




Enhancing Separate Encoding with Multi-layer Feature Alignment for Image-Text Matching

Keyu Wen¹, Linyang Li², and Xiaodong Gu¹(✉) 

¹ Department of Electronic Engineering, School of Information Science and Technology, Fudan University, Shanghai 200438, China
{kywen19, xdgu}@fudan.edu.cn

² Shanghai Key Laboratory of Intelligent Information Processing, School of Computer Science, Fudan University, Shanghai 200438, China
lyli19@fudan.edu.cn

Abstract. There is a surge of interest in cross-modal representation learning, concerning mainly images and texts. Image-Text Matching task is one major challenge in cross-modal tasks. Traditional methods use multi-paths to encode features across modalities separately and project them into a shared latent space. Recently, the development of pre-trained models inspires people to learn cross-modal features jointly and boost performances through large-scale data. However, traditional methods are less effective when both modalities use pre-trained uni-modal encoders. Methods that encode features jointly would face an unacceptable calculation cost during inference, thus less valuable for real-time applications. In this paper, we first explore the pros and cons of these methods, then we propose an enhanced separate encoding framework, using an extra encoding process to project multi-layer features of pre-trained encoders into a similar latent space. Experiments show that our framework outperforms current methods that do not use large-scale image-text pairs in both Flickr30K and MS-COCO datasets while maintaining minimal cost during inference.

Keywords: Image-text matching · Separate encoding · Cross modal

1 Introduction

With the development of deep learning, neural networks achieve great progress in computer vision and natural language processing. Cross-modal tasks, mainly between images and texts, are gaining more and more attention [5]. In this work, we focus on one major task in cross-modal learning: image-text matching.

The goal of the image-text matching task is to find the most matching pairs through a large number of given images and texts. Thus, in real-time applications, it is vital to find the best matches of the given images/texts efficiently.

K. Wen and L. Li—Equal contribution.

© Springer Nature Switzerland AG 2021

I. Farkaš et al. (Eds.): ICANN 2021, LNCS 12891, pp. 403–414, 2021.

https://doi.org/10.1007/978-3-030-86362-3_33

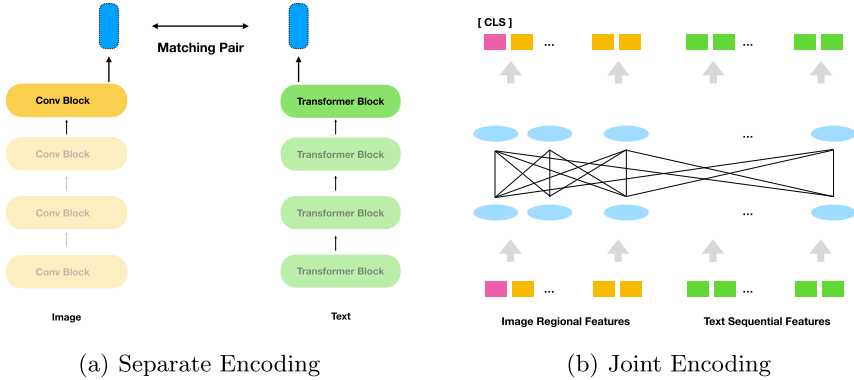


Fig. 1. Different encoding methods

Traditional solutions in deep learning are to find a shared latent space [25] by encoding image and text features separately. Normally, convolution-based [11] networks are used to encode images while RNN-based [8] networks such as LSTM [12] are applied for text encoding. Then the distance measurements like cosine similarity are used to calculate the similarity of the pooled vectors from different modalities. A triplet ranking loss [25] is then applied to train the neural network for finding the most similar pairs across modalities. These architectures can be illustrated by Fig. 1(a). As shown, features across modalities are isolated since they are separately encoded.

Recently, there has been much progress achieved with the development of pre-trained models in different modalities. These improvements make it possible to joint-encode the features across modalities to learn a joint representation of vision and language.

Pre-trained models push the state-of-the-art performances of many tasks to a new level. In the CV field, the pre-trained models, such as VGG [24] and ResNet [11], have been regarded as the backbone models to extract the visual features for the downstream tasks. In the NLP field, the pre-trained models, exemplified by ELMo [19], GPT [21] and BERT [6], use fine-tuning method to achieve new state-of-the-art performances in downstream tasks like natural language inference [2].

The arise of pre-trained encoders allows separate encoding to encode single modal features with higher representation quality. However, the distribution of pre-trained encoders are different across modalities, thus the traditional usage of bottom-up structures (Fig. 1(a)) would make it difficult to project cross-modality features into a shared latent space.

Later in the cross-modal field, following the idea of applying large-scale data to create pre-trained models, joint encoding methods are based on large-scale image-text paired data [15]. This architecture, shown in Fig. 1(b), combines texts and images together through an attention-based structure [27] encoder to learn joint representations across two modalities. These models, exemplified

by Unicoder-VL [15], UNITER [4], achieve new state-of-the-art results on many cross-modal tasks like VQA [1], image-captioning [3] as well as image-text matching [25]. In image captioning and VQA tasks, the goal is to generate corresponding captions or to find answer spans, which requires images and texts to entangle with each other. Joint-encoding models boost these tasks to a whole new level.

However, in the image-text matching task, the goal is to find the most matching pair from a large number of images and texts.

Since joint encoding methods combine the texts and images as inputs to the model, during inference, these models require the pre-trained structure to iterate all possible pairs which take massive calculation consumption. We name such unacceptable cost *Inference Disaster*. Such a problem constrains these models in real-time usage despite its outstanding performance.

As illustrated above, in the image-text matching task, traditional methods are relatively weak in representation encoding compared with joint-encoding methods based on pre-training with large-scale image-text pairs. Meanwhile, the joint-encoding methods suffer from the inference disaster.

In this work, in order to maintain the retrieval efficiency as well as promoting the performance of the model, we propose an **Enhanced Separate Encoding Framework** to modify the separate encoding framework, focusing on excavating multi-layer features of separate pre-trained visual and textual encoders and projecting them to the common subspace.

Our proposed framework is constructed based on separate encoding models, thus is very efficient during inference compared with the joint-encoding methods.

We attach extra encoding modules to align and project features across modalities. These extra modules extract features from the entire pre-trained encoder in different modalities and project them in a shared latent space, thus the representations across modalities are less distant compared with separate pre-trained features.

Experiments show that our proposed framework achieves competitive performances against joint-encoding methods without using large-scale image-text pairs for pre-training and outperforms all previous traditional separate-encoding methods in Flickr30K and MS-COCO dataset.

To summarize our Contributions:

- (a) We analyze the traditional separate-encoding methods as well as recent joint-encoding methods, pointing out the importance of both performances and efficiency in the image-text matching task.
- (b) We propose a framework to break the limit of separate encoding methods. The framework outperforms all previous separate encoding methods and achieves competitive performances against joint-encoding methods, meanwhile, it does not use large-scale image-text pairs.

2 Related Work

2.1 Traditional Methods in Image-Text Matching

Encoding features from different modalities separately is the major method used before. The goal is to find a better shared latent space of image features and text features. Triplet ranking loss is introduced by [25] and used to narrow down the distance between matching pairs. [9] incorporated a hard negative method to focus on maximum violating negative pairs, which is widely applied by later works. More recently, [14, 28] introduced faster-RCNN network to use regional semantic features to enhance the image encoding quality. Other approaches such as incorporating knowledge graphs [23], using graph networks [16, 29] are explored to further boost the performances. Most of these methods encode image features with pre-trained models such as ResNet and faster-RCNN, while encoding text features with RNNs. Thus, when incorporating pre-trained text encoders, it is more difficult to learn a shared latent space in two different distributions from pre-trained encoders across modalities.

2.2 Pre-trained Models and Joint-Encoding

In computer vision field, ResNet [11] and VGG [24] are widely used as backbones in vision models. These convolution-based structure models are trained using image classification data such as ImageNet. Models like Fast RCNN [10], Faster RCNN [22] are built based on these backbone models and aim for detection and segmentation tasks.

Recent arise of pre-trained models in natural language processing started with ELMo [19], using unsupervised data to train language models. GPT [21] and BERT [6] introduce the attention-based structure called transformer [27], take the NLP research into a new era of pre-training. These successes of pre-trained models motivate researchers to construct cross-modal pre-trained models using large-scale cross-modal datasets. These models use pre-calculated regional features combined with text sequences to create joint-encoded features, exemplified by UNITER [4], Unicoder-VL [15] and LXMERT [26]. These models achieve great performances in cross-modal tasks such as VQA, image captioning; yet in the image-text matching task, the inference efficiency is limited by its joint-encoding nature.

3 Limits of Previous Encoding Methods

3.1 Different Distribution in Separate Encoding

When both modalities are equipped with pre-trained encoders, exemplified by ResNet in images and BERT in texts, the distribution is different inherently, making previous methods difficult to project different modalities into a shared latent space.

3.2 Inference Disaster in Joint Encoding

Joint-Encoding models use large-scale image-text paired data to pre-train the joint-encoding models [4, 15, 17, 26, 30].

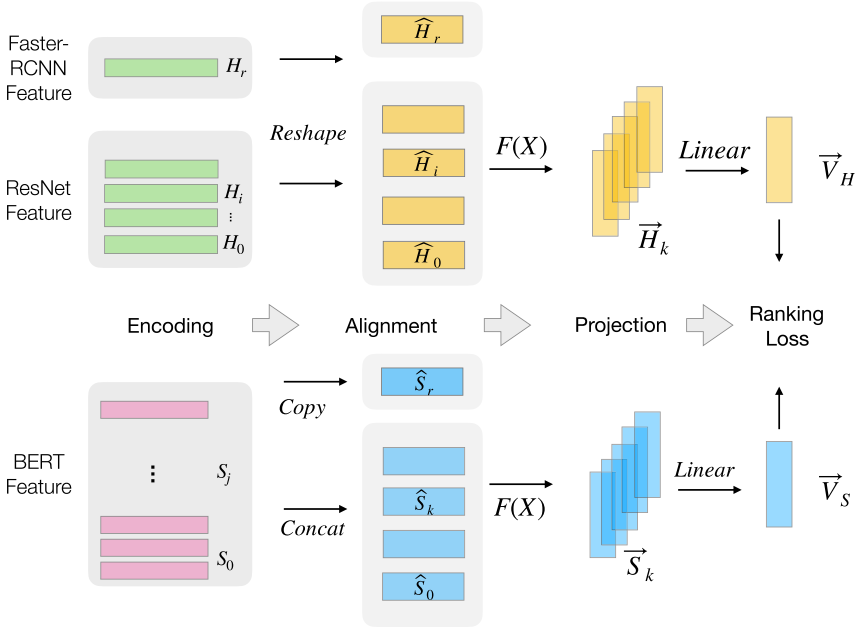


Fig. 2. Structure of enhanced separate encoding framework

Most of these methods firstly encode input image region features that are extracted from an RCNN model trained with [13]. These regional features from original images play roles as tokens in a sequence.

Despite the excellent performances in downstream tasks, such structures would face a massive calculation consume problem during inference in the matching task: Suppose N sequences(captions) and M images are to be examined, which are total $M \times N$ entangled pairs. Suppose the inference time for each pair is T , with a batch-size B . The model needs to went through $M \times N$ times inference, resulting in a time cost $\frac{M \times N \times T}{B}$, which has an $O(n^2)$ time complexity. While inference with separately encoded features only need to run a cosine similarity between pairs, which has an $O(n)$ time complexity.

4 Framework Construction

Separate encoding methods would be less effective in applying pre-trained encoders in both modalities, considering that features in two modalities are

under different distribution; meanwhile, joint encoding methods, though encoding jointly, would suffer from a less efficient inference process. Leveraging advantages and disadvantages, we propose an enhanced separate encoding method, aiming to narrow down the distance between features from two different pre-trained encoders. The core motivation is that allowing separately pre-trained features to be further encoded by non-pre-trained modules, thus these features are more similar in nature since these non-pre-trained modules are more aligned.

Therefore, we construct extra modules to align and extract pre-trained cross-modality multi-layer features and train these modules from scratch to learn a shared latent space (Fig. 2).

The entire enhanced separate encoding framework consists of three steps: feature encoding, feature alignment, and feature projection.

4.1 Feature Encoding

First, we obtain the multi-layer features of separate pre-trained encoders.

Separate encoding features are trained with different types of corpora. In image pre-training, ResNet is trained with image classification data and the feature map of ResNet can be used as the backbone of further downstream tasks. Faster-RCNN model is trained with object detection data or semantic segmentation data and the output feature is regional features of a given image. In text pre-training, BERT is trained with a mask language model, using large-scale Wikipedia corpus. Based on the transformer structure, the output is the multi-layer token-level feature.

We use all levels of separately pre-trained features combined to find better cross-modal representations: In image encoding, we denote the i^{th} layer of feature map from ResNet as $H_i \in \mathbb{R}^{W_i \times H_i \times D_i}$; W_i, H_i are the width and height size of the convolution output. We denote the regional feature from faster-RCNN as $H_r \in \mathbb{R}^{N^r \times D_r}$ and N^r is the region number. In text encoding, we denote the j^{th} layer of transformer block output from BERT as $S_j \in \mathbb{R}^{L \times D_j}$, L is the sequence length.

These obtained features are encoded separately from pre-trained models, thus are quite different across modalities.

4.2 Feature Alignment

In text encoding, the output feature is token-level, which is sub-word level feature in BERT specifically. In image encoding, the output features are feature-maps extracted from ResNet features and regional features extracted from RCNN network features. Therefore, it is difficult to directly project these features with different layers and different dimensions into a shared latent space. We manage to convert different layers of features into aligned regional features across modalities by reshaping them via feature concatenation and average pooling.

4.3 Feature Projection

After feature alignment, we have multi-level regional image features and multi-level sub-word textual features. The feature projection is a two-phase process:

Region/Token-Wise Projection. First we project both region features in encoding images and token features in encoding texts into a similar latent space. The token-region matching can be better encoded with attention-based modules as explored by [6, 14, 15], thus we construct a self-attention based encoder to encode these aligned features.

The encoder $F(X)$ follows a standard transformer structure [27].

$$A = \text{Softmax}\left(\frac{W_q X W_k^T X}{\sqrt{d}}\right)(W_v X) \quad (1)$$

$$F(X) = \text{LayerNorm}(X + A + \text{FFN}(A)) \quad (2)$$

We feed the aligned feature \widehat{H}_i , \widehat{H}_r from image encoder and \widehat{S}_i from text encoder into corresponding transformer blocks to get token/region level features. Considering that we have both ResNet features and faster-RCNN features combined, we duplicate the last layer of \widehat{S}_k to create \widehat{S}_r to match the corresponding \widehat{H}_r . We then apply average pooling over the region/token level representations to obtain vectors of the given image and text.

$$\vec{H}_i = \text{AvgPool}(F_i(\widehat{H}_i)), \vec{H}_r = \text{AvgPool}(F_r(\widehat{H}_r)), \vec{S}_k = \text{AvgPool}(F_k(\widehat{S}_k)) \quad (3)$$

Layer-Wise Projection. As mentioned in feature alignment, we use layer concatenation to align multi-level features, which is rigid in nature. We are unaware which level of features across modalities might be encoded more similar, thus we fully connect these vectors, allowing different level of features to match their potential similar features across modalities.

$$\vec{V}_H = \text{Linear}(\text{Concat}([\vec{H}_0, \dots, \vec{H}_i, \dots], \vec{H}_r)) \quad (4)$$

$$\vec{V}_S = \text{Linear}(\text{Concat}([\vec{S}_0, \dots, \vec{S}_k, \dots], \vec{S}_r)) \quad (5)$$

These two steps of feature projection encode the features that are inherently different into a similar latent space. Since joint-encoding the concatenated token and region features are not feasible in separate encoding, we decompose the separate encoding features into token-wise and layer-wise, and align them to be encoded into a more similar latent space.

After acquiring the separate encoded vectors \vec{V}_H and \vec{V}_S from two modalities, we use triplet ranking loss to train the entire model.

5 Experiment

5.1 Datasets

We use Flickr30K [20] dataset and MS-COCO [18] to test our enhance separate encoding framework.

In Flickr30K, there are 31,783 images with 5 captions each, and MS-COCO 2014 contains 123,287 images with 5 captions per image. We follow [9] for the train-valid-test split, which is 1k test for Flickr30K, 1k, and 5k for MS-COCO. which results in 113287 training, 5000 validation, and 5000 testing images for MS-COCO. Flickr30K dataset is split into 29783 training, 1000 validation, and 1000 testing images. Our results average over 5 folds of 1k test images and use the full 5000 test images for MS-COCO testing. We use recall by K (R@K) defined as the fraction of queries for which the correct item is retrieved in the closest K points to the query.

5.2 Implementation Details

For both Flickr30K and COCO dataset, we use ResNet152 and Faster-RCNN with ResNet101 as image encoding models. The Faster-RCNN features are extracted following [30], with region number 100 and hidden size 2048. The dimension of 4 layers of feature maps in ResNet152 are [56, 56, 256], [28, 28, 512], [14, 14, 1024] and [7, 7, 2048]. We apply average pooling with pooling window [8, 8], [4, 4], [2, 2] and [1, 1]. After merging and linear transformation, the output features of 4 feature maps are [49, 256], [49, 256], [49, 512], [49, 1024]. The region feature is [100, 1024]. And we use BERT-base as a text encoding model, which contains 12 layers with hidden dimension size 768. We set max sequence length to 32. During feature alignment, we concatenate every 3 layers of BERT output and use linear transformation to obtain 4 layers of features with dimension size [32, 256], [32, 256], [32, 512] and [32, 1024]. We duplicate the last layer to align with region features from faster-RCNN. The transformer block is a 1-layer transformer with 8 heads and an intermediate size 1024.

During training, we use NVIDIA 1080Ti GPUs to train the entire model, with learning rate set to $2e-5$, batch-size 128 for Flickr30K, and 320 for MS-COCO dataset. We also ensemble two single models to create an ensemble model of an enhanced separate encoding framework to boost the performances.

5.3 Experiment Setup

We establish baselines testing the matching results as well as inference cost. We implement joint-encoding approaches based on two different joint-encoding structures. In the Unicoder-VL structure, we follow the implementation in [15]. In the LXMERT structure, the core idea is encoding features across modalities jointly only in the higher layers. Thus, we use the first 8 layers of BERT-base structure for text encoding and region-features from Faster-RCNN for image encoding. Then we concatenate the image and text features and feed them into

the last 4 layers of BERT-base structure and use the special [CLS] token for similarity score learning.

The inference cost is tested on a single NVIDIA 1080Ti GPU. We set batch-size 128 evaluating our enhanced separate encoding framework. When evaluating joint-encoding methods on 1k test of Flickr30K dataset, we use batch-size 5000 which is the caption number; we iterate each image to calculate the similarity score of the matching pairs.

Table 1. Performances on Flickr30K dataset Unicoder-VL* is further pre-trained with large-scale image-text pairs.

Methods	Image-to-text			Text-to-image			Inference cost	
	R@1	R@5	R@10	R@1	R@5	R@10	Time cost	GPU cost
Joint-encoding methods								
Unicoder-VL [15]	73.0	89.0	94.1	57.8	82.2	88.9	8800 (s)	8X
LXMERT [26]	73.3	92.5	96.5	53.6	81.4	89.0	5807 (s)	6X
Unicoder-VL*	86.2	96.3	99.0	71.5	90.9	94.9	–	–
Separate-encoding methods								
VSE++ [9]	52.9	80.5	87.2	39.6	70.1	79.5	–	–
SCAN [14]	67.4	90.3	95.8	48.6	77.7	85.2	–	–
SCG [23]	71.8	90.8	94.8	49.3	76.4	85.6	–	–
VSRN [16]	71.3	90.6	96.0	54.7	81.8	88.2	–	–
SGRAF [7]	77.8	94.1	97.4	58.5	83.0	88.8	–	–
Ours	79.4	94.9	97.5	63.3	88.0	92.3	61.5 (s)	1X
Ours [ensemble]	80.9	95.5	97.9	66.0	88.8	93.1	63.1 (s)	2X

5.4 Experiment Result

As seen in Table 1 and 2, our enhanced separate encoding framework outperforms previous separate encoding approaches by a large margin, while outperforming joint encoding methods that are trained without image-text pair pre-training.

The calculation cost during inference, as seen in Table 1, is enormous in joint-encoding methods. We use 8 GPUs to run inference in joint-encoding with very large batch-size, still the time cost is unbearable. Meanwhile, without pre-training, the performance of joint-encoding is not superior to separate encoding methods.

Joint-encoding model further pre-trained with large-scale image-text pairs has great performances while it has less competitive performances when only trained with image-text pairs in the given task. This indicates that joint-encoding method relies on using large-scale image-text pairs to enhance the model while joint- Therefore, we believe that separate encoding with our enhanced framework is both effective and efficient.

Table 2. Results on MS-COCO dataset.

Methods	Image-to-text			Text-to-image			Image-to-text			Text-to-image		
	1K test images						5K test images					
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Joint-encoding methods												
Unicoder-VL	75.1	94.3	97.8	63.9	91.6	96.5	–	–	–	–	–	–
Unicoder-VL*	84.3	97.3	99.3	69.7	93.5	97.2	62.3	87.1	92.8	46.7	76.0	85.3
Separate-encoding methods												
VSE++	64.6	90.0	95.7	52.0	84.3	92.0	41.3	71.1	81.2	30.3	59.4	72.4
SCAN	72.7	94.8	98.4	58.8	88.4	94.8	50.4	82.2	90.0	38.6	69.3	80.4
SCG	76.6	96.3	99.2	61.4	88.9	95.1	56.6	84.5	92.0	39.2	68.0	81.3
VSRN	76.2	94.8	98.2	62.8	89.7	95.1	53.0	81.1	89.4	40.5	70.6	81.1
SGRAF	79.6	96.2	98.5	63.2	90.7	96.1	57.8	–	91.6	41.9	–	81.3
Ours	79.7	96.7	98.7	64.7	90.0	95.1	57.2	84.5	91.4	41.5	72.1	82.0
Ours [ensemble]	80.4	97.0	98.8	65.5	90.8	95.7	58.6	85.6	92.2	42.7	73.4	83.2

Table 3. Projection study on Flickr30K dataset; R/T-P is region/token-wise projection; L-P is layer-wise projection.

Projection		Image-to-text			Text-to-image		
R/T-P	L-P	R@1	R@5	R@10	R@1	R@5	R@10
		73.9	93.6	96.0	58.0	85.4	90.8
✓		76.1	93.4	96.4	61.4	86.4	91.8
	✓	75.5	93.1	96.4	59.5	85.7	91.3
✓	✓	79.4	94.9	97.5	63.3	88.0	92.3

6 Ablation Studies

6.1 Effectiveness of Feature Projection

The motivation of our enhanced separate encoding framework is to project separately pre-trained features into a similar latent space. Therefore, we construct ablations studies proving that feature projection modules play vital roles in our framework.

We establish baselines on both Flickr30K and COCO dataset. We concatenate the pooled \hat{H} and \hat{S} without using $F(X)$ region/token-wise projection or layer-wise linear transformation projection. That is we run baselines without feature projection, we simply use concatenated outputs features from feature align process.

As seen in Table 3, $F(X)$ projection (R/T-P) and linear transformation (L-P) are important in projecting features to be more similar, indicating that though the pre-trained features possess abundant information, they are different inherently across modalities. Therefore, though both projection methods are easy to construct, the idea of allowing separately-pre-trained features to be aligned and further encoded is extremely effective.

7 Conclusions and Future Work

In this paper, we focus on the image-text matching task. Firstly, we analyze the traditional separate encoding methods as well as recent joint-encoding methods based on pre-training with large-scale image-text pairs. We discuss the problems that constrain these methods, then we propose a framework to leverage the advantages and disadvantages of these methods, achieving competitive results while maintaining a minimal inference cost.

In the future, following our analysis, we are hoping to apply large-scale image-text pairs to train the projection modules to take performances of the image-text matching task to a higher level as well as try different languages.

Acknowledgement. This work was supported in part by National Natural Science Foundation of China under grants 61771145.

References

1. Antol, S., et al.: VQA: visual question answering. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2425–2433 (2015)
2. Bowman, S.R., Angeli, G., Potts, C., Manning, C.D.: A large annotated corpus for learning natural language inference. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics (2015)
3. Chen, X., et al.: Microsoft coco captions: data collection and evaluation server. arXiv preprint [arXiv:1504.00325](https://arxiv.org/abs/1504.00325) (2015)
4. Chen, Y.C., et al.: UNITER: learning universal image-text representations. arXiv preprint [arXiv:1909.11740](https://arxiv.org/abs/1909.11740) (2019)
5. Cheng, Q., Gu, X.: Bridging multimedia heterogeneity gap via graph representation learning for cross-modal retrieval. *Neural Netw.* **134**, 143–162 (2021)
6. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR* abs/1810.04805 (2018). <http://arxiv.org/abs/1810.04805>
7. Diao, H., Zhang, Y., Ma, L., Lu, H.: Similarity reasoning and filtration for image-text matching. arXiv preprint [arXiv:2101.01368](https://arxiv.org/abs/2101.01368) (2021)
8. Elman, J.L.: Finding structure in time. *Cogn. Sci.* **14**(2), 179–211 (1990)
9. Faghri, F., Fleet, D.J., Kiros, J.R., Fidler, S.: VSE++: improved visual-semantic embeddings. *CoRR* abs/1707.05612 (2017). <http://arxiv.org/abs/1707.05612>
10. Girshick, R.: Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. *CoRR* abs/1512.03385 (2015). <http://arxiv.org/abs/1512.03385>
12. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
13. Krishna, R., et al.: Visual genome: connecting language and vision using crowd-sourced dense image annotations. *CoRR* abs/1602.07332 (2016). <http://arxiv.org/abs/1602.07332>
14. Lee, K., Chen, X., Hua, G., Hu, H., He, X.: Stacked cross attention for image-text matching. *CoRR* abs/1803.08024 (2018). <http://arxiv.org/abs/1803.08024>

15. Li, G., Duan, N., Fang, Y., Jiang, D., Zhou, M.: Unicoder-VL: a universal encoder for vision and language by cross-modal pre-training. arXiv preprint [arXiv:1908.06066](https://arxiv.org/abs/1908.06066) (2019)
16. Li, K., Zhang, Y., Li, K., Li, Y., Fu, Y.: Visual semantic reasoning for image-text matching. In: ICCV (2019)
17. Li, L.H., Yatskar, M., Yin, D., Hsieh, C.J., Chang, K.W.: VisualBERT: a simple and performant baseline for vision and language. arXiv preprint [arXiv:1908.03557](https://arxiv.org/abs/1908.03557) (2019)
18. Lin, T., et al.: Microsoft COCO: common objects in context. CoRR abs/1405.0312 (2014). <http://arxiv.org/abs/1405.0312>
19. Peters, M.E., et al.: Deep contextualized word representations. CoRR abs/1802.05365 (2018). <http://arxiv.org/abs/1802.05365>
20. Plummer, B.A., Wang, L., Cervantes, C.M., Caicedo, J.C., Hockenmaier, J., Lazebnik, S.: Flickr30k entities: collecting region-to-phrase correspondences for richer image-to-sentence models. IJCV **123**(1), 74–93 (2017)
21. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training (2018). <https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/languageunsupervised/languageunderstandingpaper.pdf>
22. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. CoRR abs/1506.01497 (2015). <http://arxiv.org/abs/1506.01497>
23. Shi, B., Ji, L., Lu, P., Niu, Z., Duan, N.: Knowledge aware semantic concept expansion for image-text matching. In: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, pp. 5182–5189. International Joint Conferences on Artificial Intelligence Organization, July 2019. <https://doi.org/10.24963/ijcai.2019/720>
24. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv 1409.1556 (09 2014)
25. Socher, R., Karpathy, A., Le, Q.V., Manning, C.D., Ng, A.Y.: Grounded compositional semantics for finding and describing images with sentences. Trans. Assoc. Comput. Linguist. **2**, 207–218 (2014)
26. Tan, H., Bansal, M.: LXMERT: learning cross-modality encoder representations from transformers. arXiv preprint [arXiv:1908.07490](https://arxiv.org/abs/1908.07490) (2019)
27. Vaswani, A., et al.: Attention is all you need. CoRR abs/1706.03762 (2017). <http://arxiv.org/abs/1706.03762>
28. Wang, Y., et al.: Position focused attention network for image-text matching. CoRR abs/1907.09748 (2019). <http://arxiv.org/abs/1907.09748>
29. Wen, K., Gu, X., Cheng, Q.: Learning dual semantic relations with graph attention for image-text matching. IEEE Trans. Circuits Syst. Video Technol. (2020)
30. Zhou, L., Palangi, H., Zhang, L., Hu, H., Corso, J.J., Gao, J.: Unified vision-language pre-training for image captioning and VQA. arXiv preprint [arXiv:1909.11059](https://arxiv.org/abs/1909.11059) (2019)