.3	90.5
Ablations of ours	-	$\mathcal{L}_{CLIP\_FA}$	75.5	92.7	96.9	59.1	85.3	91.4	48.8	75.4	84.3	36.5	65.8	76.8
	$\mathcal{L}_{CLIP}$	-	78.7	93.6	96.8	61.6	86.6	92.0	54.4	79.7	88.1	39.0	68.0	78.8
	$\mathcal{L}_{CLIP}$	$\mathcal{L}_{CLIP\_FA}$	79.2	94.3	97.4	62.8	87.3	92.6	54.0	79.9	87.8	39.3	68.1	78.8
Ours	$\mathcal{L}_{MT}$	$\mathcal{L}_{MT\_FA}$	**80.2**	**95.9**	**97.8**	**64.1**	**88.4**	**93.4**	**55.0**	**80.4**	**88.2**	**40.2**	**69.2**	**79.5**

ers ($\mathcal{L}_{GT}$), the CLIP target distribution distillation ($\mathcal{L}_{CLIP}$) can bring more effective information, but the result will not be further improved when combining them together ($\mathcal{L}_{CLIP} + \mathcal{L}_{GT}$). This indicates that the ground truth (usually very noisy) is not a good distribution when distilling the student model. 2) When we solely use the distribution of top k output by ALBEF ($\mathcal{L}_{ALBEF}$), it does not work very well, revealing that we also need to take into account distributions of more negative samples. When we use both the distribution of ALBEF output and the ground truth ($\mathcal{L}_{ALBEF} + \mathcal{L}_{GT}$), the results are much better than using the ground truth alone, which shows that it is necessary to readjust the distribution of top k. 3) When you combine the distribution of multi-teachers ($\mathcal{L}_{MA}$), it is more effective than any single teacher. 4) Moreover, when distillation on both feature and output distribution of CLIP ($\mathcal{L}_{CLIP} + \mathcal{L}_{CLIP\_FA}$), it works better than distillation using only the similarity distribution ($\mathcal{L}_{CLIP}$), which demonstrates that aligning the student's features to the teacher's features improves student performance. 5) Furthermore, the best results are achieved when using a multi-teacher distribution and aligning the student features to the fused multi-teacher features ($\mathcal{L}_{MA} + \mathcal{L}_{MA\_FA}$). This is a good proof of the effectiveness of our multi-teacher cross-modal alignment distillation framework since both target distribution distillation and target feature distillation are important.

4.4 Finetuning Experiments

To further verify the finetuning performance of our approach, we first finetune the teacher models and then perform different distillation strategies on the MSCOCO and Flickr30K training datasets. All results are shown in Table 2, which maintains the same conclusion as before. We achieve the best results when combining multi-teacher distribution distillation ($\mathcal{L}_{MT}$) and feature distillation ($\mathcal{L}_{MT\_FA}$), surpassing dual-stream distribution distillation ($\mathcal{L}_{CLIP}$), feature distillation ($\mathcal{L}_{CLIP\_FA}$) and combining them together ($\mathcal{L}_{CLIP} + \mathcal{L}_{CLIP\_FA}$).

4.5 Backbone and Hyper-parameter Selection

Since our method is a network-agnostic framework, we replace the image encoder and text encoder with MobileViTv3 (Wadekar and Chaurasia, 2022) and ALBERT (Lan et al., 2020), with 5.5M and 12.2M

Table 3: Zero-shot performance on MSCOCO and Flickr30K testsets by employing MobileViTv3 and ALBERT as image and text encoder, respectively.

$\mathcal{L}_{total} = \mathcal{L}_{\mathcal{TDD}}$+$\mathcal{L}_{\mathcal{TFD}}$		Flickr30K						MSCOCO					
		Image Retrieval			Text Retrieval			Image Retrieval			Text Retrieval		
$\mathcal{L}_{\mathcal{TDD}}$	$\mathcal{L}_{\mathcal{TFD}}$	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
$\mathcal{L}_{CLIP}$	$\mathcal{L}_{CLIP\_FA}$	62.0	86.2	91.8	45.7	73.9	82.4	35.7	62.0	72.7	23.4	48.1	59.9
$\mathcal{L}_{MT}$	$\mathcal{L}_{MT\_FA}$	**64.8**	**88.0**	**93.9**	**49.6**	**77.5**	**85.8**	**35.9**	**63.3**	**74.5**	**26.0**	**51.6**	**63.4**

Table 4: The student model's performance is assessed with various hyper-parameters k through zero-shot evaluations on MSCOCO and Flickr30K testsets by using mobileViTv2 and TinyBERT as image and text encoder, respectively.

	Flickr30K						MSCOCO					
$\mathcal{L}_{\mathcal{TDD}}$: $\mathcal{L}_{MA}$	Image Retrieval			Text Retrieval			Image Retrieval			Text Retrieval		
$\mathcal{L}_{\mathcal{TFD}}$: $\mathcal{L}_{MA\_FA}$	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
$k = 5$	64.3	**87.5**	92.6	50.8	78.1	85.9	**38.8**	65.0	74.8	26.6	52.0	63.5
$k = 11$	**66.6**	87.4	92.7	**52.1**	**78.9**	**86.6**	38.6	**65.0**	**75.2**	**27.3**	**52.7**	**64.0**
$k = 17$	63.3	86.9	**93.0**	50.6	78.6	86.0	37.5	64.6	75.1	26.6	52.1	63.5

Table 5: Scan speed, retrieval speed and running memory of different models based on 100,000 candidate images. * indicates that ALBEF selects the top 128 candidates for the fusion module calculation.

Model	image encoder	text encoder	fusion module	param. #	scan time	retrie. time	running mem.	platform
CLIP	VIT-L/14	12-layer transformer	dot product	427.62M	11.0ms	32.5ms	∼2GB	V100 GPU
ALBEF	VIT-B/16	6-layer transformer	6-layer transformer	419.12M	7.6ms	1945ms*	∼3GB	V100 GPU
ours	mobileVitV2-1.5	TinyBERT	dot product	25.9 M	3.8ms	14.1ms	∼150MB	V100 GPU
					24.5ms	8.5ms	93MB	Snapdragon 8 Gen3
					24.8ms	7.5ms	107MB	Dimensity 9300

parameters, respectively, to validate its generality. All zero-shot results are shown in Table 3. The statistics show that our approach still outperforms $\mathcal{L}_{CLIP} + \mathcal{L}_{CLIP\_FA}$, revealing that our proposed method is general to different dual-stream models.

Further, we conduct several experiments on the selection of k, with results shown in Table 4, which illustrates the impact of the hyper-parameter k on the distillation effect. Specifically, a lower R@1 score for the Flickr30k data is observed when k is set to 5 due to the diminished information received from ALBEF. Conversely, when k is increased to 17, the distribution of information from ALBEF becomes smoother, impeding the student model's ability to learn more accurate information. Notably, this aforesaid effect is most pronounced in the R@1 scores. Therefore, it is essential to select an appropriate value of k to enhance the performance of the student model. Finally, we choose an optimal $k = 11$ for all experiments.

4.6 Mobile-device Application

Table 5 tests the performance of the lightweight model deployed on Snapdragon 8 Gen3 and MTK Dimensity 9300 chips, which uses TinyBERT as the text encoder and mobileViTv2 as the image encoder that builds an offline index using 100,000 candidate images. We successfully achieve ∼24.6ms/image scan speed, ∼8.0ms/query real-time retrieval speed, and ∼100MB running memory. Thanks to the deep optimization on the chip side, the retrieval speed even surpasses that on the V100 GPU, greatly advancing the mobile-device application of VLP models.

5 Conclusion and Liminations

In this study, we propose a multi-teacher cross-modal alignment distillation (MCAD) framework which helps better integrate heterogeneous teachers. The proposed MCAD involves the integration of the teachers' output features and similarity distributions. Moreover, MCAD uses the integrated distributions to distill the student model and align the student's features to the fused teachers' features. Our proposed MCAD is demonstrated to be a model-agnostic general framework, capable of achieving superior performance on both zero-shot and fine-tuning settings and a lightweight model has been successfully deployed on mobile devices, achieving real-time retrieval speed.

In this research, due to computational resource constraints, we only conduct a few experiments and simply determine the hyper-parameter $k = 11$ for all experiments. But it may be dynamic for different datesets and networks. Another limitation is that using MLP as the projection layers in Eq. (12) my not be optimal and more intricate designs need to be investigated in the future.

References

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. {UNITER}: Learning {un}iversal image-{te}xt representations.

Haiwen Diao, Ying Zhang, Lin Ma, and Huchuan Lu. 2021. Similarity reasoning and filtration for image-text matching. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Virtual Event, February 2-9, 2021*, pages 1218–1226. AAAI Press.

Zhiyuan Fang, Jianfeng Wang, Xiaowei Hu, Lijuan Wang, Yezhou Yang, and Zicheng Liu. 2021a. Compressing visual-linguistic model via knowledge distillation. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 1408–1418. IEEE.

Zhiyuan Fang, Jianfeng Wang, Xiaowei Hu, Lijuan Wang, Yezhou Yang, and Zicheng Liu. 2021b. Compressing visual-linguistic model via knowledge distillation. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 1408–1418. IEEE.

Ross B. Girshick. 2015. Fast R-CNN. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 1440–1448. IEEE Computer Society.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1789–1819.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *ArXiv preprint*, abs/1503.02531.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 4904–4916. PMLR.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. TinyBERT: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174, Online. Association for Computational Linguistics.

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.

Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 5583–5594. PMLR.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Jie Lei, Xinlei Chen, Ning Zhang, Mengjiao Wang, Mohit Bansal, Tamara L Berg, and Licheng Yu. 2022. Loopitr: Combining dual and cross encoder architectures for image-text retrieval. *ArXiv preprint*, abs/2203.05465.

Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Gotmare, Shafiq R. Joty, Caiming Xiong, and Steven Chu-Hong Hoi. 2021. Align before fuse: Vision and language representation learning with momentum distillation. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 9694–9705.

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: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer.

Chunxiao Liu, Zhendong Mao, An-An Liu, Tianzhu Zhang, Bin Wang, and Yongdong Zhang. 2019. Focus your attention: A bidirectional focal attention network for image-text matching. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, page 3–11, New York, NY, USA. Association for Computing Machinery.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

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 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13–23.

Sachin Mehta and Mohammad Rastegari. 2022. Separable self-attention for mobile vision transformers. *Transactions on Machine Learning Research*.

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 *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 9826–9836. Computer Vision Foundation / IEEE.

Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. 2011. Im2text: Describing images using 1 million captioned photographs. In *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*, pages 1143–1151.

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.

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, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Jun Rao, Liang Ding, Shuhan Qi, Meng Fang, Yang Liu, Li Shen, and Dacheng Tao. 2023. Dynamic contrastive distillation for image-text retrieval. *IEEE Transactions on Multimedia*.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. 2015. Faster R-CNN: towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 91–99.

Siyu Ren and Kenny Zhu. 2022. Leaner and faster: Two-stage model compression for lightweight text-image retrieval. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4085–4090, Seattle, United States. Association for Computational Linguistics.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *ArXiv preprint*, abs/2210.08402.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. 2021. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *ArXiv preprint*, abs/2111.02114.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, Melbourne, Australia. Association for Computational Linguistics.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. 2021. Training data-efficient image transformers & distillation through attention. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 10347–10357. PMLR.

Pavan Kumar Anasosalu Vasu, Hadi Pouransari, Fartash Faghri, Raviteja Vemulapalli, and Oncel Tuzel. 2023. Mobileclip: Fast image-text models through multi-modal reinforced training. *ArXiv preprint*, abs/2311.17049.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Shakti N Wadekar and Abhishek Chaurasia. 2022. Mobilevitv3: Mobile-friendly vision transformer with simple and effective fusion of local, global and input features. *ArXiv preprint*, abs/2209.15159.

Tiannan Wang, Wangchunshu Zhou, Yan Zeng, and Xinsong Zhang. 2022a. Efficientvlm: Fast and accurate vision-language models via knowledge distillation and modal-adaptive pruning. *ArXiv preprint*, abs/2210.07795.

Zekun Wang, Wenhui Wang, Haichao Zhu, Ming Liu, Bing Qin, and Furu Wei. 2022b. Distilled dual-encoder model for vision-language understanding. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8901–8913, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. 2022c. Simvlm: Simple visual language model pretraining with weak supervision. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Kan Wu, Houwen Peng, Zhenghong Zhou, Bin Xiao, Mengchen Liu, Lu Yuan, Hong Xuan, Michael Valenzuela, Xi Stephen Chen, Xinggang Wang, et al. 2023.

Tinyclip: Clip distillation via affinity mimicking and weight inheritance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21970–21980.

I. Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. 2019. Billion-scale semi-supervised learning for image classification. *ArXiv preprint*, abs/1905.00546.

Ze Yang, Linjun Shou, Ming Gong, Wutao Lin, and Daxin Jiang. 2020. Model compression with two-stage multi-teacher knowledge distillation for web question answering system. In *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020*, pages 690–698. ACM.

Chen Zhang, Yang Yang, Qifan Wang, Jiahao Liu, Jingang Wang, Wei Wu, and Dawei Song. 2023. Autodisc: Automatic distillation schedule for large language model compression.

Shiji Zhao, Jie Yu, Zhenlong Sun, Bo Zhang, and Xingxing Wei. 2022. Enhanced accuracy and robustness via multi-teacher adversarial distillation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*, pages 585–602. Springer.

Table 6: statistic of the dataset. ┬ represents the datasets only used in the fine-tuning stage.

Datasets	CC	SUB	COCO	Flickr
train	1.90M	0.85 M	0.56M┬	0.15M┬
val	-	-	25k	5k
test	-	-	25k	5k

A Dataset Statistics

Some of the images are no longer accessible on the internet, and the CC dataset we collect for training is not quite complete. Table 6 shows the statistics of the datasets. ┬ in Table 6 denotes the data we only used for the fine-tuning stage.

B Train Details

In this study, the AdamW (Loshchilov and Hutter, 2019) optimization technique with $lr = 1e-3, \beta_1 = 0.9, \beta_2 = 0.999$ is employed for all experiments, except for the test on ALBERT+ MobileVitv3 backbone, where the default learning rate is adjusted to 1e-4. To facilitate the training process and enhance the performance, warm-up with cosine decay is applied, while the apex framework is utilized to accelerate the training. Notably, no data augmentation methods are utilized in the teacher models, while the student model only employs "RandomResizedCrop". Moreover, to ensure sufficient training, each experiment is trained for 100 epochs. It is important to mention that the value of the hyperparameter k is set to 11, unless otherwise specified in this paper.

In terms of teacher models, we adopt CLIP ViT-L/14 as the dual-stream teacher[1] and ALBEF[2] as the single-stream teacher. For the projection layers $\{g_1, g_2, g_3, g_4\}$, we simply employ two-layer MLPs.

C Loss Explanation

We choose a special form of normalization term in Eq. (11). We can view it as an L1 normalization. Here we want to explain why we choose such normalization formulation instead of commonly used softmax. Given that single-stream models similar to ALBEF typically output a score for an image-text pair, to ensure that the scores of different sample pairs maintain their relative magnitude after

[1] https://huggingface.co/openai/clip-vit-large-patch14

[2] https://github.com/salesforce/ALBEF

normalization, we employed this specific normalization approach. Take an example for clearer clarification. Assume that the top 3 output scores are 0.8, 0.4, 0.2 (the probability of two-classification after ALBEF must be between 0 and 1), after the L1 normalization, the outputs are $\{\frac{0.8}{0.8+0.4+0.2}\}$ = $\{0.571, 0.286, 0.143\}$. The relative ratio is still $4 : 2 : 1$. But if we choose softmax normalization, the output becomes $\{0.451, 0.302, 0.247\}$, which is much smoother and lacking in differentiation. Actually, we have indeed tried to apply softmax normalization during our experiments, but we found that simply using a softmax would cause ALBEF's score distribution to become smoother and result in inferior performance, while incorporating a temperature-scaled softmax function would introduce additional hyper-parameters. So we finally chose the L1 normalization method.