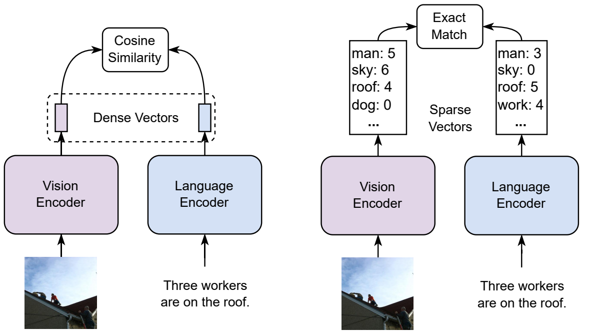
LexLIP: Lexicon-Bottlenecked Language-Image Pre-Training for Large-Scale Image-Text Sparse Retrieval

Ziyang Luo[1]*, Pu Zhao[2], Can Xu[2], Xiubo Geng[2], Tao Shen[2], Chongyang Tao[2],
Jing Ma[1]†, Qingwei Lin[2], Daxin Jiang[2]†
[1] Hong Kong Baptist University, Hong Kong SAR, China
[2] Microsoft Corporation
cszyluo@comp.hkbu.edu.hk, majing@hkbu.edu.hk
{pu.zhao,caxu,xigeng,shentao,chongyang.tao,qlin,djiang}@microsoft.com

Abstract

Image-text retrieval (ITR) aims to retrieve images or texts that match a query originating from the other modality. The conventional ***dense retrieval paradigm*** *relies on encoding images and texts into dense representations with dual-stream encoders. However, this approach is limited by slow retrieval speeds in large-scale scenarios. To address this issue, we propose a novel* ***sparse retrieval paradigm*** *for ITR that exploits sparse representations in the vocabulary space for images and texts. This paradigm enables us to leverage bag-of-words models and efficient inverted indexes, significantly reducing retrieval latency. A critical gap emerges from representing continuous image data in a sparse vocabulary space. To bridge this gap, we introduce a novel pre-training framework,* ***Lexicon-Bottlenecked Language-Image Pre-Training (LexLIP)****, that learns importance-aware lexicon representations. By using lexicon-bottlenecked modules between the dual-stream encoders and weakened text decoders, we are able to construct continuous bag-of-words bottlenecks and learn lexicon-importance distributions. Upon pre-training with same-scale data, our* ***LexLIP*** *achieves state-of-the-art performance on two ITR benchmarks, MSCOCO and Flickr30k. Furthermore, in large-scale retrieval scenarios,* ***LexLIP*** *outperforms CLIP with* $5.8\times$ *faster retrieval speed and* $19.1\times$ *less index storage memory. Beyond this,* ***LexLIP*** *surpasses CLIP across 8 out of 10 zero-shot image classification tasks.*

1. Introduction



(a) Dense Retrieval Paradigm. (b) Sparse Retrieval Paradigm.

Figure 1: Comparing the traditional dense retrieval paradigm and our brand-new sparse retrieval paradigm.

Image-text retrieval (ITR) is a critical problem that involves retrieving relevant images and texts based on textual and visual queries. Its practical applications span multiple domains, including e-commerce product search [27] and social media image search [19]. Due to the lack of publicly available large-scale benchmarks, the evaluation of existing ITR models [20, 32, 39, 44] is commonly conducted on small-scale datasets such as MSCOCO [29] and Flickr30k [38], which contain a limited number of samples in their test sets. Thus, the significance of retrieval speed is frequently disregarded. However, real-world scenarios, such as Google Image Search, may involve a massive number of candidate samples, easily exceeding 1M. Hence, retrieval speed is a crucial concern. The current dense retrieval paradigm, in which each image and text is represented as a dense vector (as shown in Figure 1a), can become computationally expensive, resulting in slow retrieval speed. The large-scale exact k-nearest neighbor (KNN) dense retrieval involves calculating the similarity between the query and all candidate samples, resulting in a linear increase in retrieval time as the number of samples increases [5]. This limitation poses a challenge for the dense retrieval paradigm in real-world applications and underscores the need for more efficient and

*Work done during the internship at Microsoft.
†Corresponding author

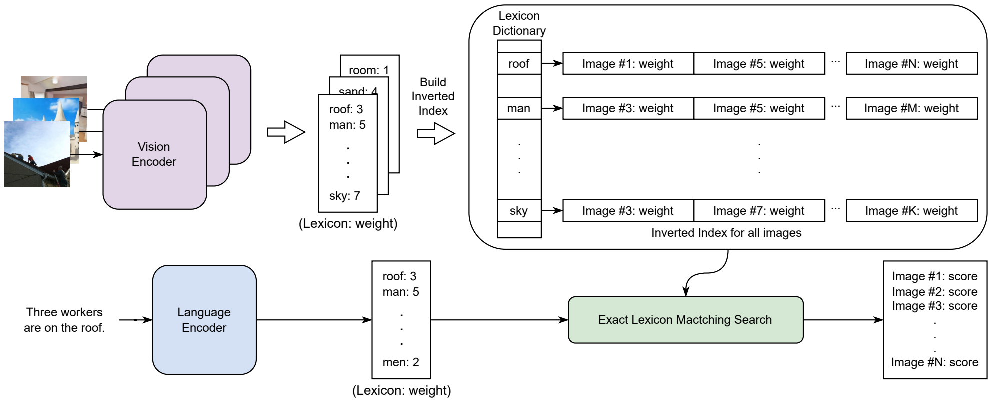


Figure 2: An overview of the retrieval process in our proposed sparse retrieval paradigm for text-to-image retrieval.

effective methods for large-scale retrieval.

In this work, we present a novel **sparse retrieval paradigm** for ITR, as illustrated in Figure 1b. This paradigm encodes images and texts as sparse representations in the vocabulary space, wherein the relevant lexicons are assigned high weights, and the others are set to zero. The retrieval process involves transforming these lexicon-weighted representations into inverted indexes, as depicted in Figure 2. Then, we apply the Exact Lexicon Matching Search algorithm [40] to find matching pairs, which only calculates similarity scores with candidates that share common lexicons. This mechanism avoids iterating over all samples and substantially reduces retrieval latency. Moreover, this paradigm leverages lexicon-level contextualization by considering both implicit object expansion [15] and explicit object concurrence [36].

The sparse retrieval paradigm poses a significant challenge for image processing because images are continuous data that need to be projected into a discrete vocabulary space. To bridge this gap, we propose a novel pre-training framework, termed **Lexicon-Bottlenecked Language Image Pre-Training (LexLIP)**, to learn importance-aware lexicon representations. **LexLIP** comprises two pre-training phases: i) lexicon-bottlenecked pre-training and ii) momentum lexicon-contrastive pre-training.

In the first pre-training phase, we introduce two novel objectives: image/text lexicon-bottlenecked masked language modeling. These objectives aim to establish the lexicon-weighting representations as the bottlenecks between the images/texts data and the sparse vocabulary space. Specifically, we pass an image or masked text into a vision or language encoder to derive the lexicon-weighting representations. Meanwhile, we utilize a weakened masking-style text decoder to reconstruct the masked text from these representations. Due to the aggressive masking, the decoder is inclined to recover masked tokens based on the lexicon-weighting representations. As a result, the LexLIP encoders assign higher importance scores to crucial vocabulary lexicons of the image or text and lower scores to trivial ones. This aligns well with the goal of the sparse retrieval paradigm and enhances its performance.

The second pre-training phase is momentum lexicon-contrastive learning, where images and texts are further aligned in the sparse vocabulary space with a large-scale negative sample size. The experimental results reveal that our **LexLIP** pre-trained with same-scale image-text pairs achieves state-of-the-art (SOTA) performance on the widely used ITR benchmarks, MSCOCO [29] and Flickr30k [38]. In the large-scale retrieval scenario (i.e., the candidate pool has one million samples), **LexLIP** demonstrates a remarkable improvement in retrieval speed with a 5.8 times faster and a significant reduction in index storage memory with a 19.1 times decrease, compared to CLIP [39]. Beyond this, **LexLIP** surpasses CLIP across 8 out of 10 zero-shot image classification tasks. Our codes are available at `https://github.com/ChiYeungLaw/LexLIP-ICCV23`.

Our contributions can be listed as follows:

1. We introduce the novel **Sparse Retrieval Paradigm** to ITR. By representing images and texts in the lexicon vocabulary space, this approach significantly improves the efficiency of large-scale ITR.
2. We propose a new framework, **Lexicon-Bottlenecked Language Image Pre-Training (LexLIP)**, to learn the lexicon-weighting representations.
3. We conduct extensive experiments on both small-scale and large-scale ITR benchmarks. Pre-trained with same-scale data, our **LexLIP** achieves SOTA performance on small-scale ITR datasets, MSCOCO and Flickr30k. Moreover, our **LexLIP** achieves 5.8× speed-up and 19.1× less index storage memory than CLIP on large-scale ITR. Beyond this, our **LexLIP** outperforms CLIP across 8 out of 10 zero-shot image classification tasks.

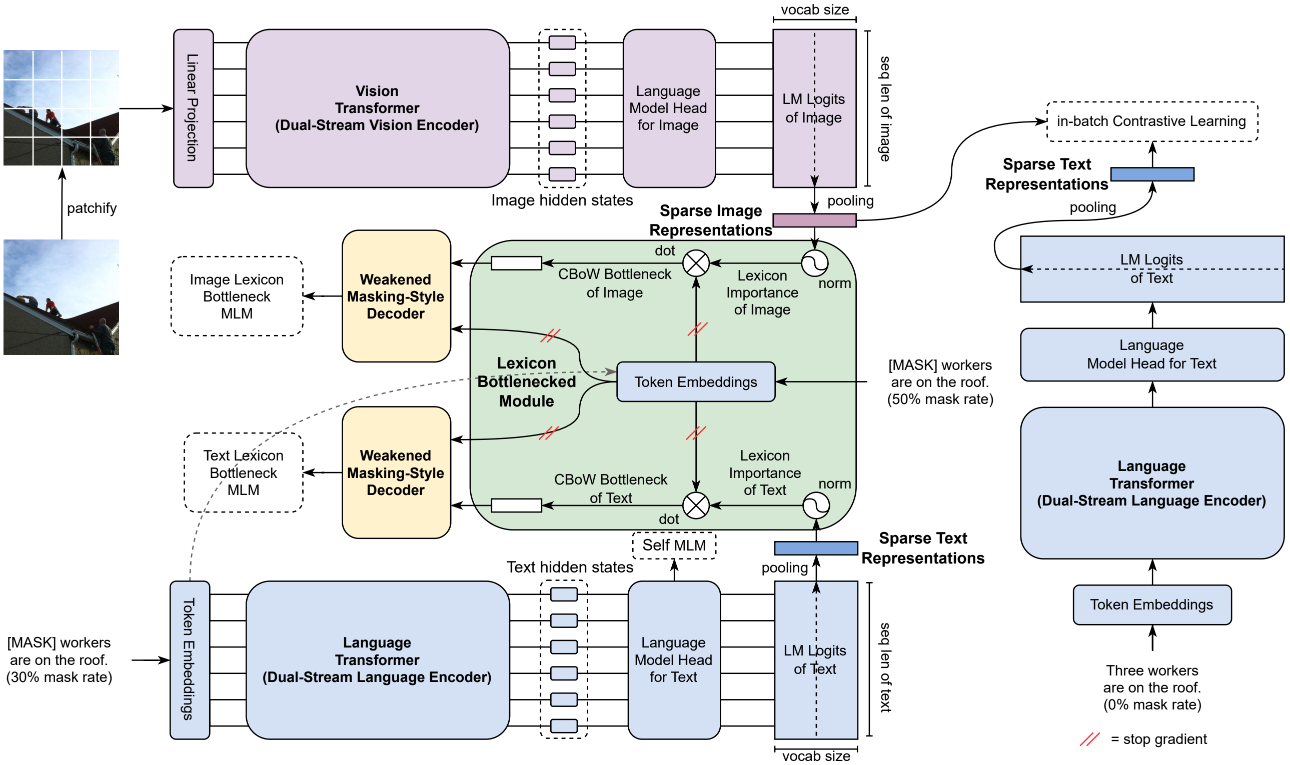


Figure 3: An overview of the Lexicon-Bottlenecked Pre-training phase, including self-supervised masked language modeling, image/text lexicon-bottlenecked masked language modeling, and in-batch lexicon-contrastive learning.

2. Related Work

Image-Text Retrieval. ITR has received considerable attention in the cross-modal community. Traditional approaches to ITR utilized Convolutional Neural Networks (CNNs) [3] as encoders to individually encode images and texts [12, 45]. In recent years, the popularity of transformer-based models and large-scale language-image pre-training have seen a surge [20, 26, 28, 32, 33, 34, 39, 44, 51, 52, 53]. These models introduce several methods to enhance ITR performance, including knowledge distillation, data augmentation, multitask learning, and gigantic-scale pre-training. They have achieved remarkable performance on various ITR benchmarks. However, these models rely on the traditional dense retrieval paradigm, which faces the challenge of slow retrieval speed in large-scale scenarios.

Sparse Retrieval Paradigm. This paradigm originates from the BM25 [40] for exact lexicon matching, which utilizes an inverted index to reduce retrieval latency by only considering samples with overlapping lexicons during the retrieval process. This method has recently gained popularity in NLP document retrieval [14, 15, 24, 42, 43, 55], due to its efficiency in handling large-scale text data and the ability to integrate well with neural network-based models. However, while the text data is naturally discrete and can be projected into a vocabulary space, images are continuous and pose a challenge for sparse lexicon representation.

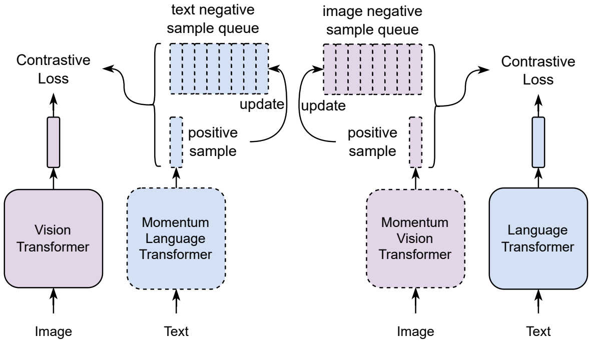


Figure 4: An overview of the Momentum Lexicon-Contrastive Pre-training phase.

Bottlenecked Pre-training in Retrieval. This method is widely studied in the document retrieval [16, 17, 30, 42, 46]. The masked language modeling objective is conditioned on dense representations. Despite its proven success in NLP, this method has not yet been widely explored in ITR. In this work, we aim to fill this gap by proposing a novel pre-training method, which leverages sparse lexicon representations as bottlenecks to enhance the performance.

3. LexLIP

Overview. Adhering to the recent trends in ITR [20, 32, 39, 44], our **LexLIP** framework employs the dual-stream encoding structure. Both images and texts are embedded into distinct sparse lexicon representations. To achieve this, we introduce two pre-training phases in our framework, namely: (i) Lexicon-Bottlenecked Pre-training (as shown in Figure 3), and (ii) Momentum Lexicon-Contrastive Pre-training (as shown in Figure 4). In the following sections, we will delve into the encoding, pre-training, and inference details of our framework.

3.1. Dual-Stream Encoders and Sparse Lexicon Representations

Following recent works [32, 39], the backbone of the visual encoder is the Vision Transformer [11], while the Language Transformer [10] serves as the backbone of the textual encoder. The input image is first transformed into a series of flattened 2D patches, which are subsequently processed by the visual encoder to generate the corresponding hidden states. Formally, given all patches of an image $x = [x_1, \ldots, x_m]$, the visual encoder transform them into fixed-length vectors:

$$H^v = \text{Trans}^v\left([\text{CLS}^v; x]\right) \in \mathbb{R}^{(m+1)\times d}, \tag{1}$$

where Trans^v is the visual encoder and d is the model size. In the dense retrieval paradigm, it is a common practice to utilize the first hidden state of H^v as the dense representations of images [32, 33, 39]. Differently, our visual encoder is followed by a language model head which projects the hidden states into the sparse vocabulary space:

$$\boldsymbol{S}_x^{(\text{enc})^v} = \text{LM-Head}^v(H^v) \in \mathbb{R}^{(m+1)\times|\mathbb{V}|}, \tag{2}$$

where $|\mathbb{V}|$ is the vocabulary size. We denote $\boldsymbol{S}_x^{(\text{enc})^v}$ as the LM logits of images from visual encoder. Then, we follow the SPLADE model in document retrieval [15] to represent an image in the high-dimensional vocabulary space by

$$p^v = \log(1 + \text{MaxPool}(\max(\boldsymbol{S}_x^{(\text{enc})^v}, 0))) \in \mathbb{R}^{|\mathbb{V}|}, \tag{3}$$

where $\max(\cdot, 0)$ ensures all values greater than or equal to zero for the sparse requirements, $\text{MaxPool}(\cdot)$ denotes max pooling along with the sequence axis, and the saturation function $\log(1 + \text{MaxPool}(\cdot))$ prevents some terms from dominating. p^v stands for the lexicon-weighting sparse representation of an image.

Similarly, the language encoder generates the lexicon-weighting sparse representation of the input text $y = [y_1, ..., y_n]$ by

$$H^l = \text{Trans}^l\left([\text{CLS}^l; y]\right) \in \mathbb{R}^{(n+1)\times d}, \tag{4}$$

$$\boldsymbol{S}_y^{(\text{enc})^l} = \text{LM-Head}^l(H^l) \in \mathbb{R}^{(n+1)\times|\mathbb{V}|}, \tag{5}$$

$$p^l = \log(1 + \text{MaxPool}(\max(\boldsymbol{S}_y^{(\text{enc})^l}, 0))) \in \mathbb{R}^{|\mathbb{V}|}, \tag{6}$$

where Trans^l is the language encoder, $\boldsymbol{S}_y^{(\text{enc})^l}$ is the LM logits of texts from language encoder, LM-Head^l is the language model head for texts, and p^l is the lexicon-weighting sparse representation of a text.

3.2. Phase 1: Lexicon-Bottlenecked Pre-training

As shown in Figure 3, this pre-training phase consists of four different objectives, including self-supervised masked language modeling, two lexicon-bottlenecked masked language modelings and in-batch lexicon-contrastive learning.

Self-Supervised Masked Language Modeling (Self-MLM). Consistent with the standard practice of pre-training the language encoder in an unsupervised manner, the masked language modeling (MLM) objective is utilized for pre-training our language encoder, Trans^l. Formally, the tokens in the input text y are masked to obtain $\bar{y}$, with $\alpha\%$ tokens being replaced by a special token [MASK] or a random token in the vocabulary set, $\mathbb{V}$, and the remaining being kept unchanged. The masked $\bar{y}$ is then processed by the language encoder to generate the language model (LM) logits, $\boldsymbol{S}_{\bar{y}}^{(\text{enc})^l}$, and reconstruct the masked tokens through the following objective function:

$$\mathcal{L}_{\text{self}} = -\sum_{\mathbb{D}} \sum_{j\in\mathbb{M}^{(\text{enc})}} \log P(\text{w}^j = y_j|\bar{y}), \tag{7}$$

where $P(\text{w}^j)$ is calculated as $softmax\left(\boldsymbol{S}_{\bar{y}}^{(\text{enc})^l}[j,:]\right)$, $\mathbb{D}$ represents the set of all samples, $\mathbb{M}^{(\text{enc})}$ denotes the set of masked indices in $\bar{y}$, w^j represents the discrete variable over $\mathbb{V}$ at the j-th position of y, and y_j refers to its original token.

Lexicon-Bottlenecked Masked Language Modelings (LexMLM). Regarding to the token-level logits from Eq. 2 and 5 defined in the lexicon vocabulary space, we propose to calculate the lexicon-importance distributions of images and masked texts by

$$a^v = \text{Normalize}\left(\text{MaxPool}(\boldsymbol{S}_x^{(\text{enc})^v})\right) \in [0,1]^{|\mathbb{V}|}, \tag{8}$$

$$a^l = \text{Normalize}\left(\text{MaxPool}(\boldsymbol{S}_{\bar{y}}^{(\text{enc})^l})\right) \in [0,1]^{|\mathbb{V}|}, \tag{9}$$

where $\text{Normalize}(\cdot) = softmax(\cdot)$ denotes the normalization function (let $\sum a_i = 1$). $a^{(\cdot)}$ denotes lexicon-importance distribution over $\mathbb{V}$ to indicate the relative importance of the different lexicons in the vocabulary.

To obtain the lexicon-importance distributions, we are inspired by the bottleneck-enhanced dense representation learning strategy from recent works in document retrieval [16, 17, 30, 42]. Our framework introduces a **lexicon-bottlenecked module** to utilize these distributions as a bridge to guide the reconstruction of masked lexicons, leading the vision and language encoders to focus on the most critical tokens/words in the data. However, directly utilizing the high-dimensional distribution vectors $a^{(\cdot)} \in [0,1]^{|\mathbb{V}|}$ as the bottlenecks faces challenges. First, the distribution over the entire vocabulary space possesses a capacity to encapsulate the majority of data semantics [50]. Consequently, the efficacy of the bottleneck is reduced. Second, it is difficult to input the high-dimensional vector into a decoder for text reconstruction.

Therefore, we present a novel approach in which we generate continuous bag-of-words (CBoW) representations as the bottlenecks, guided by the lexicon-importance distributions acquired from Equations 8 and 9. That is

$$b^{(\cdot)} = a^{(\cdot)} sg\left(W^{(\text{te})}\right) \in \mathbb{R}^{d}, \tag{10}$$

where $W^{(\text{te})} \in R^{|\mathbb{V}| \times d}$ is the token embeddings matrix of the language encoder and $sg(\cdot)$ refers to stop gradient. Thereby, $b^{(\cdot)}$ stands for **CBoW bottleneck representations**.

To guide the learning of the bottleneck representations $b^{(\cdot)}$, which in turn leads to the learning of the lexicon-importance distributions $a^{(\cdot)}$, we use two decoders, one for vision and one for language, to reconstruct the masked text $\bar{y}$ from $b^{(\cdot)}$. This approach follows recent advancements in bottleneck-enhanced neural structures [16, 17, 42]. The two decoders, which we refer to as **weakened masking-style decoders**, are designed to place a heavy reliance on the bottleneck representations by employing two strategies: (i) an aggressive masking strategy, and (ii) using only two shallow transformer layers.

In particular, given the masked text $\bar{y}$, we adopt an aggressive masking strategy to produce the masked text $\hat{y}$ with a larger masking rate. This prompts the encoders to compress the rich contextual information into the bottleneck representation, $b^{(\cdot)}$. Subsequently, the bottleneck representation prefixes $\hat{y}$ by replacing the [CLS] special token. Therefore, our weakened masking-style decoding can be formulated as

$$\boldsymbol{S}_{\hat{y}}^{(\text{dec})^{v}} = \text{Decoder}^{v}\left([b^{v}; \hat{y}]\right) \in \mathbb{R}^{(n+1)\times|\mathbb{V}|}, \tag{11}$$

$$\boldsymbol{S}_{\hat{y}}^{(\text{dec})^{l}} = \text{Decoder}^{l}\left([b^{l}; \hat{y}]\right) \in \mathbb{R}^{(n+1)\times|\mathbb{V}|}. \tag{12}$$

Similar to the Self-MLM, the loss functions are:

$$\mathcal{L}_{i2t} = -\sum_{\mathbb{D}} \sum_{j \in \mathbb{M}^{(\text{dec})}} \log P^{v}(\text{w}^{j} = y_j | \hat{y}), \tag{13}$$

$$\mathcal{L}_{t2i} = -\sum_{\mathbb{D}} \sum_{j \in \mathbb{M}^{(\text{dec})}} \log P^{l}(\text{w}^{j} = y_j | \hat{y}), \tag{14}$$

where $P^{(\cdot)}(\text{w}^{j}) = softmax\left(\boldsymbol{S}_{\hat{y}}^{(\text{dec})^{(\cdot)}}[j,:]\right)$, and $\mathbb{M}^{(\text{dec})}$ denotes the set of masked tokens in $\hat{y}$.

In-Batch Lexicon-Contrastive Learning (BaCo). Given the lexicon sparse representations from Eqs. 3 and 6, we perform in-batch contrastive learning in this phase to align images and texts in the vocabulary space. The models learn by contrasting the lexicon-weighting sparse representations of different samples within a single batch of data. The loss functions are:

$$\mathcal{L}_{baco}^{i2t} = -\sum_{\mathbb{D}} \log \frac{\exp\left(p^{v}(p^{l})^{T}\right)/\tau}{\sum_{j \in \mathcal{B}} \exp\left(p^{v}(p^{l}_{j})^{T}\right)/\tau} + \lambda \mathcal{F}(p^{v}), \tag{15}$$

$$\mathcal{L}_{baco}^{t2i} = -\sum_{\mathbb{D}} \log \frac{\exp\left(p^{v}(p^{l})^{T}\right)/\tau}{\sum_{j \in \mathcal{B}} \exp\left(p^{v}_{j}(p^{l})^{T}\right)/\tau} + \lambda \mathcal{F}(p^{l}), \tag{16}$$

where $\mathcal{B}$ denotes all the data in a batch, τ is the temperature hyperparameter, $\mathcal{F}(\cdot)$ is the FLOPS function introduced in SPLADE [15] for representation sparsity, and λ is the regularization hyperparameter. The overall loss function is:

$$\mathcal{L}_{baco} = \left(\mathcal{L}_{baco}^{i2t} + \mathcal{L}_{baco}^{t2i}\right)/2. \tag{17}$$

Phase 1 Learning. The final loss function of the Lexicon-Bottlenecked Pre-training is a direct addition of all losses:

$$\mathcal{L}_{p1} = \mathcal{L}_{self} + \mathcal{L}_{i2t} + \mathcal{L}_{t2i} + \mathcal{L}_{baco}. \tag{18}$$

3.3. Phase 2: Momentum Lexicon-Contrastive Pre-training.

After learning the sparse lexicon representations in the first phase, we further align the representations of images and texts in the vocabulary space. It has been shown that the large-scale negative samples is crucial for achieving good performance in ITR [39]. However, the negative sample size is limited by the mini-batch size in traditional in-batch contrastive learning, which can be constrained by the GPU's memory. To address this issue, we adopt the momentum contrastive learning in MoCo [18] to cache negative samples with two different queues, Q^{v} and Q^{l}, for images and texts, respectively. This approach decouples the negative sample size from the mini-batch size, making the learning process more computationally feasible.

In accordance with prior works [32, 33], two momentum encoders, θ_{m}^{v} and θ_{m}^{l}, are employed to update the samples in the queues. These encoders share the same structures and initial parameters as the original encoders, but they have truncated gradients and are updated utilizing the exponential moving average (EMA) mechanism:

$$\theta_{m}^{v} = m\theta_{m}^{v} + (1-m)\theta_{o}^{v}, \tag{19}$$

Model	∗#I-T	Flickr30k Test (1K Images)						MSCOCO Test (5K Images)					
		T2I Retrieval			I2T Retrieval			T2I Retrieval			I2T Retrieval		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Dense-vector Dual-Stream Retriever													
Frozen (ICCV21 [1])	5.5M	61.0	87.5	92.7	-	-	-	-	-	-	-	-	-
LD (NAACL21 [44])	9.5M	69.9	91.1	95.2	83.9	97.2	98.6	45.8	74.6	83.8	60.1	85.1	91.8
COOKIE (ICCV21 [47])	5.9M	68.3	91.1	95.2	84.7	96.9	98.3	46.6	75.2	84.1	61.7	86.7	92.3
ViSTA (CVPR22 [7])	9.5M	68.9	91.1	95.1	84.8	97.4	99.0	47.8	75.8	84.5	63.9	87.8	93.6
⋄**CLIP** (ICML21 [39])	4.3M	70.6	90.4	94.4	87.3	97.6	98.7	47.2	75.0	83.9	62.2	86.5	93.2
†**Dense** (ours)	4.3M	74.0	92.8	95.6	88.0	98.1	99.5	49.5	76.7	85.4	65.5	88.6	94.3
‡**COTS** (CVPR22 [32])	5.3M	75.2	93.6	96.5	88.2	98.5	99.7	50.5	77.6	86.1	66.9	88.8	94.0
†**Dense** (ours)	14.3M	75.6	93.6	96.6	90.2	98.7	99.7	51.7	77.9	86.1	68.5	89.8	94.5
‡**COTS** (CVPR22 [32])	15.3M	76.5	93.9	96.6	90.6	98.7	99.7	52.4	79.0	**86.9**	69.0	90.4	94.9
Sparse-vector Dual-Stream Retriever													
LexLIP (ours)	4.3M	76.7	93.7	96.8	89.6	98.7	99.6	51.9	78.3	86.3	67.9	89.7	94.8
LexLIP (ours)	14.3M	**78.4**	**94.6**	**97.1**	**91.4**	**99.2**	**99.7**	**53.2**	**79.1**	86.7	**70.2**	**90.7**	**95.2**

∗ #I-T corresponds to the number of image-text pairs during pre-training.

⋄ Re-implement the CLIP model with the same-scale pre-training as our models.

† Represent data with the dense CLS representations and conduct the similar pre-training process as LexLIP.

‡ The state-of-the-art dense-vector dual-stream retriever, pre-trained with the same-scale image-text data.

Table 1: Evaluation our **LexLIP** in the small-scale retrieval scenario after fine-tuning.

$$\theta_m^l = m\theta_m^l + (1-m)\theta_o^l, \tag{20}$$

where θ_o is the parameters of the original encoders and m is the EMA decay weight. The momentum lexicon sparse representations of images and texts are denoted as $\hat{p}^v$ and $\hat{p}^l$. These momentum encoders are dropped after pre-training. The momentum contrastive loss functions are:

$$\mathcal{L}_{moco}^{i2t} = -\sum_{\mathbb{D}} \log \frac{\exp\left(p^v(\hat{p}^l)^T\right)/\tau}{\sum_{q_j^l \in Q^l \cup \{\hat{p}^l\}} \exp\left(p^v(q_j^l)^T\right)/\tau}, \tag{21}$$
$$+ \lambda\mathcal{F}(p^v),$$

$$\mathcal{L}_{moco}^{t2i} = -\sum_{\mathbb{D}} \log \frac{\exp\left(p^l(\hat{p}^v)^T\right)/\tau}{\sum_{q_j^v \in Q^v \cup \{\hat{p}^v\}} \exp\left(p^l(q_j^v)^T\right)/\tau},$$
$$+ \lambda\mathcal{F}(p^l). \tag{22}$$

The overall momentum contrastive loss is:

$$\mathcal{L}_{moco} = (\mathcal{L}_{moco}^{i2t} + \mathcal{L}_{moco}^{t2i})/2. \tag{23}$$

3.4. Exact Lexicon Search for Large-Scale Retrieval

In the inference phase of large-scale retrieval, there exist notable differences between the dense and sparse retrieval methods. As in Eq. 15, 16, 21, and 22, we use the dot-product between the real-valued sparse lexicon-weighted representations to measure the similarity, where the dot-product is necessary for gradient back-propagation and end-to-end learning. However, it is infeasible for open-source term-based sparse retrieval systems like Anserini [49]. To overcome this challenge, we have employed the quantization method to transform the high-dimensional sparse vectors into the corresponding lexicons and their virtual weights. The lexicons are obtained from the non-zero elements of the high-dimensional sparse vector, while the weights are determined through a simple quantization approach, i.e., $\lfloor 100 \times p^{(\cdot)} \rfloor$. Given a query and a candidate sample, the exact lexicon matching score is defined as:

$$score = \sum_{l \in L^q \cap L^s} W_q(l) \times W_s(l), \tag{24}$$

where L^q and L^s denote the lexicon lists of the query and candidate sample, and $W_{(\cdot)}(l)$ is the weight of the lexicon.

Overall, our framework, **LexLIP**, comprises the following large-scale retrieval steps: i) converting all candidate samples into high-dimensional sparse representations, and subsequently into lexicons and frequencies (weights); ii) constructing a term-based inverted index using Anserini [49] for the entire sample collection; iii) generating the lexicons and frequencies for a test query similarly; and iv) querying the inverted index to retrieve the relevant samples.

4. Small-Scale Retrieval Experiment

4.1. Setup

Pre-Training and Evaluation. We use two different image-text datasets to pre-train our **LexLIP**: (1) CC4.3M contains 4.3M image-text pairs from Conceptual Captions 3.3M [41] (about 2.8M urls are valid), SBU [37], MSCOCO [29] training set and Flickr30K [38] training set. (2) CC14.3M consists of CC4.3M and Conceptual Captions 12M [6] (about 10M urls are valid), which contains 14.3M image-text pairs. For the downstream tasks, models are

Model	Index Size↓	Repr Byte↓	QPS↑	Time↓	Large-Scale Flickr30k Test					
					T2I Retrieval			I2T Retrieval		
					R@1	R@5	R@10	R@1	R@5	R@10
Single-Modal Sparse Text Retriever										
BM25 [40]	195M	Avg 195	780.18	1.28ms	16.8	27.3	31.9	34.0	50.7	58.2
Cross-Modal Dense Dual-Stream Retriever										
CLIP [39]	2.9G	Avg 2897	3.42	292.40ms	45.8	64.1	69.5	62.9	85.6	90.8
Dense (ours)	2.9G	Avg 2897	3.55	281.69ms	47.7	66.2	71.7	63.5	86.2	91.9
Our Cross-Modal Sparse Dual-Stream Retriever										
LexLIP (ours)	152M	Avg 152	19.70	50.76ms	48.6	66.8	71.8	64.1	86.5	91.4
-top64 sparsify	114M	Upto192	63.43	15.77ms	47.6	65.7	70.6	64.4	86.7	91.1
-top32 sparsify	71M	Upto 96	249.06	4.02ms	42.0	59.3	64.7	59.4	83.4	87.7
-top16 sparsify	47M	Upto 48	610.08	1.64ms	28.3	44.2	50.6	47.4	70.1	75.9
-top12 sparsify	41M	Upto 36	796.65	1.26ms	22.3	35.9	41.4	40.9	60.4	70.2
-top8 sparsify	34M	Upto 24	985.40	1.01ms	14.6	24.7	29.5	28.6	47.2	55.5

Table 2: Evaluating our **LexLIP** in the large-scale retrieval scenario. "Index Size" corresponds to the storage requirement to embed all candidate images. "Repr Byte" denotes the storage requirement for an embedded image. Each activated (non-zero) term in a lexicon-weighed sparse vector needs 3 bytes (2 bytes for indexing and 1 byte for its weight). "QPS" corresponds to query-per-second (the higher, the faster). "Time" denotes the average time for a query to reach the retrieval result (the lower, the faster). Both "QPS" and "Time" measure the retrieval latency.

Model	Time↓	R@1	R@5	R@10
CLIP (w/o accelerate)	292.40ms	45.8	64.1	69.5
LexLIP (w/o accelerate)	50.76ms	48.6	66.8	71.8
CLIP (ANN accelerate)	4.65ms	25.9	33.7	35.8
LexLIP (top16 sparsify)	1.64ms	28.3	44.2	50.6

Table 3: Accelerate CLIP with Approximate Nearest Neighbor Searching (ANN). The T2I retrieval scores are reported on the large-scale Flickr30k test set. Though the retrieval speed can be increased, the recall decreases a lot.

Tasks	CLIP	LexLIP	Tasks	CLIP	LexLIP
CIFAR10	92.5	**94.0**	DTD	38.9	**42.6**
CIFAR100	70.5	**74.7**	Pets	**61.2**	57.7
Caltech101	**81.9**	81.8	Flowers	41.8	**44.3**
Food101	59.7	**61.0**	MNIST	16.4	**23.2**
SUN397	57.7	**63.2**	ImgNet1K	50.0	**52.6**

Table 4: Zero-shot image classification.

evaluated on MSCOCO and Flickr30k test sets with fine-tuning. Each image in these datasets is accompanied by 5 different captions. We follow the Karpathy split [22] to divide the datasets into train/val/test sets, with 113.2k/5k/5k (MSCOCO) and 29.8k/1k/1k (Flickr30k) images. For evaluation, we use the standard R@k (k=1,5,10) to calculate the retrieval scores of our models, the same as previous works [32, 39, 44, 47].

Implementation Details. For computational efficiency, we follow [32] to initialize the dual-stream encoders with the pre-trained vision [2] and language transformer [10], whereas the other parts are randomly initialized. Both of them are the base-size, 12-layer transformer encoders with 768 hidden size. The pre-trained input image resolution is 224×224. The fine-tuning resolution is 384×384. Models are pre-trained with 20 epochs in the first phase, 10 epochs in the second phase, and fine-tuned with 10 epochs. The AdamW optimization algorithm [31] with a learning rate of 5e-5, linear learning rate decay and 10% warm-up steps, and mixed-precision training are employed. The masking rate for the language encoder is set to 30% and 50% for the decoder, with an EMA weight of 0.99 and a temperature τ of 0.05. The regularization term λ is set to 0.002. Further details can be found in the supplementary material section A.1.

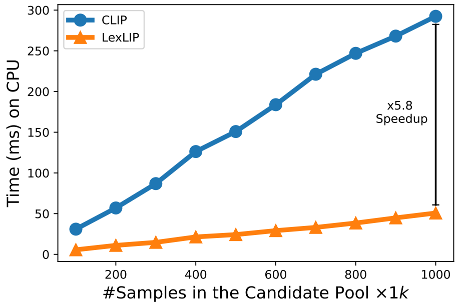


Figure 5: Comparing the retrieval time per query on CPU with different numbers of samples in the candidate pool.

4.2. Results

As shown in Table 1, under a fair comparison (excluding the models pre-trained with billions of image-text pairs), our **LexLIP** achieves the SOTA performance over all previous works for most evaluation metrics. Specifically, in comparison to the previous SOTA COTS [32], **LexLIP** obtains higher results by 1.5% (76.7% vs. 75.2%) for T2I R@1 and 1.4% (89.6% vs. 88.2%) for I2T R@1 on Flickr30k, while utilizing less pre-training data (4.3M vs. 5.3M). Furthermore, with a larger pre-training dataset, **LexLIP** further enhances performance by 1.9% (78.4% vs. 76.5%) for T2I R@1 and 0.8% (91.4% vs. 90.6%) for I2T R@1 on Flickr30k, while still utilizing less data (14.3M vs. 15.3M).

5. Large-Scale Retrieval Experiment

5.1. Setup

Baselines and Large-Scale Benchmark. In addition to our re-implemented CLIP model and Dense model, we have incorporated another baseline, the single-modal sparse text retriever BM25 [40]. By leveraging captions to represent images, BM25 can perform image-text retrieval as caption-text retrieval. For large-scale ITR, we expand the test set of Flickr30k [38] by including 1M randomly selected image-text pairs from Conceptual Caption 12M [6]. Each image in Flickr30k is associated with 5 captions, from which we randomly select one as the alternative image representation for BM25 retrieval. In text-to-image retrieval, models retrieve images from the 1k images of Flickr30k and an additional 1M images, given 4k captions as queries from Flickr30k. Conversely, for image-to-text retrieval, models retrieve captions from the 4k captions of Flickr30k and an additional 1M captions, given 1k images as queries from Flickr30k.

Pre-training and Retrieval. To compare with BM25 in the zero-shot settings, we exclude the Flickr30k training set from the CC4.3M and result in a new dataset CC4.2M for pre-training. For all models, we first embed them into the Index file and then retrieve the results on the CPU. The dense retrieval is conducted with the efficient dense vector similarity search library, Faiss [21]. The sparse retrieval is conducted with the efficient sparse vector similarity search library, Anserini [49]. More details can be found in the supplemental material section A.2.

5.2. Results

Figure 5 compares the retrieval time per query between CLIP and our **LexLIP**, highlighting the inefficiencies of the dense retriever as the number of samples in the candidate pool increases. With a candidate pool of 1M samples, Table 2 shows that the retrieval speed of CLIP is 292.40ms per query. In contrast, our sparse retriever, **LexLIP**, demonstrates a substantial improvement, with a 5.8 times faster retrieval speed (50.76ms vs. 292.40ms) and 19.1 times less storage memory (152M vs. 2.9G). Moreover, our **LexLIP** also exhibits superior performance on the benchmark.

To further analyze our **LexLIP** efficacy-efficiency trade-off, we adopt a simple yet effective sparsification method by retaining only the top-K weighted terms in the representation of the sample during the inference phase and constructing the inverted index using the sparsified samples. As shown in Table 2, our **LexLIP** achieves the best efficacy-efficiency trade-off among all baselines. With top-12 sparsity, **LexLIP** has the 4.8 times smaller index size, faster speed, and better retrieval performance than the sparse text retriever BM25.

Furthermore, we employ the widely utilized technique of Approximate Nearest Neighbor (ANN) Searching to enhance the retrieval efficiency of the dense retriever, CLIP. The results presented in Table 3 demonstrate that while the retrieval speed of CLIP indeed experiences a significant increase, it is accompanied by a notable reduction in recall. Our **LexLIP** with top-16 sparsity has around 2.8 times faster speed, and better retrieval performance than CLIP with ANN acceleration.

6. Zero-shot Image Classification Experiment

6.1. setup

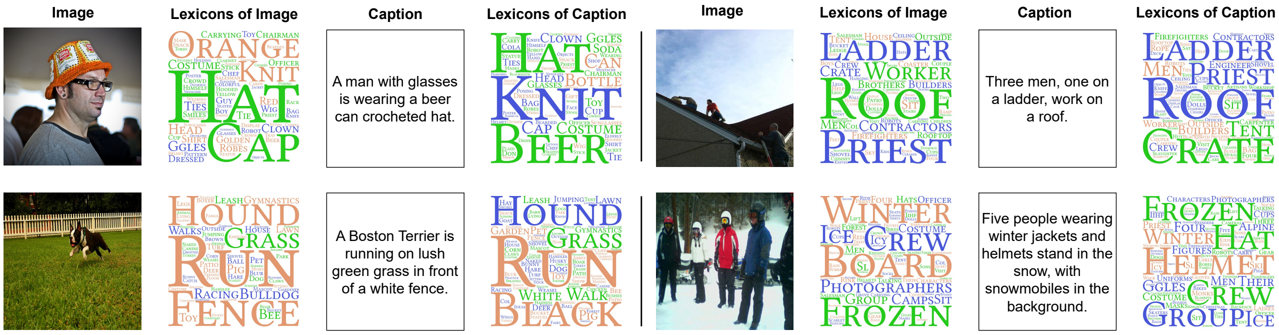
In this study, we undertake a zero-shot image classification experiment utilizing the same models, LexLIP and CLIP, as section 5. The experiment adopts 10 image classification tasks, including CIFAR10, CIFAR100 [23], Caltech101 [13], Food101 [4], SUN397 [48], DTD [8], Pets [54], Flowers [35], MNIST [25], and ImageNet1K [9]. All evaluation settings are the same as the paper of CLIP [39].

6.2. Results

Table 4 provides a comparative analysis of the zero-shot image classification performance between CLIP and our **LexLIP**. It is noteworthy that **LexLIP** exhibits superior performance over CLIP across 8 out of 10 tasks. This noteworthy outcome underscores the efficacy of our model, demonstrating its prowess not solely in retrieval tasks but also in the domain of image classification tasks.

7. Further Analysis

Ablation Study. In Table 5, we present the impact of different pre-training objectives and phases of our **LexLIP**. A pre-training dataset was constructed by randomly sampling 1.4M image-text pairs from Conceptual Captions 3.3M [41] and including all pairs from the Flickr30k [38] training set. The retrieval performance was evaluated on the Flickr30k test set. The results indicate that all pre-training objectives and phases contribute positively to the retrieval performance. The greatest effects were observed for the in-batch contrastive learning in Phase 1 and the momentum contrastive learning in Phase 2, which may be attributed to the alignment of these objectives with the retrieval target. The MLM-based



A man with glasses is wearing a beer can crocheted hat.

Three men, one on a ladder, work on a roof.

A Boston Terrier is running on lush green grass in front of a white fence.

Five people wearing winter jackets and helmets stand in the snow, with snowmobiles in the background.

Figure 6: Visualizing the lexicons cloud of the images and their corresponding captions in the Flickr30k test set.

Phase 1				Phase 2	T2I Retrieval			I2T Retrieval		
$\mathcal{L}_{self}$	$\mathcal{L}_{i2t}$	$\mathcal{L}_{t2t}$	$\mathcal{L}_{baco}$	$\mathcal{L}_{moco}$	R@1	R@5	R@10	R@1	R@5	R@10
✓	✓	✓	✓	✓	**67.5**	**89.7**	**94.1**	81.3	**96.4**	**98.9**
✓	✓	✓	✓		59.5	85.6	91.8	73.4	92.9	96.5
✓	✓	✓		✓	61.0	86.7	92.1	72.7	92.6	96.6
✓	✓		✓	✓	66.5	89.4	93.8	**81.7**	95.0	97.9
✓		✓	✓	✓	66.9	89.6	**94.1**	80.8	95.5	98.7
	✓	✓	✓	✓	67.0	89.3	93.5	78.7	95.2	97.8

Table 5: **LexLIP** ablation experiments on different pre-training objectives.

objectives were also found to be beneficial for learning the lexicon-importance distributions.

Lexicon-Weighting Examples. In Figure 6, we visualize the lexicons of 4 images and their captions. If the lexicon has a high weight, the size is large in the lexicon cloud. We can find that the major features of the images and texts are successfully captured by the lexicons. For example, "hat" is an important lexicon in the first image and caption. More examples are in the supplemental material section B.

8. Conclusion

In this study, we present the novel **Sparse Retrieval Paradigm** for ITR. To overcome the challenge of projecting continuous image data onto the discrete vocabulary space, we introduce the innovative **Lexicon-Bottlenecked Language-Image Pre-training (LexLIP)** framework. Our experiments demonstrate that **LexLIP** outperforms the SOTA models on small-scale retrieval when pre-trained with the same-scale data. Furthermore, in large-scale retrieval, **LexLIP** achieves a substantial improvement in both retrieval speed (5.8× faster) and index storage requirements (19.1× less) compared to the traditional dense retrieval paradigm. Beyond this, **LexLIP** outperforms CLIP across 8 out of 10 zero-shot image classification tasks.

9. Acknowledgments

This work is partially supported by National Natural Science Foundation of China Young Scientists Fund (No. 62206233) and Hong Kong RGC ECS (No. 22200722). We would like to thank the anonymous reviewers for their excellent feedbacks.

References

[1] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 1708–1718. IEEE, 2021. 6

[2] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: BERT pre-training of image transformers. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. 7

[3] Yoshua Bengio, Yann LeCun, and Donnie Henderson. Globally trained handwritten word recognizer using spatial representation, convolutional neural networks, and hidden markov models. In Jack D. Cowan, Gerald Tesauro, and Joshua Alspector, editors, *Advances in Neural Information Processing Systems 6, [7th NIPS Conference, Denver, Colorado, USA, 1993]*, pages 937–944. Morgan Kaufmann, 1993. 3

[4] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 - mining discriminative components with random forests. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI*, volume 8694 of *Lecture Notes in Computer Science*, pages 446–461. Springer, 2014. 8

[5] Leonid Boytsov, David Novak, Yury Malkov, and Eric Nyberg. Off the beaten path: Let's replace term-based retrieval with k-nn search. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, page 1099–1108, New York, NY, USA, 2016. Association for Computing Machinery. 1

[6] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 3558–3568. Computer Vision Foundation / IEEE, 2021. 6, 8

[7] Mengjun Cheng, Yipeng Sun, Longchao Wang, Xiongwei Zhu, Kun Yao, Jie Chen, Guoli Song, Junyu Han, Jingtuo Liu, Errui Ding, and Jingdong Wang. Vista: Vision and scene text aggregation for cross-modal retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 5174–5183. IEEE, 2022. 6

[8] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 3606–3613. IEEE Computer Society, 2014. 8

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 248–255. IEEE Computer Society, 2009. 8

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. 4, 7

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 4

[12] Fartash Faghri, David J. Fleet, Jamie Ryan Kiros, and Sanja Fidler. VSE++: improving visual-semantic embeddings with hard negatives. In *British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018*, page 12. BMVA Press, 2018. 3

[13] Li Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 Conference on Computer Vision and Pattern Recognition Workshop*, pages 178–178, 2004. 8

[14] Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. From distillation to hard negative sampling: Making sparse neural IR models more effective. In Enrique Amigó, Pablo Castells, Julio Gonzalo, Ben Carterette, J. Shane Culpepper, and Gabriella Kazai, editors, *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 2353–2359. ACM, 2022. 3

[15] Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. SPLADE: sparse lexical and expansion model for first stage ranking. In Fernando Diaz, Chirag Shah, Torsten Suel, Pablo Castells, Rosie Jones, and Tetsuya Sakai, editors, *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, pages 2288–2292. ACM, 2021. 2, 3, 4, 5

[16] Luyu Gao and Jamie Callan. Condenser: a pre-training architecture for dense retrieval. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 981–993. Association for Computational Linguistics, 2021. 3, 5

[17] Luyu Gao and Jamie Callan. Unsupervised corpus aware language model pre-training for dense passage retrieval. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2843–2853. Association for Computational Linguistics, 2022. 3, 5

[18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9726–9735. Computer Vision Foundation / IEEE, 2020. 5

[19] Yuting Hu, Liang Zheng, Yi Yang, and Yongfeng Huang. Twitter100k: A real-world dataset for weakly supervised cross-media retrieval. *IEEE Trans. Multim.*, 20(4):927–938, 2018. 1

[20] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In Marina Meila and Tong Zhang, editors, *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, 2021. 1, 3, 4

[21] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Trans. Big Data*, 7(3):535–547, 2021. 8

[22] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 3128–3137. IEEE Computer Society, 2015. 7

[23] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. 8

[24] Carlos Lassance and Stéphane Clinchant. An efficiency study for SPLADE models. In Enrique Amigó, Pablo Castells, Julio Gonzalo, Ben Carterette, J. Shane Culpepper, and Gabriella Kazai, editors, *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 2220–2226. ACM, 2022. 3

[25] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998. 8

[26] Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Gotmare, Shafiq R. Joty, Caiming Xiong, and Steven Chu-Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *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, 2021. 3

[27] Sen Li, Fuyu Lv, Taiwei Jin, Guli Lin, Keping Yang, Xiaoyi Zeng, Xiao-Ming Wu, and Qianli Ma. Embedding-based product retrieval in taobao search. In Feida Zhu, Beng Chin Ooi, and Chunyan Miao, editors, *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 3181–3189. ACM, 2021. 1

[28] Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. 3

[29] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pages 740–755. Springer, 2014. 1, 2, 6

[30] Zheng Liu and Yingxia Shao. Retromae: Pre-training retrieval-oriented transformers via masked auto-encoder. *CoRR*, abs/2205.12035, 2022. 3, 5

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

[32] Haoyu Lu, Nanyi Fei, Yuqi Huo, Yizhao Gao, Zhiwu Lu, and Ji-Rong Wen. COTS: collaborative two-stream vision-language pre-training model for cross-modal retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 15671–15680. IEEE, 2022. 1, 3, 4, 5, 6, 7, 8

[33] Ziyang Luo, Yadong Xi, Rongsheng Zhang, GongZheng Li, Zeng Zhao, and Jing Ma. Conditioned masked language and image modeling for image-text dense retrieval. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 130–140, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics. 3, 4, 5

[34] Antoine Miech, Jean-Baptiste Alayrac, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 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, 2021. 3

[35] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Sixth Indian Conference on Computer Vision, Graphics & Image Processing, ICVGIP 2008, Bhubaneswar, India, 16-19 December 2008*, pages 722–729. IEEE Computer Society, 2008. 8

[36] Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. From doc2query to docTTTTTquery. *Online preprint*, 6, 2019. 2

[37] Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. Im2text: Describing images using 1 million captioned photographs. In John Shawe-Taylor, Richard S. Zemel, Peter L. Bartlett, Fernando C. N. Pereira, and Kilian Q. Weinberger, editors, *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, 2011. 6

[38] Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 2641–2649. IEEE Computer Society, 2015. 1, 2, 6, 8

[39] 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. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *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, 2021. 1, 2, 3, 4, 5, 6, 7, 8

[40] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. Okapi at TREC-3. In Donna K. Harman, editor, *Proceedings of The Third Text REtrieval Conference, TREC 1994, Gaithersburg, Maryland, USA, November 2-4, 1994*, volume 500-225 of *NIST Special Publication*, pages 109–126. National Institute of Standards and Technology (NIST), 1994. 2, 3, 7, 8

[41] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2556–2565. Association for Computational Linguistics, 2018. 6, 8

[42] Tao Shen, Xiubo Geng, Chongyang Tao, Can Xu, Xiaolong Huang, Binxing Jiao, Linjun Yang, and Daxin Jiang. LexMAE: Lexicon-bottlenecked pretraining for large-scale retrieval. In *International Conference on Learning Representations*, 2023. 3, 5

[43] Tao Shen, Xiubo Geng, Chongyang Tao, Can Xu, Kai Zhang, and Daxin Jiang. Unifier: A unified retriever for large-scale retrieval. *CoRR*, abs/2205.11194, 2022. 3

[44] Siqi Sun, Yen-Chun Chen, Linjie Li, Shuohang Wang, Yuwei Fang, and Jingjing Liu. 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, Online, June 2021. Association for Computational Linguistics. 1, 3, 4, 6, 7

[45] Liwei Wang, Yin Li, Jing Huang, and Svetlana Lazebnik. Learning two-branch neural networks for image-text matching tasks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(2):394–407, 2019. 3

[46] Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Simlm: Pre-training with representation bottleneck for dense passage retrieval. *CoRR*, abs/2207.02578, 2022. 3

[47] Keyu Wen, Jin Xia, Yuanyuan Huang, Linyang Li, Jiayan Xu, and Jie Shao. COOKIE: contrastive cross-modal knowledge sharing pre-training for vision-language representation. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 2188–2197. IEEE, 2021. 6, 7

[48] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, San Francisco, CA, USA, 13-18 June 2010*, pages 3485–3492. IEEE Computer Society, 2010. 8

[49] Peilin Yang, Hui Fang, and Jimmy Lin. Anserini: Enabling the use of lucene for information retrieval research. In Noriko Kando, Tetsuya Sakai, Hideo Joho, Hang Li, Arjen P. de Vries, and Ryen W. White, editors, *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pages 1253–1256. ACM, 2017. 6, 8

[50] Zhilin Yang, Zihang Dai, Ruslan Salakhutdinov, and William W. Cohen. Breaking the softmax bottleneck: A high-rank RNN language model. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 5

[51] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. FILIP: fine-grained interactive language-image pre-training. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. 3

[52] Xin Yuan, Zhe Lin, Jason Kuen, Jianming Zhang, Yilin Wang, Michael Maire, Ajinkya Kale, and Baldo Faieta. Multimodal contrastive training for visual representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 6995–7004. Computer Vision Foundation / IEEE, 2021. 3

[53] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 18102–18112. IEEE, 2022. 3

[54] Hui Zhang, Shenglong Zhou, Geoffrey Ye Li, and Naihua Xiu. 0/1 deep neural networks via block coordinate descent. *CoRR*, abs/2206.09379, 2022. 8

[55] Kai Zhang, Chongyang Tao, Tao Shen, Can Xu, Xiubo Geng, Binxing Jiao, and Daxin Jiang. LED: lexicon-enlightened dense retriever for large-scale retrieval. *CoRR*, abs/2208.13661, 2022. 3