

VLDeformer: Vision-Language Decomposed Transformer for Fast Cross-Modal Retrieval

Lisai Zhang¹, Hongfa Wu¹, Qingcai Chen^{*1}, Yimeng Deng^{*2}, Joanna Siebert¹, Zhonghua Li²
Yunpeng Han¹, Dejiang Kong², Zhao Cao²

¹Harbin Institute of Technology, Shenzhen, ²Huawei Technologies Co., Ltd

{LisaiZhang,hongfawu}@foxmail.com, qingcai.chen@hit.edu.cn

{dengyimeng,lizhonghua3,kongdejiang,caozhao1}@huawei.com

Abstract

Cross-modal retrieval has emerged as one of the most important upgrades for text-only search engines (SE). Recently, with powerful representation for pairwise text-image inputs via early interaction, the accuracy of vision-language (VL) transformers has outperformed existing methods for text-image retrieval. However, when the same paradigm is used for inference, the efficiency of the VL transformers is still too low to be applied in a real cross-modal SE. Inspired by the mechanism of human learning and using cross-modal knowledge, this paper presents a novel Vision-Language Decomposed Transformer (VLDeformer), which greatly increases the efficiency of VL transformers while maintaining their outstanding accuracy. By the proposed method, the cross-modal retrieval is separated into two stages: the VL transformer learning stage, and the VL decomposition stage. The latter stage plays the role of single modal indexing, which is to some extent like the term indexing of a text SE. The model learns cross-modal knowledge from early-interaction pre-training and is then decomposed into an individual encoder. The decomposition requires only small target datasets for supervision and achieves both 1000+ times acceleration and less than 0.6% average recall drop. VLDeformer also outperforms state-of-the-art visual-semantic embedding methods on COCO and Flickr30k.¹

1. Introduction

Cross-modal retrieval is important for modern applications such as search engines, social media, and e-commerce, which involves searching for instances semantically similar to the query from another modal. Just like text search engines, cross-modal retrieval requires not only high accuracy but also fast retrieval speed.

¹Our code will be publicly available upon acceptance.

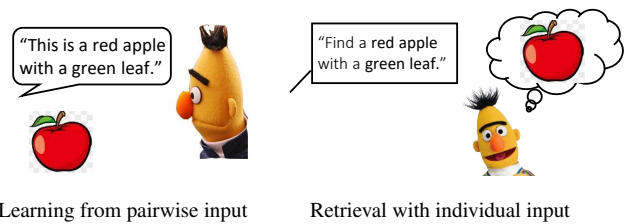


Figure 1. Illustration of the pre-training and retrieval paradigm for VLDeformer.

Vision-language transformers (VL transformers) [12, 14, 16, 19] have been known for their superior accuracy in cross-modal retrieval. These models learn cross-modal matching relationships with a BERT network [3] where the features are fused with early interaction dataflow. Similar to the process of human learning cross-modal alignment, most VL transformers compare a pair of text-images each time and produce one joint representation during pre-training. These models significantly outperform conventional visual-semantic embedding methods and prove the effectiveness of pre-training and early interaction.

However, the VL transformers still keep the paradigm for inference, whose computing costs are too high to be applied in real applications. Because of the early-interaction dataflow, VL transformers have to compute a joint representation for every text-image combination for matching. The paradigm results in $O(n^2)$ inference times on text-image data. In practice, finding the most similar pairs from $1k$ text-image records requires $500k$ inferences, which costs about 30 minutes on a modern V100 machine.

Visual-semantic embeddings models [5, 6, 10, 20, 26, 27] are another kind of mainstream cross-modal retrieval methods. They use two-branch encoders for single image and text input. In these models, late-interaction dataflow is used to learn the cross-modal alignment, which contributes to much faster retrieval efficiency than VL transformers. Dur-

ing retrieval, the embeddings could be cached and reused for other comparisons, which results in $O(n)$ inference times. The main weakness of these models is their relatively low accuracy. Recent works [8,23] attempt to address the trade-off between accuracy and speed by pre-training. However, as shown in Table 1, these models are still not as effective as VL transformers while using similar or even larger pre-training data. Until now, two-branch encoders are reported [8] to outperform VL transformers only using hundreds of times larger pre-training data. Therefore, VL transformers are still the most effective method for cross-modal retrieval but have shortcomings in retrieval speed.

The early-interaction pre-training of VL transformers is similar to humans learning the cross-modal alignment by observing pairwise language and vision inputs. However, humans are able to handle individual text and image information separately after learning, as in Fig. 1, which is a faster paradigm for retrieval. From this point of view, existing VL transformers only achieve the learning stage. To inspect the feasibility of converting a VL transformer into an individual encoder for text and image, we analyze the dataflow of VinVL [28] in a pilot experiment (Sec. 3). Surprisingly, we find an interesting phenomenon: most cross-modal interactions do not happen point-to-point, but through some “routing nodes”, as illustrated in Fig. 2. Moreover, the special tokens [CLS] and [SEP] are routing nodes most of the time. Since these special tokens do not belong to any specific modalities, it is possible to divide the dataflow by modality from them without breaking the learned dataflow. Following these findings, we prove that the pre-trained VL transformer could be decomposed into an individual encoder while maintaining most of the accuracy with only a small target dataset as supervision. We also verify in the experiment that the “routing nodes” are successfully kept after decomposition.

Combined with the pre-training stage, the proposed Vision-Language Decomposed Transformer (VLDeformer) presents a new paradigm that pre-trains a VL transformer with early-interaction dataflow and then decomposes it into an individual encoder. Using this paradigm, we can build a VL transformer of not only high accuracy but also fast speed. On the public COCO and Flickr30k benchmarks, VLDeformer achieves both 1000+ times acceleration and less than 0.6% average recall drop for VL transformer. VLDeformer also outperforms state-of-the-art visual-semantic embedding methods while using similar or even smaller pre-training data.

2. Related Work

VL Transformers Pre-trained VL transformers have shown impressive performance for many multimodal tasks. These models learn the cross-modal interaction with an early-interaction dataflow, where the text and image fea-

Type	Model	T2I	I2T	Pre-train data	Time (s)
(a)	VinVL [28]	74.6	58.1	8.8M	32537.5
(b)	LightningDOT [23]	70.0	54.0	9.5M	7.4
(b)	ALIGN-small [8]	52.0	39.2	180M	-

Table 1. R@1 retrieval score, pre-training data scale, and inference time of the state-of-the-art VL Transformer (a) and Visual-Semantic Embedding (b) models on COCO 5k test.

tures in each layer are fused with the attention mechanism. For example, ViLBERT [16] achieves early-interaction with two-branch transformer networks connected by co-attention. In the model, the outputs of the image and text of each layer are fused by a third transformer. The following models [2, 12–14, 19, 22] use single-stream architecture where the features are fused with fully-connected attention mechanisms. These models achieve improvement on cross-modal retrieval tasks with different self-supervised tasks. In common practice, the pre-trained VL transformers still keep the early-interaction dataflow while inferencing on downstream tasks, which has high computation costs in large-scale cross-modal retrieval tasks.

Visual-Semantic Embedding Models Text-image retrieval methods [7, 17, 30] usually learn embeddings for the image and text with a two-branch network. Generally, the network includes a convolutional neural network as the image encoder and a sequence model as the text encoder. Researchers have found that the fine-grained relations between the visual objects and text tokens are essential for improving the embedding quality. For example, Lee et al., [10] calculates attention between detected object features and word embedding from both visual and text view, respectively. The following work SCAN [11] explores early-interaction structure by stacked cross-attention and gets significant improvements. However, the early-interaction dataflow decreases its retrieval speed. During inference, the late-interaction dataflow enables extracting representations offline to achieve fast online retrieval. Therefore, most of the following visual-semantic embedding methods still use late-interaction dataflow; e.g, Qu et al. [29] and Zhang et al. [20] develop the interaction from global and local views.

Pre-trained Visual-Semantic Embedding Models Pre-training transformers typically use several millions of text-image pairs. Recently, some researchers have explored pre-training visual-semantic embedding methods with larger text-image pairs. For example, CLIP [21] trains a two-branch network with contrastive learning on 400 million image text pairs and achieves competitive results on various downstream tasks. ALIGN [8] expands the pre-training data even further to a larger and noisier 1,800 million scale. The model consists of an EfficientNet [24] with global pooling as the image encoder and a BERT as the text encoder and achieves higher performance than the pre-trained VL

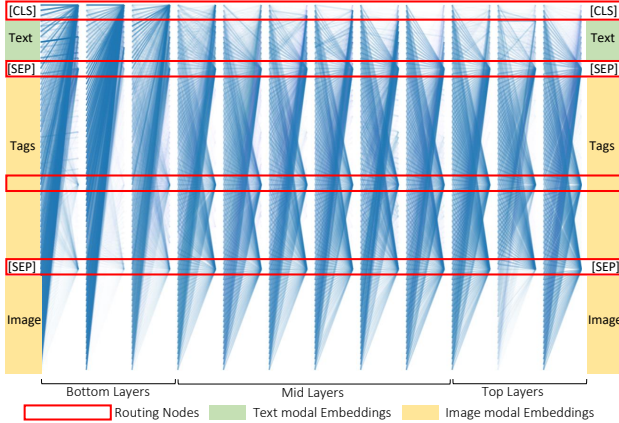


Figure 2. Illustration of the “routing nodes” inside each layers of the VL transformer VinVL [28] that receive most of the attention (better viewed with zoom in). [CLS] and [SEP] are constantly routing nodes in different samples.

transformers after fine-tuning. Although these pre-trained visual-semantic embedding models achieve advanced performance, their corpora are hundreds of times larger than VL transformers. Inspired by the pre-training of VL transformers, LightingDOT [23] tries to train two-branch transformers with a data scale closer to the data scale used by the VL transformers. The network uses a late-interaction dataflow to accelerate the inference time. However, there is still a performance gap from the VL transformers, which has to be made up by collecting top- M samples and using a VL transformer to select the final top- K ($M > K$) results.

In contrast with the above methods, we regard the early-interaction pre-training as the first stage to learning cross-modal knowledge, and the model could be decomposed as an individual encoder for fast inference. Following this process, we build a transformer that achieves both state-of-the-art accuracy and fast speed at the same time.

3. Pilot Analysis of VL Transformer

To study the dataflow inside a VL transformer, we first conduct a visualization of the attention computation of pre-trained VinVL-base [28]². The visualized samples are from the COCO 1k test set and depicted by the VIG tool [25]. As the attention map case illustrates in Fig. 2, there are four nodes that receive the most attention weights. Among them, three nodes belongs to the special tokens, i.e., [CLS] or [SEP], which do not have any modality properties. Meanwhile, the “routing nodes” are fixed in each layer. Therefore, it would be possible to decompose the vision and language inputs while keeping these paths unbroken.

We collected 1k samples and recorded the attention per-

²The phenomenon is also observed in Uniter [2], see appendix C for more cases.

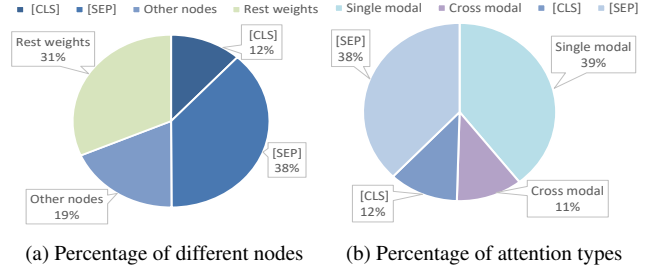


Figure 3. The proportion of attention weights in VinVL according to routing nodes (a) and modality (b) on COCO test set. Only 11% weights are cross-modal attention.

Layers	Neutral (%)			Single (%)	Cross (%)
	[CLS]	[SEP]	Total		
Bottom (1-3)	39.1	9.6	48.7	38.7	12.6
Mid (4-9)	2.6	46.0	48.6	40.7	10.6
Top (10-12)	1.5	49.1	50.6	38.6	10.8

Table 2. The attention type proportions of each VinVL layers. Only about 10% are immediate cross-modal attention.

centage according to modality and the paths to verify this phenomenon. To distinguish, we define the following:

- Routing node: The top- k tokens³ that significantly take more proportion of the attention weights than others.
- Neutral attention: The attention that starts from or points to the special tokens [CLS] and [SEP] that do not belongs to any modalities.
- Single modal attention: The attention that points from text embeddings to text embeddings or image embeddings to image embeddings.
- Cross-modal attention: The attention that points to text embeddings from image embeddings or vice versa.

As is shown in Fig. 3 (a), 69% of attention weights are on the routing nodes, and half of the attention weights are undertaken by the neutral nodes. Other nodes take a total of 19% of the weights. If the routing nodes are kept after decomposition, the pre-training knowledge is likely to be maintained. From Fig. 3 (b) we can see only 11% of the attention weights are immediate cross-modal interactions; the remaining 89% are not for immediate cross-modal interaction. If the rest of 89% attention dataflow is not broken after decomposition, we will only need small data to reconstruct the cross-modal attention and maintain most of the pre-training knowledge.

³The value k varies from different VL Transformers, e.g. $k = 4$ for VinVL and $k = 2$ for Uniter.

Table 2 shows the percentage of different attention types in each layer. We can see that single modal attention constantly takes more attention weights than cross-modal attention weights. [CLS] receives more attention in bottom layers, while on the mid and top layers, [SEP] becomes more important. Therefore, using [SEP] or the average of the layer outputs as the representation vector is likely to be more effective than the common practice to use [CLS].

4. Methodology

The paradigm of the VLDeformer is illustrated in Fig. 4. It includes early-interaction pre-training, vision-language decomposing, and retrieval stages. Since there are many ways [2, 12, 19, 28] to achieve the pre-training stage, in this section, we mainly elaborate on the principles for vision-language decomposing.

4.1. Early-interaction Pre-training

The early-interaction pre-training plays the role of learning the cross-modal alignment from large-scale datasets. To exploit the fine-grained relationship between text and image modality, the pairwise text and image input are concatenated and fed to the network simultaneously.

Pairwise Image and Text Input The pairwise text and image input includes position embedding, segment embedding, and token embedding. The input text T is tokenized as a token sequence $\{w_1, \dots, w_L\}$ where L is the length of the WordPiece [9] tokenizer output. The input image I is pre-processed by the object detection network [28] to extract region features and tags. As for the segment tokens, we assign [T] segment token to mark the word tokens and the object tags, and [V] to represent the region features. The final embedding for both text and image input is obtained by the summing up of position, segment, and token embedding, followed by a layer normalization.

The text and image features interact through self-attention in the network. The pre-training objectives are self-supervision tasks, i.e., mask language modeling, and contrastive learning for cross-modal alignment on the joint representation. Since most of the existing VL transformers are built in this pre-training procedure [2, 12, 19, 28], VLDeformer can be applied to any of them. In this section the model is trained as a VinVL-base [28] model.

4.2. Vision-language Decomposition

Individual Image and Text Input The concatenated text and image input are divided for individual encoding. Guided by the pilot analysis, there are two differences in the format: the special tokens and the position embedding. The special [CLS] token is added to the beginning of both modalities, and [SEP] is added to the end of the text and tag tokens. The position index for tokens, tags, and objects are assigned separately to distinguish the modality. The text

position index ranges from 0 to $L - 1$, while for the image input, the position index starts again from 0 to $K - 1$ where K is the number of the objects or tags.

Decomposed VL Transformer The individual image and text input isolates the cross-modal interaction of the VL transformer. To keep the other interaction as much as possible, the network shares weights for the text and image modality. According to the analysis of the pilot experiment, the [CLS] node receives small attention weights in the top layers of VL transformer, which is in conflict with the common practice to use [CLS] as representation. Therefore we experiment with three kinds of representations: [CLS], [SEP] and average pooling of all the outputs. In practice, we find the average pooling and [SEP] representation are more effective than the [CLS] vector for representation, while average pooling representation is slightly better than [SEP] (see details in Sec. 5.4). Finally, the representations \mathbf{r}_t or \mathbf{r}_v are obtained by the average pooling layer with tanh activation.

Decomposition Loss The objective of decomposition is to reconstruct the broken cross-modal interactions and learn cross-modal similarity through the late-interaction dataflow. We experiment with BCE loss, Triplet loss, and infoNCE loss, with infoNCE loss ultimately achieving the best performance (see details in Sec. 5.4). The infoNCE loss minimizes the cosine distance between semantically aligned samples and maximizes the distance between dissimilar samples. In a mini-batch with N text-image pairs, we regard the aligned pairs as the positive samples and other combinations as the negatives. We use an objective as Eq. 1 to pull semantically close images representation \mathbf{r}_v^j to the text representation \mathbf{r}_t^i and push non-close samples apart:

$$\mathcal{L}_c^t = -\log \frac{e^{\cos(\mathbf{r}_t^i, \mathbf{r}_v^i)/\tau}}{\sum_{j=1}^N e^{\cos(\mathbf{r}_t^i, \mathbf{r}_v^j)/\tau}} \quad (1)$$

where τ is a temperature hyper-parameter, and \cos is the cosine similarity $\frac{\mathbf{r}_t^i \cdot \mathbf{r}_v^j}{\|\mathbf{r}_t^i\| \cdot \|\mathbf{r}_v^j\|}$. The \mathcal{L}_c^t term can also be regarded as optimizing text-to-image retrieval in a mini-batch.

Symmetric to \mathcal{L}_c^t , we use the loss term as in Eq. 2 to learn the image-to-text condition.

$$\mathcal{L}_c^v = -\log \frac{e^{\cos(\mathbf{r}_t^i, \mathbf{r}_t^i)/\tau}}{\sum_{j=1}^N e^{\cos(\mathbf{r}_v^i, \mathbf{r}_t^j)/\tau}} \quad (2)$$

The complete contrastive learning loss is the summing up of these two terms:

$$\mathcal{L}_c = \mathcal{L}_c^t + \mathcal{L}_c^v \quad (3)$$

Since the main goal of this paper is to show the pre-training and decomposition paradigm, we find that the simple infoNCE loss is enough to maintain comparable performance to the VL transformer. Other self-supervision could also be useful, which will be left for future work.

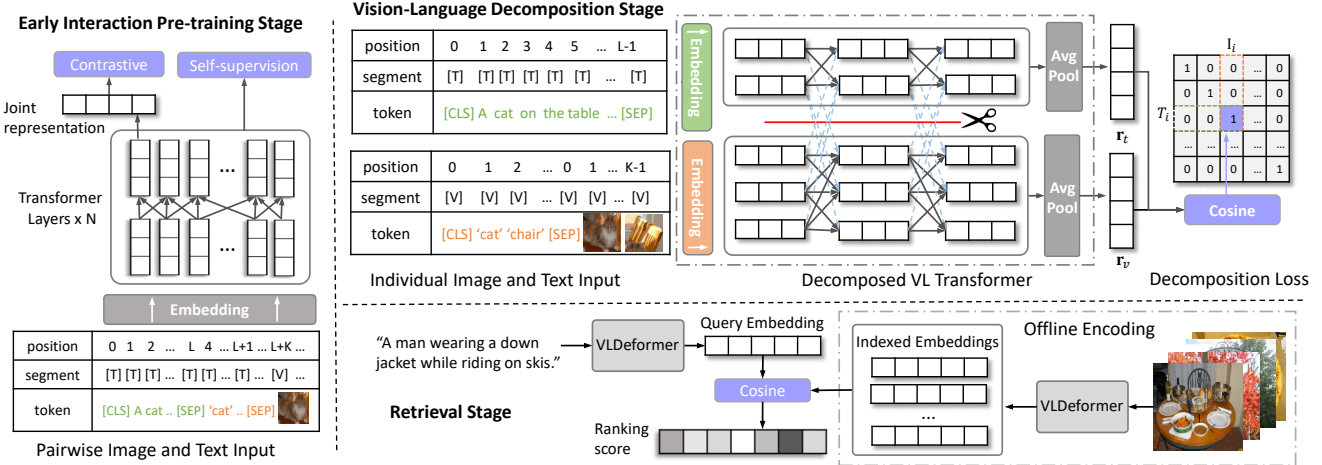


Figure 4. Overall structure of the Vision-Language Transformer Decomposing (VLDeformer). The cross-modal attention inside the transformer module is decomposed during decomposition stage (illustrated by the red line with the scissors).

4.3. VLDeformer based Cross-modal Retrieval

VLDeformer is an individual encoder and therefore enables encoding the retrieval contents offline. For example, in text-to-image retrieval, the images are encoded to embeddings offline so that the online computation only includes the query encoding and the cosine similarity, which is the main reason to achieve the retrieval speed acceleration.

In this part, we take text-to-image retrieval as an example to introduce the retrieval process. To formulate, the image set is denoted as $\{I_i\}_{i=1}^N$ where N is the image set size. The query is denoted as T .

In the offline encoding stage, the images are processed following Sec. 4.2 to get the position, segment, and token embeddings and then passed to the VLDeformer model to get the image embedding $\{r_v^i\}_{i=1}^N$. The image embeddings could be reused to compare with each text query.

During online retrieval, the query text is first processed to position, segment, and token embeddings then encoded into query embedding r_t . The index of top-1 related images to the query text is calculated as Eq. 4:

$$n = \arg \max_{i \in [0, N]} \cos(r_v^i, r_t) \quad (4)$$

The top-1 retrieved images I_n could be obtained from the image set.

4.4. Implementation Details

All the processed images are first resized to 256×256 , then 50 region-of-interests and the object tags are extracted. The max sequence length of text tokens is set to 35. The batch size for contrastive decomposing is set to 1750, while the temperature is set to 0.005. The AdamW optimizer is adopted with a learning rate of $5e^{-5}$ and weight decay of

$1e^{-4}$. The model is trained on an NVIDIA DGX with a Ubuntu 18.04 system and 8 V100 GPU.

5. Experiments

5.1. Datasets and Evaluation Protocols

Datasets The pre-training stage uses 8.8M text-image pairs from public datasets as VinVL [28]. The COCO [15] and the Flickr30k [18] datasets are used for the decomposition stage. The COCO dataset contains 123K images and is divided into 114K training, 5K validation, and 5K test images. We also use a common split of 1k tests for comprehensive evaluation. The Flickr30k dataset contains 31K images which are divided into 29K/1K/1K for training, validation, and test. Each image has 5 caption texts.

Evaluation The retrieval performance is measured by the recall at top- k samples ($R@k$). Three k values, $R@1$, $R@5$, and $R@10$, are reported for text-to-image retrieval and vice versa. We evaluate the retrieval speed for the text-to-image retrieval using 1k, 5k, and 10k text-image pairs.

5.2. Retrieval Accuracy Analysis

5.2.1 Comparison with Visual-Semantic Embeddings

Table 3 shows the comparison results of VLDeformer and visual-semantic embedding methods. Both VLDeformer and other pre-trained models substantially outperform the models without pre-training like CAAN and DIME. It is worth noting that the performance of pre-trained visual-semantic embeddings varies depending on the pre-training data scale. For example, the *ALIGN trained on the largest data outperforms the other models. However, the pre-training dataset of *ALIGN is 204 times larger than our pre-training dataset and 189 times larger than that of LightningDOT, making it hard to judge these models fairly. If we

Methods	COCO Test (5k images)						Flickr30k Test (1k images)						Pre-train data $\times 10^6$ pairs	Params $\times 10^6$
	Text Retrieval			Image Retrieval			Text Retrieval			Image 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		
Visual-Semantic Embeddings														
CAAN [29]	52.5	83.3	90.9	41.2	70.3	82.9	70.1	91.6	97.2	52.8	79.0	87.9	-	11
IMRAM [1]	53.7	83.2	91.0	39.7	69.1	79.8	74.1	93.0	96.6	53.9	79.4	87.2	-	18
SGRAF [4]	57.8	-	91.6	41.9	-	81.3	77.8	94.1	97.4	58.5	83.0	88.8	-	19
DIME [20]	59.3	85.4	91.9	43.1	73.0	83.1	81.0	95.9	98.4	63.6	88.1	93.0	-	116
Pre-trained Visual-Semantic Embeddings														
*ALIGN	77.0	93.5	96.9	59.9	83.3	89.8	95.3	99.8	99.9	84.9	97.4	98.6	1800	900
ALIGN-small [8]	52.0	-	-	39.2	-	-	-	-	-	-	-	-	180	235
LightningDOT [23]	70.0	91.6	95.5	54.0	80.8	88.5	83.9	97.2	98.6	69.9	91.1	95.2	9.5	220
VLDeformer	72.6	91.9	96.2	54.9	81.3	88.8	93.5	98.7	99.2	80.2	95.1	97.8	8.8	111

*ALIGN is achieved using 200+ times larger pre-train data than other pre-trained models, so we mainly compare with ALIGN-small.

Table 3. Comparison results with Visual-Semantic Embedding Methods on COCO and Flickr30k dataset. VLDeformer outperforms other pre-trained models using similar or even smaller data.

Methods	Flickr30k Test (1k images)							COCO Test (1k images)							Time (s)	
	Text Retrieval			Image Retrieval				AR	Text Retrieval			Image 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	AR		
LightningDOT	83.9	97.2	98.6	69.9	91.1	95.2	89.3	-	-	-	-	-	-	-	2.7	
+Reranker [23]	87.2	98.3	99.0	75.6	94.0	96.5	91.7	-	-	-	-	-	-	-	37.6	
UnicoderVL [12]	86.2	96.3	99.0	71.5	90.9	94.9	89.8	84.3	97.3	99.3	69.7	93.5	97.2	90.2	-	
Uniter [2]	86.9	98.1	99.2	75.5	94.0	96.6	91.7	-	-	-	-	-	-	-	405.3	
Oscar-base [14]	-	-	-	-	-	-	-	88.4	99.1	99.8	75.7	95.2	98.3	92.7	1300.5	
VinVL-base [28]	93.6	99.1	99.9	82.0	95.7	97.7	94.6	89.8	98.8	99.7	78.2	95.6	98.0	93.3	1301.5	
VLDeformer	93.5	98.7	99.2	80.2	95.1	97.8	94.0	89.2	98.9	99.9	75.9	95.4	98.0	92.9	1.2	

Table 4. Cross-modal retrieval comparison results to VL transformers on COCO and Flickr30k dataset. VLDeformer achieves 1000+ acceleration with less than 0.6% average recall drop.

compare the models in similar pre-training data scales, the smaller ALIGN-small model trained on 180M text-image pairs has a dramatic performance drop as the data scale decrease.

It is worth noting that VLDeformer outperforms all state-of-the-art visual-semantic embedding models when compared to similar or even smaller data sizes. Therefore we can conclude that VLDeformer is the most effective visual-semantic embeddings method on a comparable data scale.

5.2.2 Comparison with VL Transformers

Table 4 shows the retrieval score and time cost comparison between the VLDeformer network and state-of-the-art VL transformers on COCO and Flickr30k 1k test sets. Compared with the backbone VinVL-base model, VLDeformer achieves thousands of acceleration with less than 0.5% drop in average recall and can even outperform it on R@5 and R@10 levels at COCO 1k text retrieval set. VLDeformer also outperforms other VL transformers like Unicoder-VL and Uniter. Compared with the pre-trained two-branch

transformer, LightningDOT, VLDeformer achieves better results in both accuracy and inference time and also outperforms LightningDOT with an Oscar Reranker.

Still, there is a performance gap between VLDeformer and the backbone VinVL model, which is more obvious on the R@1 image retrieval score, i.e., 2.3% on COCO and 1.8% on Flickr. However, the difference on R@5 and R@10 is very small, which means that many ground truth images are not hit by the top1 result but recalled within top5 records.

5.2.3 Qualitative Case Analysis

Since the R@1 metric only calculates the hit ratio of the one aligned ground truth image, it may be inflected by other semantically similar samples. Therefore, we inspect the cases that are properly predicted by the backbone VinVL model at top1 but flipped by the VLDeformer. Fig. 5 shows the top5 retrieved images for such cases⁴. Interestingly, many images share the same semantics with the query text although

⁴See Appendix B for more detailed qualitative comparisons.

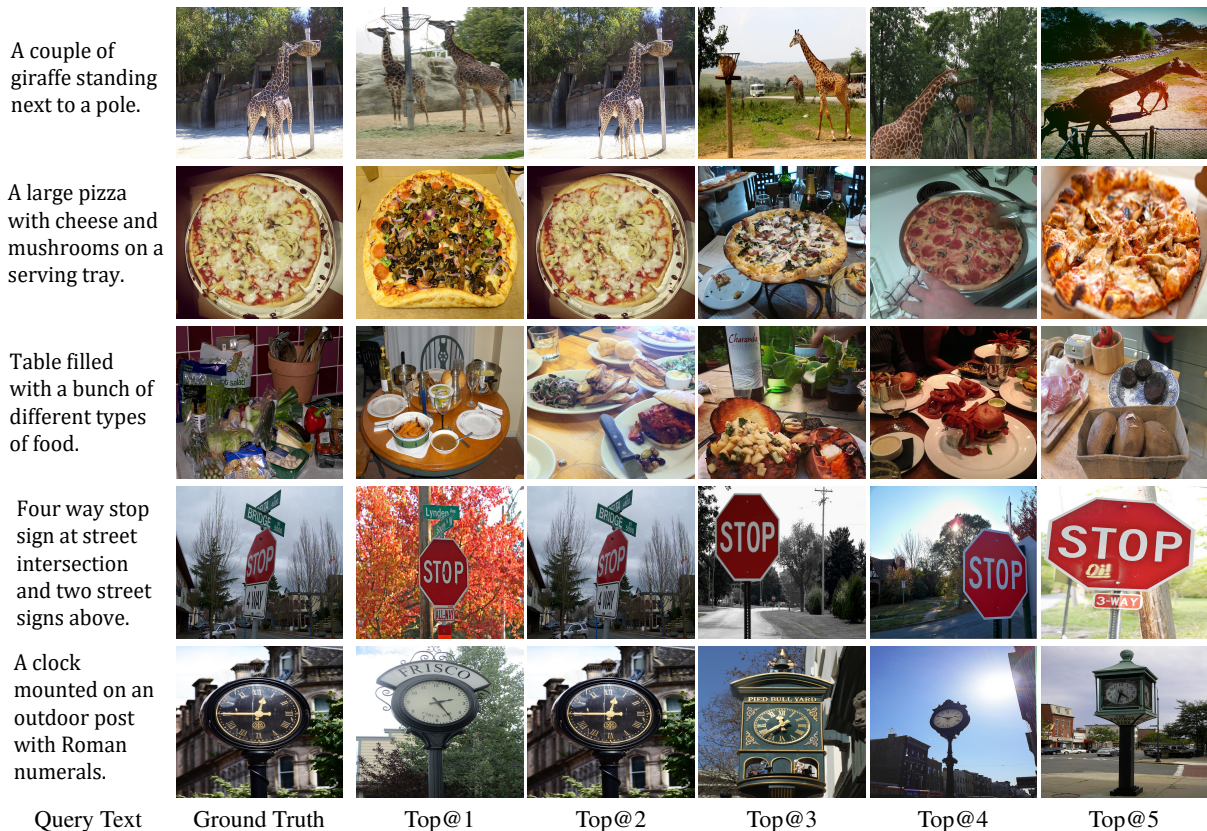


Figure 5. Retrieved top5 images where VLDeformer flips backbone VL Transformer from right to wrong in R@1.

they are not the ground truth, e.g, “two giraffes next to a pole” or “four-way stop signs”. For such queries, the top1 metric is not applicable to judge the retrieval results. Some queries like the third case have rough semantics that could be aligned with a wide range of images, e.g, “table with different food, e.g, “table with different food”. These queries also decrease the top1 metrics because it is hard to recall the ground truth at top5 or even top10 records. The samples also show some limitations of VLDeformer. For example, the fourth case mainly focuses on the “clock mounted on outdoor post” but fails to distinguish the “roman numeral” on the dial, indicating that more detailed matching is necessary for future works.

5.3. Retrieval Efficiency Analysis

The time costs to match all the text-image pairs⁵ are shown in Fig. 6. The models are compared on the same machine using one V100 GPU using 400 batch size. Only the inference time is recorded to exclude the data loading time. VinVL uses a very long time⁶ (about 0.5 hours on 1k and even more on larger data), and the time response goes

⁵We also report the single query response time in Appendix A.

⁶Since VinVL costs a very long time on 5k and 10k, the total time is estimated through the average time cost of the first 1k batches.

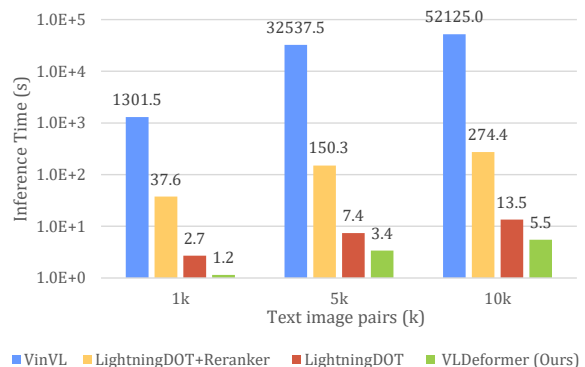


Figure 6. Text-to-image retrieval time on 1k and 5k and 10k image corpus with 400 mini-batch size.

like a quadratic function with the data size. In quantitative comparison, VLDeformer achieves more than 1k times acceleration on 1k data and 9k times on 5k than VinVL. Both VLDeformer and LightningDOT show linear time cost curves as data size increases, but LightningDOT costs more time than VLDeformer, likely because the model is built on a larger BERT-large network. It is worth noting that when LightningDOT uses an Oscar-large [14] reranker to achieve compatible accuracy to VLDeformer, its retrieval time will

Methods	Text Retrieval			Image Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10
VLDeformer	89.2	98.9	99.9	75.9	95.4	98.0
VLDeformer-[SEP]	88.5	99.1	99.8	74.9	94.8	98.1
VLDeformer-[CLS]	83.0	95.3	96.6	69.7	87.8	90.4
VLDeformer-BCE	72.5	95.8	99.0	60.7	90.2	96.2
VLDeformer-Triplet	48.9	75.5	85.0	30.6	62.3	76.4
w/o pre-training	81.5	97.3	98.8	64.8	91.6	95.9
w/o decomposition [†]	0.3	1.0	2.0	0.1	0.2	1.6

Table 5. Ablation study of VLDeformer on COCO 1k Test. (w/o decomposition[†] is tested directly using the pre-trained VinVL transformer as an individual encoder.)

increase by order of magnitude.

5.4. Ablation study

To verify the designs for VLDeformer, we conduct an ablation study over the COCO 1k test set. The compared models are trained with the same hyper-parameters. The results are shown in Table 5.

Representation selection. The results of using [CLS] and [SEP] as representation are shown in VLDeformer-[CLS] and VLDeformer-[SEP]. Compared with average pooling representation, VLDeformer-[SEP] is higher on R@5 and R@10 but lower on R@1, while VLDeformer-[CLS] decreases significantly on all metrics. The results qualify the observation in Table 2 that [CLS] is not as important as [SEP] at the top layers.

Decomposition loss selection. To evaluate the effectiveness of the infoNCE loss for decomposition, we compare BCE for pairwise cosine similarity VLDeformer-BCE and triplet loss VLDeformer-triplet. As a result, both of the two objectives perform worse than infoNCE loss, and causes dramatic performance drops, especially on R@1.

Has VLDeformer kept knowledge from the pre-training stage? The (w/o pre-training) trains VLDeformer from the vision-language decomposition stage using randomly initialized weights. The comparison between VLDeformer and (w/o pre-training) is a black box test. Since the performance drops significantly without the pre-training stage, it can be inferred that some pre-training knowledge is kept after decomposition.

Did the decomposition stage reconstruct the pre-trained model? Since the cross-modal attention of the pre-trained model is broken in VLDeformer, we wonder to what extent the decomposition reconstructs the pre-trained model. (w/o decompose[†]) shows the performance of directly using the VinVL model as an individual embedding encoder. To our surprise, the scores are very low, indicating that the decomposing stage is necessary to maintain the performance.

Model	Modal	Layers	Neutral Route (%)			Others (%)
			[CLS]	[SEP]	total	
w/ pre-train	V	Bottom (1-3)	66.3	8.1	74.3	25.6
		Mid (4-9)	6.1	77.9	84.0	16.0
		Top (10-12)	14.2	51.3	65.5	34.5
		Total	20.5	58.1	78.6	21.3
w/o pre-train	L	Bottom (1-3)	48.0	9.6	57.7	42.3
		Mid (4-9)	3.5	70.6	74.1	25.9
		Top (10-12)	1.8	54.4	56.2	43.8
		Total	14.2	51.3	65.6	34.4
w/o pre-train	V	Total	4.1	6.6	10.7	89.2
	L	Total	6.5	6.5	13.0	87.0

Table 6. The proportion of different attention weights in each layer of VLDeformer with (w/) and without (w/o) pre-training stage.

5.5. Decomposition Analysis

We further analyze the attention of VLDeformer after decomposition to verify the observation and hypothesis in Sec. 3. In Table 6, we can find that the [CLS] and [SEP] are still important routing nodes which have large proportions of attention weights. In contrast, in the VLDeformer without pre-training, these nodes are not routing nodes. In Fig. 7 we compare VLDeformer without pre-train from the same sample⁷ as in Fig. 2. It can be clearly seen that VLDeformer keeps the routing nodes [CLS] and [SEP], but there is no clear routing node in the VLDeformer without pre-training.

6. Conclusion

VL transformers are effective in cross-modal retrieval but slow in speed. We observed that most of the interactions in VL transformers are not immediate cross-modal attention, but highly rely on neutral nodes. Therefore we proposed a novel Vision-language Decomposed Transformer (VLDeformer) that pre-trains a VL transformer with early-interaction dataflow and then decomposes it into an individual encoder. The VLDeformer achieves both 1000+ times acceleration and less than 0.6% average recall drop and also outperforms state-of-the-art visual-semantic embedding models on COCO and Flickr30k datasets.

References

- [1] Hui Chen, Guiguang Ding, Xudong Liu, Zijia Lin, Ji Liu, and Jungong Han. Imram: Iterative matching with recurrent attention memory for cross-modal image-text retrieval. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12652–12660, 2020. 6
- [2] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European*

⁷See more visualization samples in the supplementary files.

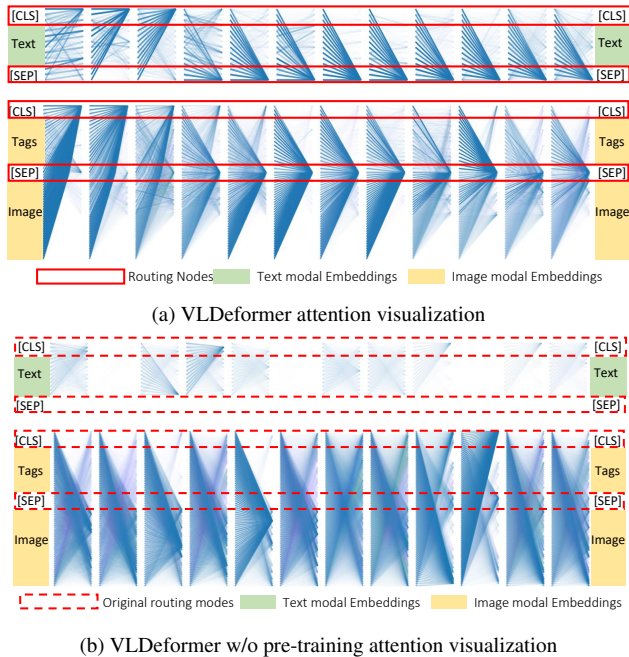


Figure 7. Attention visualization of VLDeformer with and without pre-training. (From the same sample in Fig. 2).

conference on computer vision, pages 104–120, 2020. 2, 3, 4, 6

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019. 1

[4] Haiwen Diao, Ying Zhang, Lin Ma, and Huchuan Lu. Similarity reasoning and filtration for image-text matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1218–1226, 2021. 6

[5] Aviv Eisenschat and Lior Wolf. Linking image and text with 2-way nets. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4601–4611, 2017. 1

[6] Kiros Faghri, Fleet and Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. In *BMVC*, page 12, 2018. 1

[7] Jiuxiang Gu, Jianfei Cai, Shafiq R Joty, Li Niu, and Gang Wang. Look, imagine and match: Improving textual-visual cross-modal retrieval with generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7181–7189, 2018. 2

[8] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. *arXiv preprint arXiv:2102.05918*, 2021. 2, 6

[9] Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas,

Martin Wattenberg, Greg Corrado, et al. Google’s multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351, 2017. 4

[10] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision*, pages 201–216, 2018. 1, 2

[11] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 201–216, 2018. 2

[12] Gen Li, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11336–11344, 2020. 1, 2, 4, 6

[13] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 2

[14] Xiujuan Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137, 2020. 1, 2, 6, 7

[15] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 5

[16] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*, 2019. 1, 2

[17] Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. Dual attention networks for multimodal reasoning and matching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 299–307, 2017. 2

[18] 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 *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015. 5

[19] Di Qi, Lin Su, Jia Song, Edward Cui, Taroon Bharti, and Arun Sacheti. Imagebert: Cross-modal pre-training with large-scale weak-supervised image-text data. *arXiv*, pages 1–12, 2020. 1, 2, 4

[20] Leigang Qu, Meng Liu, Jianlong Wu, Zan Gao, and Liqiang Nie. Dynamic modality interaction modeling for image-text retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1104–1113, 2021. 1, 2, 6

[21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learn-

- ing transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021. [2](#)
- [22] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vi-bert: Pre-training of generic visual-linguistic representations. In *International Conference on Learning Representations*, 2019. [2](#)
- [23] 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, 2021. [2](#), [3](#), [6](#)
- [24] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019. [2](#)
- [25] Jesse Vig. A multiscale visualization of attention in the transformer model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 37–42, Florence, Italy, 2019. Association for Computational Linguistics. [3](#)
- [26] Liwei Wang, Yin Li, Jing Huang, and Svetlana Lazebnik. Learning two-branch neural networks for image-text matching tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):394–407, 2018. [1](#)
- [27] Yaxiong Wang, Hao Yang, Xiuxiu Bai, Xueming Qian, Lin Ma, Jing Lu, Biao Li, and Xin Fan. Pfan++: Bi-directional image-text retrieval with position focused attention network. *IEEE Transactions on Multimedia*, pages 1–1, 2020. [1](#)
- [28] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5579–5588, 2021. [2](#), [3](#), [4](#), [5](#), [6](#)
- [29] Qi Zhang, Zhen Lei, Zhaoxiang Zhang, and Stan Z Li. Context-aware attention network for image-text retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3536–3545, 2020. [2](#), [6](#)
- [30] Zhedong Zheng, Liang Zheng, Michael Garrett, Yi Yang, Mingliang Xu, and Yi-Dong Shen. Dual-path convolutional image-text embeddings with instance loss. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 16(2):1–23, 2020. [2](#)