

Multimodal Convolutional Neural Networks for Matching Image and Sentence

Lin Ma Zhengdong Lu Lifeng Shang Hang Li
Noah's Ark Lab, Huawei Technologies

forest.linma@gmail.com, FLu.Zhengdong, Shang.Lifeng, HangLi.HLG@huawei.com

Abstract

In this paper, we propose multimodal convolutional neural networks (m-CNNs) for matching image and sentence. Our m-CNN provides an end-to-end framework with convolutional architectures to exploit image representation, word composition, and the matching relations between the two modalities. More specifically, it consists of one image CNN encoding the image content and one matching CNN modeling the joint representation of image and sentence. The matching CNN composes different semantic fragments from words and learns the inter-modal relations between image and the composed fragments at different levels, thus fully exploit the matching relations between image and sentence. Experimental results demonstrate that the proposed m-CNNs can effectively capture the information necessary for image and sentence matching. More specifically, our proposed m-CNNs significantly outperform the state-of-the-art approaches for bidirectional image and sentence retrieval on the Flickr8K and Flickr30K datasets.

1. Introduction

Associating image with natural language sentence plays an essential role in many applications. Describing the image with natural sentences is useful for image annotation and captioning [8, 21, 26], while retrieving image with a natural language as query is more natural for image search [13, 16]. The association between image and sentence can be formalized as a multimodal matching problem, where the semantically related image and sentence pairs should be assigned higher matching scores than unrelated ones.

Multimodal matching between image and sentence is complicated, and usually occurs at different levels as shown in Figure 1. The words in the sentence, such as “grass”, “dog”, and “ball”, denote the objects in the image. The phrases describing the objects and their attributes or activities, such as “black and brown dog”, and “small black and brown dog play with a red ball”, correspond to the image areas of their grounding meanings. The

Figure 1. Multimodal matching relations between image and sentence. The words and phrases, such as “grass”, “a red ball”, and “small black and brown dog play with a red ball”, correspond to the image areas of their grounding meanings. The sentence “small black and brown dog play with a red ball in the grass” expresses the meaning of the whole image.

whole sentence “small black and brown dog play with a red ball in the grass”, expressing a complete meaning, associates with the whole image. These matching relations should be all taken into consideration for an accurate multimodal matching between image and sentence. Recently, much research work focuses on modeling the image and sentence matching relation at the specific level, namely the word level [31, 32, 6], phrase level [37, 27], and sentence level [13, 16, 30]. However, to the best of our knowledge, there are no models to fully exploit the matching relations between image and sentence by considering the inter-modal correspondences at all the levels together.

The multimodal matching between image and sentence requires good representations of image and sentence. Recently, deep neural networks have been employed to learn better image and sentence representations. Specifically, convolutional neural networks (CNNs) have shown their powerful abilities on learning of image representation [10, 29, 33, 11] and sentence representation [14, 15, 18]. However, the ability of CNN on multimodal matching, specifically the image and sentence matching problem, has not been studied.

In this paper, we propose a novel multimodal convolutional neural network (m-CNN) framework for the image

and sentence matching problem. Trained on a set of image and sentence pairs, the proposed m-CNNs are able to retrieve and rank images given a natural language sentence as query, and vice versa. Our core contributions are:

1. CNN is first proposed for the image and sentence matching problem. We employ convolutional architectures to encode the image, compose different semantic fragments from the words, and learn the matching relations between the image and the composed fragments.
2. The complicated matching relations between image and sentence are fully captured in our proposed m-CNN by letting image and the composed fragments of the sentence meet and interact at different levels. We validate the effectiveness of m-CNNs on the bidirectional image and sentence retrieval experiments, and demonstrate that m-CNNs can achieve performances superior to the state-of-the-art approaches.

2. Related Work

There is a long thread of work on the association between image and text. Early work usually focuses on modeling the relation between the image and annotating words [6, 9, 31, 32, 35] or phrases [27, 37]. These models cannot effectively capture the complicated matching relations between image and sentence. Recently, the association between image and sentence has been studied for bidirectional image and sentence retrieval [13, 16, 30] and automatic image captioning [3, 5, 17, 19, 20, 23, 24, 34].

For bidirectional image and sentence retrieval, Hodosh *et al.* [13] proposed using kernel canonical correlation analysis (KCCA) to discover the shared feature space between image and sentence. However, the highly non-linear inter-modal relations cannot be effectively exploited with the shallow representations of image and sentence. Recently, researchers seek better representations of image and sentence by using deep architectures. Socher *et al.* [30] propose to employ the semantic dependency-tree recursive neural network (SDT-RNN) to map the sentence and the image into the same semantic space, and measure their association by the distance in that space. The global level matching relations between image and sentence are captured by representing the sentence as a global vector. However, they neglect the local fragments of the sentence and their correspondences to the image. In contrary, Karpathy *et al.* [16] work on a finer level by aligning the fragments of sentence and the regions of image. The local inter-modal correspondences between image and sentence fragments are learned, while the global matching relations are not considered. As illustrated in Figure 1, the image content corresponds to different fragments of sentence from local words to the global sentence. To fully exploit the inter-modal matching relations, we propose m-CNNs to compose different semantic

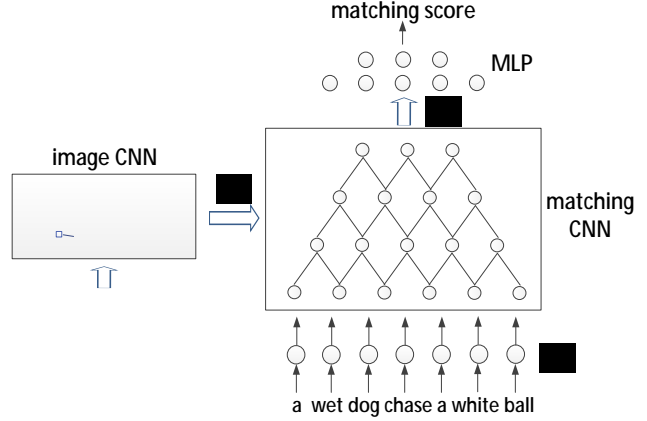


Figure 2. The m-CNN architecture for matching image and sentence. Image representation is generated by the image CNN. The matching CNN composes different fragments from the words of the sentence and learns the joint representation of image and sentence fragments. MLP summarizes the joint representation and produces the matching score.

fragments from the words, let the fragments interact with the image at different levels, and model their matching relations.

For automatic image captioning, researchers employ recurrent visual representation (RVP) [3], multimodal recurrent neural network (m-RNN) [23, 24], multimodal neural language model (MNLN) [19, 20], neural image caption (NIC) [34], deep visual-semantic alignments (DVSA) [17], and long-term recurrent convolution networks (LRCN) [5] to learn the relations between image and sentence and generate a caption for a given image. Please note that those models naturally produce scores for the association between image and sentence (e.g., the likelihood of a sentence as the caption for a given image). It can thus be readily used for the bidirectional image and sentence retrieval.

3. m-CNNs for Matching Image and Sentence

As illustrated in Figure 2, m-CNN takes the image and sentence as input and generates the matching score between them. More specifically, m-CNN consists of three components.

- **Image CNN** The image CNN is used to generate the image representation for matching the fragments composed from words of the sentence, which is computed as follows:

$$im = (w_{im}(CNN_{im}(I)) + b_{im}), \quad (1)$$

where (\cdot) is a nonlinear activation function (e.g., Sigmoid or ReLU [4]). CNN_{im} is an image CNN which takes the image as input and generates a fixed length image representation. The state-of-the-art image CNNs for image recognition, such as [28, 29], can

be used to initialize the image CNN, which returns a 4096-dimensional feature vector from the fully connected layer immediately before the last ReLU layer. The matrix w_{im} is of dimension $d \times 4096$, where d is set as 256 in our experiments. Each image is thus represented as one d -dimension vector im .

- **Matching CNN** The matching CNN takes the encoded image representation im and word representations w_d^i as input and produces the joint representation J_R . As illustrated in Figure 1, the image content may correspond to the sentence fragments with varying scales, which will be adequately captured in the learnt joint representation of image and sentence. Aiming at fully exploiting the inter-modal matching relations, our proposed matching CNNs first compose different semantic fragments from the words and then learn the inter-modal structures and interactions between the image and composed fragments. More specifically, different matching CNNs are designed to make the image interact with the composed fragments of the sentence at different levels to generate the joint representation, from the word and phrase level to the sentence level. Detailed description of the matching CNNs at different levels will be introduced in the following subsections.
- **MLP** Multilayer perceptron (MLP) takes the joint representation J_R as input and produces the final matching score between image and sentence, which is calculated as follows:

$$S_{match} = W_s (W_h(J_R) + b_h) + b_s, \quad (2)$$

where (\cdot) is a nonlinear activation function. w_h and b_h are used to map J_R to the representation in the hidden layer. w_s and b_s are used to compute the matching score between image and sentence.

The three components of our proposed m-CNN are fully coupled in the end-to-end image and sentence matching framework, with all the parameters (e.g., those for image CNN, matching CNN, MLP, w_{im} and b_{im} in Eq. (1), and word representations) being jointly learned under the supervision from matching instances. Threefold benefits are provided. Firstly, the image CNN can be tuned to generate a better image representation for matching. Secondly, word representations can be tuned for further composition and matching processes. Thirdly, the matching CNN (as detailed in the following) composes different fragments from word representations and lets the image representation interact with the fragments at different levels, which can fully exploit the inter-modal matching correspondences between image and sentence. With the nonlinear projection in Eq. (1), the image representations im for different matching CNNs are expected to encode the image content for matching the composed semantic fragments of the sentence.

Figure 3. The word-level matching CNN. (a) The word-level matching CNN architecture. (b) The convolution unit of multimodal convolution layer of MatchCNN_{wd}. The dashed lines indicate the zero padded word and image representations, which are gated out during the convolution process.

3.1. Different Variants of Matching CNN

To fully exploit the matching relations between image and sentence, we let the image representation meet and interact with different composed fragments of the sentence (roughly the word, phrase, and sentence) to generate the joint representation.

3.1.1 Word-level Matching CNN

In order to find the word-level matching relations, we let the image interact with the word-level fragments of sentence and learn their interactions. Moreover, as most convolutional models [1, 14, 22], we consider the convolution unit with a local “receptive field” and shared weights to adequately model the rich structures for word composition and inter-modal interaction. The word-level matching CNN, denoted as MatchCNN_{wd}, is designed as in Figure 3 (a). After sequential layers of convolution and pooling, the joint representation of image and sentence is generated as the input of MLP to produce the matching score.

Convolution Generally, with a sequential input x , the convolution unit for feature map of type- f (among F of them) on the i -th layer is

$$y_{(i,f)} \stackrel{\text{def}}{=} (w_{(i,f)} \cdot x_{(i-1)} + b_{(i,f)}), \quad (3)$$

where $w_{(i,f)}$ are the parameters for the f feature map on i -th layer, (\cdot) is the activation function, and $x_{(i-1)}$ denotes the segment of $(i-1)$ -th layer for the convolution at location i , which is defined as follows:

$$x_{(i-1)} \stackrel{\text{def}}{=} [x_{(i-1)} \quad x_{(i-1)}^{i+1} \quad \cdots \quad x_{(i-1)}^{i+k_{rp}-1}]. \quad (4)$$

k_{rp} defines the size of local “receptive field” for convolution. “ \cdot ” concatenates the neighboring k_{rp} word vectors into a long vector. In this paper, k_{rp} is chosen as 3 for the convolution process.

As MatchCNN_{wd} aims at exploring word-level matching relations, the multimodal convolution layer is introduced by letting the image interact with the word-level fragments of

sentence. The convolution unit of the multimodal convolution layer is illustrated in Figure 3 (b). The input of the multimodal convolution unit is denoted as:

$$\begin{matrix} i \\ (0) \end{matrix} \stackrel{\text{def}}{=} \begin{matrix} i \\ \text{wd} \end{matrix} \quad \begin{matrix} i+1 \\ \text{wd} \end{matrix} \quad \dots \quad \begin{matrix} i+k_{rp}-1 \\ \text{wd} \end{matrix} \quad \begin{matrix} i_m \end{matrix}, \quad (5)$$

where $\begin{matrix} i \\ \text{wd} \end{matrix}$ is the vector representation of word i in the sentence, and i_m is the encoded image feature for matching word-level fragments of sentence. It is not hard to see that the first convolution layer with this input makes the “interaction” between word and image representations, yielding the local matching signal at word level. From the sentence perspective, the multimodal convolution on $\begin{matrix} i \\ (0) \end{matrix}$ composes a higher semantic representation from the words $\begin{matrix} i \\ \text{wd} \end{matrix}, \dots, \begin{matrix} i+k_{rp}-1 \\ \text{wd} \end{matrix}$ in local “receptive field”, such as the phrase “a white ball”. From the matching perspective, the multimodal convolution on $\begin{matrix} i \\ (0) \end{matrix}$ captures the inter-modal correspondence between image representation and the word-level fragments of sentence. The meanings of the word “ball” and the composed phrase “a white ball” are grounded in the image.

Moreover, in order to handle natural sentences of variable lengths, the maximum length of sentence is fixed for MatchCNN_{wd}. Zero vectors are padded for the image and word representations, represented as the dashed lines in Figure 3 (a). The output of the convolution process on zero vectors is gated to be zero. The convolution process in Eq. (3) is further formulated as:

$$\begin{matrix} i \\ (,f) \end{matrix} = g\left(\begin{matrix} i \\ (-1) \end{matrix}\right) \cdot (w_{(,f)}\left(\begin{matrix} i \\ (-1) \end{matrix}\right) + b_{(,f)})$$

where, $g(x) = \begin{matrix} 0, & x == \mathbf{0} \\ 1, & \text{otherwise} \end{matrix} \quad (6)$

The gating function can eliminate the unexpected matching noises composed from the convolution process.

Max-pooling After each convolution layer, a max-pooling layer is followed. Taking a two-unit window max-pooling as an example, the pooled feature is obtained by:

$$\begin{matrix} i \\ (+1,f) \end{matrix} = \max\left(\begin{matrix} 2i \\ (,f) \end{matrix}, \begin{matrix} 2i+1 \\ (,f) \end{matrix}\right). \quad (7)$$

The effects of max-pooling are two-fold. 1) With the stride as two, the max-pooling process lowers the dimensionality of the representation by half, thus making it possible to quickly generate the final joint representation of the image and sentence. 2) It helps filter out the undesired interactions between image and fragments of sentence. Taking the sentence in Figure 3 (a) as an example, the composed phrase “dog chase a” matches better to the image than “chase a white”. Therefore, we can imagine that a well-trained multimodal convolution unit will generate a better matching representation of “dog chase a” and image. The max-pooling process will pool the matching representation out for further convolution and pooling processes.

Figure 4. The phrase-level matching CNN and composed phrases. (a) The short phrase is composed by one layer of convolution and pooling. (b) The long phrase is composed by two sequential layers of convolution and pooling. (c) The phrase-level matching CNN architecture.

The convolution and pooling processes summarize the local matching signals explored at the word level. More layers of convolution and pooling can be further employed to conduct matching decisions at higher levels and finally reach a global joint representation. Specifically, in this paper another two more layers of convolution and max-pooling are employed to capture the inter-modal correspondences between image and word-level fragments of the sentence.

3.1.2 Phrase-level Matching CNN

Different from the word-level matching CNN, we let CNN work solely on words to certain levels before interacting with the image. Without seeing the image feature, the convolution process composes higher semantic representations from the words in the “receptive field”, while the max-pooling process filters out the undesired compositions. These composed representations are roughly correspond to phrases from the language perspective. We let the image interact with the composed phrases to reason their inter-modal matching relations.

As illustrated in Figure 4 (a), after one layer of convolution and max-pooling, short phrases (denoted as $\begin{matrix} i \\ (2) \end{matrix}$) are composed from four words, such as “a woman in jeans”. These composed short phrases offer richer and more detailed descriptions about the objects and their relationships in the image, compared with single words, such as “woman” and “jeans”. With an additional layer of convolution and max-pooling on short phrases, long phrases (denoted as $\begin{matrix} i \\ (4) \end{matrix}$) are composed from four short phrases (also from ten words), such as “a black dog be in the

Figure 5. The sentence-level matching CNN. The joint representation is obtained by concatenating the image and sentence representations together.

grass with a woman” in Figure 4 (b). Compared with the short phrases and single words, the long phrases present even richer and much more detailed descriptions about the objects, their activities, and their relative positions.

In order to reason the inter-modal relations between image and the composed phrases, a multimodal convolution layer is introduced by performing the convolution on the image and phrase representations. The input of the multimodal convolution unit is

$$i_{ph} \stackrel{\text{def}}{=} i_{ph} \quad i_{ph+1} \quad \dots \quad i_{ph+k_{rp}-1} \quad im, \quad (8)$$

where i_{ph} is the composed phrase representation, which can be either short phrases $i_{(2)}$ or long phrases $i_{(4)}$. The multimodal convolution process produces the phrase-level matching decisions. Then the layers after that (namely the max-pooling and convolution layers) can be viewed as further fusion of these local phrase-level matching decisions to the joint representation, which captures the local matching relations between image and composed phrase fragments. Specifically, for short phrases, two sequential layers of convolution and pooling are followed to generate the joint representation. We name the matching CNN for short phrases and image as MatchCNN_{phs}. For long phrases, only one sequential layer of convolution and pooling is used to summarize the local matching to the joint representation. The matching CNN for long phrases and image is named as MatchCNN_{phi}.

3.1.3 Sentence-level Matching CNN

The sentence-level matching CNN, denoted as MatchCNN_{st}, goes one step further in the composition and defers the matching until the sentence is fully represented, as illustrated in Figure 5. More specifically, one image CNN encodes the image into a feature vector. One sentence CNN, consisting of three sequential layers of convolution and pooling, represents the whole sentence as a feature vector. The multimodal layer concatenates the image and sentence representations together as their joint representation

$$JR = im \quad st, \quad (9)$$

Table 1. Configurations of MatchCNN_{wd}, MatchCNN_{phs}, MatchCNN_{phi}, and MatchCNN_{st} in each column. (conv denotes the convolution layer; multi-conv denotes the multimodal convolution layer; max denotes the max pooling layer.)

MatchCNN _{wd}	MatchCNN _{phs}	MatchCNN _{phi}	MatchCNN _{st}
+ im			
multi-conv-200	conv-200	conv-200	conv-200
max-2	max-2	max-2	max-2
	+ im		
conv-300	multi-conv-300	conv-300	conv-300
max-2	max-2	max-2	max-2
		+ im	
conv-300	conv-300	multi-conv-300	conv-300
max-2	max-2	max-2	max-2
			+ im

where st denotes the sentence representation by vectorizing the features in the last layer of the sentence CNN.

For the sentence “a little boy in a bright green field have kick a soccer ball very high in the air” illustrated in Figure 5, although word-level and phrase-level fragments, such as “boy”, “kick a soccer ball”, correspond to the objects as well as their activities in the image, the whole sentence needs to be fully represented to make an accurate association with the image. The sentence CNN with layers of convolution and pooling is used to encode the whole sentence as a feature vector representing its semantic meaning. Concatenating the image and sentence representation together, MatchCNN_{st} does not conduct matching, but transfer the representations of the two modalities to the later MLP for fusing and matching.

3.2 m-CNNs with Different Matching CNNs

We can get different m-CNNs with different variants of Matching CNNs, namely m-CNN_{wd}, m-CNN_{phs}, m-CNN_{phi}, and m-CNN_{st}. To fully exploit the inter-modal matching relations between image and sentence at different levels, we use an ensemble m-CNN_{ENS} of the four variants by summing the matching scores generated from the four m-CNNs together.

4. Implementation Details

In this section, we describe the detailed configurations of our proposed m-CNN models and how we train the proposed networks.

4.1. Configurations

We use two different image CNNs, OverFeat [28] (the “fast” network) and VGG [29] (with 19 weight layers), with which we take not only the architectures but also the original parameters (learnt on the ImageNet dataset) for initialization. By chopping the softmax layer and last ReLU layer,

Table 2. Bidirectional image and sentence retrieval results on Flickr8K.

	Sentence Retrieval				Image Retrieval			
	R@1	R@5	R@10	Med r	R@1	R@5	R@10	Med r
Random Ranking	0.1	0.6	1.1	631	0.1	0.5	1.0	500
DeViSE [6]	4.8	16.5	27.3	28.0	5.9	20.1	29.6	29
SDT-RNN [30]	6.0	22.7	34.0	23.0	6.6	21.6	31.7	25
MNLM [20]	13.5	36.2	45.7	13	10.4	31.0	43.7	14
MNLM-VGG [20]	18.0	40.9	55.0	8	12.5	37.0	51.5	10
m-RNN [24]	14.5	37.2	48.5	11	11.5	31.0	42.4	15
Deep Fragment [16]	12.6	32.9	44.0	14	9.7	29.6	42.5	15
RVP (T) [3]	11.6	33.8	47.3	11.5	11.4	31.8	45.8	12.5
RVP (T+I) [3]	11.7	34.8	48.6	11.2	11.4	32.0	46.2	11
DVSA (DepTree) [17]	14.8	37.9	50.0	9.4	11.6	31.4	43.8	13.2
DVSA (BRNN) [17]	16.5	40.6	54.2	7.6	11.8	32.1	44.7	12.4
NIC [34]	20.0	*	61.0	6	19.0	*	64.0	5
OverFeat [28]:								
m-CNN _{wd}	8.6	26.8	38.8	18.5	8.1	24.7	36.1	20
m-CNN _{phs}	10.5	29.4	41.7	15	9.3	27.9	39.6	17
m-CNN _{phi}	10.7	26.5	38.7	18	8.1	26.6	37.8	18
m-CNN _{st}	10.6	32.5	43.6	14	8.5	27.0	39.1	18
m-CNN _{ENS}	14.9	35.9	49.0	11	11.8	34.5	48.0	11
VGG [29]:								
m-CNN _{wd}	15.6	40.1	55.7	8	14.5	38.2	52.6	9
m-CNN _{phs}	18.0	43.5	57.2	8	14.6	39.5	53.8	9
m-CNN _{phi}	16.7	43.0	56.7	7	14.4	38.6	52.2	9
m-CNN _{st}	18.1	44.1	57.9	7	14.6	38.5	53.5	9
m-CNN _{ENS}	24.8	53.7	67.1	5	20.3	47.6	61.7	5

the output of the last fully-connected layer is deemed as the image representation, denoted as $\text{CNN}_{\text{im}}(\mathbf{I})$ in Eq. (1).

The configurations of $\text{MatchCNN}_{\text{wd}}$, $\text{MatchCNN}_{\text{phs}}$, $\text{MatchCNN}_{\text{phi}}$, and $\text{MatchCNN}_{\text{st}}$ are outlined in Table 1. We use three convolution layers, three max pooling layers, and an MLP with two fully connected layers for all these four convolutional networks. The first convolution layer of $\text{MatchCNN}_{\text{wd}}$, second convolution layer of $\text{MatchCNN}_{\text{phs}}$, and third convolution layer of $\text{MatchCNN}_{\text{phi}}$ are the multi-modal convolution layers, which blend the image representation and fragments of the sentence together to compose a higher level semantic representation. The $\text{MatchCNN}_{\text{st}}$ concatenates the image and sentence representations together and leaves the interactions to the final MLP. The matching CNNs are designed on fixed architectures, which need to be set to accommodate the maximum length of the input sentences. During our experiments, the maximum length is set as 30. The word representations are initialized by the skip-gram model [25] with the dimension as 50. The joint representation obtained from the matching CNN is fed into the MLP with one hidden layer with the size as 400.

4.2. Learning

The m-CNNs can be trained with contrastive sampling using a ranking loss function. More specifically, for the score function $S_{\text{match}}(\cdot)$ as in Eq. (2), the objective function is defined as:

$$\begin{aligned} e(\mathbf{x}_n, \mathbf{y}_n, \mathbf{y}_m) = \\ \max(0, \mu - S_{\text{match}}(\mathbf{x}_n, \mathbf{y}_n) + S_{\text{match}}(\mathbf{x}_n, \mathbf{y}_m)) \end{aligned} \quad (10)$$

where μ denotes the parameters, $(\mathbf{x}_n, \mathbf{y}_n)$ denotes the related image-sentence pair, and $(\mathbf{x}_n, \mathbf{y}_m)$ is the randomly sampled unrelated image-sentence pair with $n = m$. The meanings of \mathbf{x} and \mathbf{y} vary depending on the matching task. For image retrieval from sentence, \mathbf{x} denotes the natural language sentence and \mathbf{y} denotes the image; for sentence retrieval from image, it is just the opposite. The objective is to force the matching score of the related pair $(\mathbf{x}_n, \mathbf{y}_n)$ to be greater than the unrelated pair $(\mathbf{x}_n, \mathbf{y}_m)$ by a margin μ , which is simply set as 0.5 during the training process.

We use the stochastic gradient descent (SGD) with mini-batches of 100–150 for optimization. In order to avoid overfitting, early-stopping [2] and dropout (with probability 0.1) [12] are used. ReLU is used as the activation function throughout the m-CNNs.

5. Experiments

In this section, we evaluate the effectiveness of our m-CNNs on bidirectional image and sentence retrieval. We begin by describing the datasets used for evaluation, followed by a brief description of competitor models. As our m-CNNs are bidirectional, we evaluate the performances on both image retrieval and sentence retrieval.

5.1. Datasets

We test our matching models on the public image-sentence datasets, with varying sizes and characteristics.

Flickr8K [13] This dataset consists of 8,000 images col-

Table 3. Bidirectional image and sentence retrieval results on Flickr30K.

	Sentence Retrieval				Image Retrieval			
	R@1	R@5	R@10	Med r	R@1	R@5	R@10	Med r
Random Ranking	0.1	0.6	1.1	631	0.1	0.5	1.0	500
DeViSE [6]	4.5	18.1	29.2	26	6.7	21.9	32.7	25
SDT-RNN [30]	9.6	29.8	41.1	16	8.9	29.8	41.1	16
MNLM [20]	14.8	39.2	50.9	10	11.8	34.0	46.3	13
MNLM-VGG [20]	23.0	50.7	62.9	5	16.8	42.0	56.5	8
m-RNN [24]	18.4	40.2	50.9	10	12.6	31.2	41.5	16
m-RNN-VGG [23]	35.4	63.8	73.7	3	22.8	50.7	63.1	5
Deep Fragment [16]	14.2	37.7	51.3	10	10.2	30.8	44.2	14
RVP (T) [3]	11.9	25.0	47.7	12	12.8	32.9	44.5	13
RVP (T+I) [3]	12.1	27.8	47.8	11	12.7	33.1	44.9	12.5
DVSA (DepTree) [17]	20.0	46.6	59.4	5.4	15.0	36.5	48.2	10.4
DVSA (BRNN) [17]	22.2	48.2	61.4	4.8	15.2	37.7	50.5	9.2
NIC [34]	17.0	*	56.0	7	17.0	*	57.0	7
LRCN [5]	*	*	*	*	17.5	40.3	50.8	9
OverFeat [28]:								
m-CNN _{wd}	12.7	30.2	44.5	14	11.6	32.1	44.2	14
m-CNN _{phs}	14.4	38.6	49.6	11	12.4	33.3	44.7	14
m-CNN _{phl}	13.8	38.1	48.5	11.5	11.6	32.7	44.1	14
m-CNN _{st}	14.8	37.9	49.8	11	12.5	32.8	44.2	14
m-CNN _{ENS}	20.1	44.2	56.3	8	15.9	40.3	51.9	9.5
VGG [29]:								
m-CNN _{wd}	21.3	53.2	66.1	5	18.2	47.2	60.9	6
m-CNN _{phs}	25.0	54.8	66.8	4.5	19.7	48.2	62.2	6
m-CNN _{phl}	23.9	54.2	66.0	5	19.4	49.3	62.4	6
m-CNN _{st}	27.0	56.4	70.1	4	19.7	48.4	62.3	6
m-CNN _{ENS}	33.6	64.1	74.9	3	26.2	56.3	69.6	4

lected from Flickr. Each image is accompanied with 5 sentences describing the image content. This dataset provides the standard training, validation, and testing split.

Flickr30K [36] This dataset consists of 31,783 images collected from Flickr. Each image is also accompanied with 5 sentences describing the content of the image. Most of the images depict varying human activities. We use the public split in [24] for training, validation, and testing.

5.2. Competitor Models

We compare our models with recently developed models on the performances of the bidirectional image and sentence retrieval, specifically DeVISE [6], SDT-RNN [30], Deep Fragment [16], m-RNN [23, 24], MNLM [20], RVP [3], DVSA [17], NIC [34], and LRCN [5]. DeVISE, Deep Fragment, and SDT-RNN are regarded as working on the word-level, phrase-level, and sentence-level respectively, which all embed the image and sentence into the same semantic space. The other models, namely MNLM, m-RNN, RVP, DVSA, NIC, and LRCN, which are originally proposed for automatic image captioning, can also be used for retrieval in both directions.

5.3. Experimental Results and Analysis

5.3.1 Bidirectional Image and Sentence Retrieval

We adopt the evaluation metrics [16] for comparison. More specifically, for bidirectional retrieval, we report the median

rank (Med r) of the closest ground truth result in the list, as well as the R@K (with $K = 1, 5, 10$) which computes the fraction of times the correct result is found among the top K results. The performances of the proposed m-CNNs for bidirectional image and sentence retrieval on Flickr8K and Flickr30K are provided in Table 2 and 3, respectively. We highlight the best performance of each evaluation metric. In most cases, m-CNN_{ENS} (with VGG) outperforms all the competitor models.

On Flickr8K, only NIC slightly outperforms m-CNN_{ENS} (with VGG) on the image retrieval task in terms of R@10. One possible reason is that NIC uses a better image CNN [33], compared with VGG. As discussed in Section 5.3.3, the performance of image CNN greatly affects the performance of the bidirectional image and sentence retrieval. Another possible reason is the lack of training samples. Flickr8K consists of only 8,000 images, which are insufficient for adequately tuning the parameters of the convolutional architectures in m-CNNs.

On Flickr30K, with more training instances (30,000 images), the best performing competitor model becomes m-RNN-VGG on both tasks, while NIC only achieves moderate retrieval accuracies. Only m-RNN-VGG outperforms m-CNN_{ENS} (with VGG) on the sentence retrieval task in terms of R@1. When it comes to image retrieval, m-CNN_{ENS} (with VGG) is consistently better than m-RNN-VGG. One possible reason may be that m-RNN-VGG is

Table 4. The matching scores of the image and sentence. The natural language sentence (in bold) is the true caption of the image, while the other three sentences are generated by random reshuffle of words.

image	sentence	m-CNN _{wd}	m-CNN _{phs}	m-CNN _{phl}	m-CNN _{st}
	three person sit at an outdoor table in front of a building paint like the union jack .	-0.87	1.91	-1.84	2.93
	like union at in sit three jack the person a paint building table outdoor of front an .	-1.49	1.66	-3.00	2.37
	sit union a jack three like in of paint the person table outdoor building front at an .	-2.44	1.55	-3.90	2.53
	table sit three paint at a building of like the an person front outdoor jack union in .	-1.93	1.64	-3.81	2.52

designed for the caption generation and is particularly good at finding the suitable sentence for a given image.

5.3.2 Performances of Different m-CNNs

The proposed m-CNN_{wd} and DeViSE [6] both aim at exploiting word-level inter-modal correspondences between the image and sentence. However, DeViSE treats each word equally and averages their word vectors as the representation of the sentence, while our m-CNN_{wd} lets image interact with each word to compose higher semantic representations, which significantly outperforms DeViSE. On the other end, both SDT-RNN [30] and the proposed m-CNN_{st} exploit the matching relations between the image and sentence at the sentence level. However, SDT-RNN encodes each sentence recursively into a feature vector based on a pre-defined dependency tree, while m-CNN_{st} works on a more flexible manner with a sliding window on the sentence to finally generate the sentence representation. Therefore, a better performance is obtained by m-CNN_{st}.

Deep Fragment [16] and the proposed m-CNN_{phs} and m-CNN_{phl} match the image and sentence fragments at the phrase level. However, Deep Fragment uses the edges of the dependency tree to model the sentence fragments, making it impossible to describe more complex relations within the sentence. For example, Deep Fragment parses a relative complex phrase “black and brown dog” to two relations “(CONJ, black, brown)” and “(AMOD, brown, dog)”, while m-CNN_{phs} handles the same phrase as a whole to compose a higher semantic representation. Moreover, m-CNN_{phl} can readily handle longer phrases and reason their grounding meanings in the image. Consequently, better performances of m-CNN_{phs} and m-CNN_{phl} (with VGG) are obtained compared with Deep Fragment.

Moreover, it can be observed that m-CNN_{st} consistently outperforms the other m-CNNs. In m-CNN_{st}, the sentence CNN can effectively summarize the sentence and make a better sentence-level association with image. The other m-CNNs can capture the matching relations at the word and phrase levels. The matching relations should be considered together to fully depict the inter-modal correspondences between the image and sentence. Thus m-CNN_{ENS} achieves the best performances, which indicates that m-CNNs at different levels are complementary to each other to capture the complicated image and sentence matching relations.

5.3.3 Influence of Image CNN

We use OverFeat and VGG to initialize the image CNN in m-CNN for the retrieval tasks. It can be observed that m-CNNs with VGG significantly outperform that with OverFeat by a large margin, which is consistent with their performance on image classification on ImageNet (14% and 7% top-5 classification errors for OverFeat and VGG, respectively). Clearly the retrieval performance depends heavily on the efficacy of the image CNN, which might explain the good performance of NIC on Flickr8K. Moreover, the so-called region with CNN features in [7] can be used to encode image regions, which are adopted as the image fragments in Deep Fragment and DVSA. In the future, we will consider to incorporate these image CNNs into our m-CNNs to make more accurate inter-modal matching.

5.3.4 Composition Abilities of m-CNNs

m-CNNs can compose different semantic fragments from words of the sentence for the inter-modal matching at different levels, and therefore possess the ability of word composition. More specifically, we want to check whether the m-CNNs can compose reliable semantic fragments from words of random orders for matching the image. As demonstrated in Table 4, the matching scores between an image and its accompanied sentence (from different m-CNNs) greatly drop after the random reshuffle of words. It is a fairly strong evidence that m-CNNs will compose highly semantic representations from words of natural language sentence and thus make the inter-modal matching relations between image and sentence.

6. Conclusion

We have proposed multimodal convolutional neural networks (m-CNNs) for matching image and sentence. The proposed m-CNNs rely on convolutional architectures to compose different semantic fragments of the sentence and learn the interactions between image and the composed fragments at different levels, and therefore can fully exploit the inter-modal matching relations. Experimental results on bidirectional image and sentence retrieval tasks demonstrate the consistent improvements of m-CNNs over the state-of-the-art approaches.

Acknowledgement: The work is partially supported by China National 973 project 2014CB340301.

References

- [1] O. Abdel Hamid, A. R. Mohamed, H. Jiang, and G. Penn. Applying convolutional neural networks concepts to hybrid nn-hmm model for speech recognition. *ICASSP*, 2012. **3**
- [2] R. Caruana, S. Lawrence, and C. L. Giles. Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. *NIPS*, 2000. **6**
- [3] X. Chen and C. L. Zitnick. Learning a recurrent visual representation for image caption generation. *arXiv:1411.5654*, 2014. **2, 6, 7**
- [4] G. E. Dahl, T. N. Sainath, and G. E. Hinton. Improving deep neural networks for lvcsr using rectified linear units and dropout. *ICASSP*, 2013. **2**
- [5] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darreell. Long-term recurrent convolutional networks for visual recognition and description. *arXiv:1411.4389*, 2014. **2, 7**
- [6] A. Frame, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. A. Ranzato, and T. Mikolov. Devise: A deep visual-semantic embedding model. *NIPS*, 2013. **1, 2, 6, 7, 8**
- [7] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *CVPR*, 2014. **8**
- [8] Y. Gong, L. Wang, M. Hodosh, J. Hockenmaier, and S. Iazebnik. Improving image-sentence embeddings using large weakly annotated photo collections. *ECCV*, 2014. **1**
- [9] D. Grangier and S. Bengio. A neural network to retrieve images from text queries. *ICANN*, 2006. **2**
- [10] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *ECCV*, 2014. **1**
- [11] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: surpassing human-level performance on imagenet classification. *arXiv:1502.01852*, 2015. **1**
- [12] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv:1207.0580*, 2012. **6**
- [13] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47:853–899, 2013. **1, 2, 6**
- [14] B. Hu, Z. Lu, H. Li, and Q. Chen. Convolutional neural network architectures for matching natural language sentences. *NIPS*, 2014. **1, 3**
- [15] N. Kalchbrenner, E. Grefenstette, and P. Blunsom. A convolutional neural network for modelling sentences. *ACL*, 2014. **1**
- [16] A. Karpathy, A. Joulin, and F.-F. Li. Deep fragment embeddings for bidirectional image sentence mapping. *NIPS*, 2014. **1, 2, 6, 7, 8**
- [17] A. Karpathy and F.-F. Li. Deep visual-semantic alignments for generating image descriptions. *arXiv:1412.2306*, 2014. **2, 6, 7**
- [18] Y. Kim. Convolutional neural network for sentence classification. *EMNLP*, 2014. **1**
- [19] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Multimodal neural language model. *ICML*, 2014. **2**
- [20] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv:1411.2539*, 2014. **2, 6, 7**
- [21] R. Kiros and C. Szepesvári. Deep representations and codes for image auto-annotation. *NIPS*, 2012. **1**
- [22] Y. LeCun and Y. Bengio. Convolutional networks for images, speech and time series. *The Handbook of Brain Theory and Neural Networks*, 3361, 1995. **3**
- [23] J. Mao, W. Xu, Y. Yang, J. Wang, and A. L. Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). *arXiv:1412.6632*, 2014. **2, 7**
- [24] J. Mao, W. Xu, Y. Yang, J. Wang, and A. L. Yuille. Explain images with multimodal recurrent neural networks. *arXiv:1410.1090*, 2014. **2, 6, 7**
- [25] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv:1301.3781*, 2013. **6**
- [26] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2txt: Describing images using 1 million captioned photographs. *NIPS*, 2011. **1**
- [27] M. A. Sadeghi and A. Farhadi. Recognition using visual phrases. *CVPR*, 2011. **1, 2**
- [28] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. Lecun. Overfeat: Intergrated recognition, localization and detection using convolutional networks. *arXiv:1312.6229*, 2014. **2, 5, 6, 7**
- [29] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014. **1, 2, 5, 6, 7**
- [30] R. Socher, Q. V. L. A. Karpathy, C. D. Manning, and A. Y. Ng. Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, 2:207–218, 2014. **1, 2, 6, 7, 8**
- [31] N. Srivastava and R. Salakhutdinov. Learning representations for multimodal data with deep belief nets. *ICML Representation Learning Workshop*, 2012. **1, 2**
- [32] N. Srivastava and R. Salakhutdinov. Multimodal learning with deep boltzmann machines. *NIPS*, 2012. **1, 2**
- [33] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv:1409.4842*, 2014. **1, 7**
- [34] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: a neural image caption generator. *arXiv:1411.4556*, 2014. **2, 6, 7**
- [35] J. Weston, S. Bengio, and N. Usunier. Wsabie: Scaling up to large vocabulary image annotation. *IJCAI*, 2011. **2**
- [36] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: new similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2014. **7**
- [37] C. L. Zitnick, D. Parikh, and L. Vanderwende. Learning the visual interpretation of sentences. *ICCV*, 2013. **1, 2**