Deep learning for visual understanding

Guo, Y.; Guo Y.

Chapter 6

What Convnets Make for Image Captioning?

Nowadays, a general pipeline for the image captioning task takes advantage of image representation based on convolutional neural networks (CNNs) and sequence modeling based on recurrent neural networks (RNNs). Captioning performance closely depends on the discriminative capacity of CNNs. Our work aims to investigate the effects of different Convnets (CNN models) on image captioning. We train three Convnets based on different classification tasks: single-label, multi-label and multi-attribute, and feed the image features from these Convnets into a Long Short-Term Memory (LSTM) to model the sequence of words. Since the three Convnets focus on different visual contents in one image, we propose aggregating them together to generate a richer visual representation. Furthermore, during testing, we use an efficient multi-scale augmentation approach based on fully convolutional networks (FCNs). Extensive experiments on MS COCO 2014 dataset provide significant insights into the effects of Convnets. Moreover, we achieve comparable results to the state-of-the-art for both caption generation and image-sentence retrieval tasks.

6.1 Introduction

Image captioning is a fundamental and important task in vision-to-language research. It aims to describe an image with meaningful and sensible sentence-level captions. The automatically generated descriptions should cover the salient content in images, including objects, actions and other relations. In early research of image captioning, it has been converted to a retrieval-based task. Those retrieval-based approaches [297–299] focus on mapping images to sentences based on predefined captions. However, they fail to generate novel sentences for unseen scenes. To address this issue, generative approaches are developed to estimate novel sentences, such as Midge [300] and Baby Talk [301].

Recently, a new paradigm for image captioning is proposed in many state-of-the-art approaches [266, 267, 302–304]. This paradigm mainly integrates a convolutional neural network (CNN) and a recurrent neural network (RNN) together. The CNN is used to capture high-level image features, while the RNN generates a sequence of words based on the image features. In particular, a rich visual representation contributes much to generating accurate image captions. However, some Convnets (CNN models) are originally trained for image classification, but not for image captioning. It thus raises an important question: *What Convnets make for image captioning?*

Our aim in this work is to fully investigate the effects of different Convnets on image captioning. We exploit three kinds of Convnets: single-label Convenet, multi-label Convnet, and multi-attribute Convnet. (1) A single-label Convnet indicates a CNN model pre-trained on ImageNet dataset [21], such as AlexNet [14] and VGG-16 [24]. This Convnet can often offer one generic image representation. (2) A multi-label Convnet can predict multiple class labels given one image. It is consistent with the observation that sentence-level captions often talk about many salient objects jointly in images. Therefore, we fine-tune a multi-label Convnet on MS COCO 2014 [305] that consists of 80 object categories. Each image is annotated with multiple object labels. (3) A multi-attribute Convnet can not only reflect multiple object classes, but also describe actions and other relations about objects, for example jumping, sitting and interacting. Therefore,

a multi-attribute Convnet is able to narrow the gap between vision and language. We fine-tune a multi-attribute Convnet based on 300 attributes derived from MS COCO captions [305].

By observing the feature maps learned in the three Convnets, we find that their maps focus on different visual fields in images. Therefore, we propose aggregating their features together to generate a richer representation.

In addition, during the test stage, we take advantage of the efficient fully convolutional networks (FCNs) [60] for multi-scale augmentation. We use two scales of FCNs that are interpreted from one pre-trained CNN. This augmentation approach can be applied to both the single Convnet and multi-Convnet aggregation. Finally, we employ the Long Short-Term Memory (LSTM) [277] to build the language model. Figure 6.1 shows an image example from MS COCO 2014 [305]. Note that the visual feature is fed to the LSTM unit at each time step.

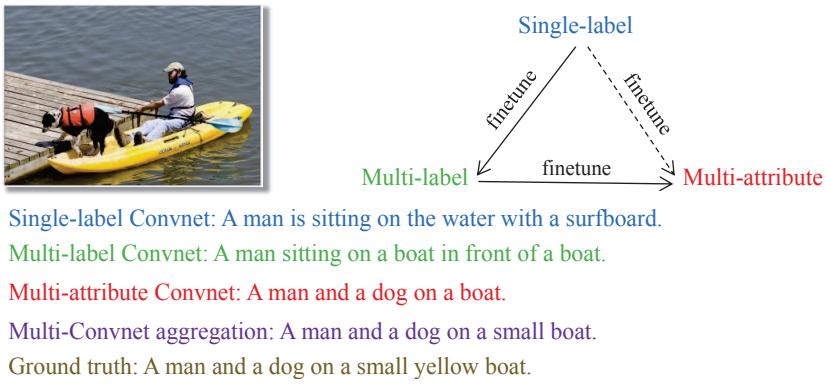

Figure 6.1: Example of image captioning using different Convnets. Each Convnet shows meaningful description. As compared to the human-written ground-truth, the multi-Convnet can generate closer result than any single Convnet.

In a nutshell, **our contributions** can be summarized as follows:

- We present a full comparison among the three Convnets for the image captioning task. Furthermore, we study the benefits of each Convnet and then integrate multiple Convnets for a richer visual representation. Our work can provide promising insights into deeply diagnosing and understanding Convnets for vision-to-language tasks.

- We employ an efficient multi-scale augmentation approach using FCNs.
- We achieve comparable results to the state-of-the-art on MS COCO 2014 dataset, both for caption generation and image-sentence retrieval tasks.

6.2 Related Work

In this section, we summarize related image captioning approaches based on CNN-RNN as below.

A prior work in NIC [267] employed a CNN-RNN scheme to model the image captioning problem. CNNs are used as the "encoder" to visually represent the input image with a fixed-length feature vector. Then RNNs, as the "decoder", can translate the feature vector into sentence-level captions. Similarly, other similar approaches [266, 304, 306] followed this CNN-RNN paradigm. Instead of only using CNN features, Jia et al. [307] added extra semantic information to each unit of the LSTM block. Jin et al. [308] integrated scene-specific contexts in order to highlight higher-level semantic information in images. In addition, Xu et al. [303] introduced a visual attention based model inspired by human visual system. The attention mechanism can automatically learn latent alignments between regions and words. Apart from the whole image captioning, there were some works focusing on image regions based captioning [302, 309, 310]. They first localized salient regions in images and then described them with natural language.

Recent work in [268] began capturing attributes to represent visual content. Notably, Yao et al. [311] investigated the performance upper bounds based on attributes for image and video captioning. However, both of these works did not train a new CNN model based on attributes. The most similar work in [312] fine-tuned a CNN based on the task of image-attribute classification. In comparison, our work had several main differences from [312]:

First, we intended to add a multi-label Convnet as a bridge from a single-label to a multi-attribute Convnet (see the two solid lines in Figure 6.1). Thus our multi-attribute Convnet had two-stage fine-tuning. In contrast, [312] directly fine-tuned a multi-attribute Convnet from a single-label Convnet (see the dash line

in Figure 6.1), and failed to study the effects of a multi-label Convnet. Second, we further evaluated the aggregation of multiple Convnets that has not been studied previously in [312]. Third, we presented an efficient multi-scale testing approach as compared to using expensive region proposals in [312]. In addition, their testing step was not end-to-end.

6.3 Proposed Approach

In this section, we will present our image captioning system in three aspects. First, we show the usage of single Convnet for capturing visual representation. Second, we find that integrating image features from the three Convnets is beneficial for a richer representation. Third, at the test stage, we use a multi-scale testing approach based on FCNs.

6.3.1 Convnets for Image Captioning

This part introduces the training details about the three Convnets. Notably, the multi-attribute Convnet also belongs to a multi-label classification task, but it has different training from the multi-label Convnet.

Single-label Convnet. CNNs trained on ImageNet dataset [21] are widely used as off-the-shelf feature extractors, such as Alexnet [14] and VGG-16 [24]. We call these CNNs as single-label Convnets, since they are originally trained for single-label classification, for example 1000 classes in ImageNet 2012. Here we use the VGG-16 net as a single-label Convnet for our image captioning system. As the left part in Figure 6.2, an image from MS COCO [305] is fed to a single-class Convnet that outputs a 1000-Dim visual feature.

Multi-label Convnet. Image captions often focus on multiple objects in images, instead of mentioning only one salient object. We thus train a multi-label Convnet on MS COCO 2014 dataset [305] that consists of 80 object categories. Each image in MS COCO is annotated by about 3 object labels on average. Instead of training from scratch, we fine-tune the single-label Convnet for a multi-label recognition

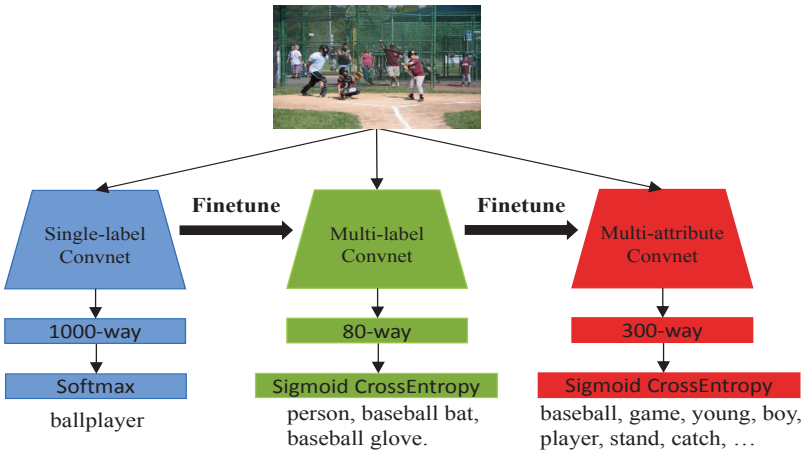

Figure 6.2: Illustration of the three Convnets for visual representations. The multi-label Convnet is fine-tuned from the pre-trained single-label Convnet. The multi-attribute Convnet performs two-stage fine-tuning.

task. Note that we replace the original 1000-way layer with 80-way layer. We use a sigmoid cross-entropy function to compute the element-wise loss. Assume that there are K classes (e.g. 80), the total cost sums up K of sigmoid losses by

$$l_1(x) = -\sum_{k=1}^{K} y_k(x) \log p_k(x) + (1 - y_k(x)) \log(1 - p_k(x)), \tag{6.1}$$

where $y_k \in \{0, 1\}$ is the ground-truth label indicating the absence or presence of the category k in the input image x. $P_k(x)$ indicates the prediction probability of containing the category k. During fine-tuning, the parameters of the last fully-connected layer (i.e. the multi-class prediction layer) are initialized with gaussian filters. We initialize the learning rate of the last fully-connected layer with 0.01. Instead, the learning rates of other convolutional layers and fully-connected layers (i.e. fc6 and fc7) are initialized with 0.0001 and 0.001, respectively. The learning rate is divided by 10 after 2×10^4 iterations. The whole training will be terminated after 5×10^4 iterations. Besides, we use a weight decay of 0.0001, a momentum of 0.9, and a mini-batch size of 100. The multi-label Convnet is shown in the middle part in Figure 6.2.

Multi-attribute Convnet. Apart from object categories, a descriptive caption should mention more information like actions (e.g. sit, run) and other relations

(e.g. blue, small). Hence, using a Convnet that can reflect more attributes is beneficial for narrowing the gap between visual features and language words. Based on a multi-label Convnet, we further fine-tune a multi-attribute Convnet. First, we build an attribute dictionary based on MS COCO captions dataset. In [311], they summarize three groups of atoms: entity, action and attribute. We select top-100 atoms from each group, therefore, the attribute dictionary consists of 300 words (or attributes) in total. Note that the atoms defined in [311] are renamed as attributes in our work. Then, we remake the topmost layer with a 300-way fully-connected layer, as shown in the right part in Figure 6.2. Assume that G denotes the number of attributes (e.g. $G = 300$). Similarly, the sigmoid cross-entropy loss is computed by

$$l_2(x) = -\sum_{g=1}^{G} y_g(x) \log p_g(x) + (1 - y_g(x)) \log(1 - p_g(x)), \tag{6.2}$$

where $y_g \in \{0, 1\}$ is the ground truth; $P_g(x)$ is the prediction probability. Since each image in MS COCO has five human-written captions, we merge five captions together to generate the ground-truth. During fine-tuning the multi-attribute model, we use the same hyper-parameters as the multi-label training.

To compare the visual features from the three Convnets, we visualize their most activated feature maps learned in the fifth convolutional layer (i.e. conv5_3), as illustrated in Figure 6.3. Here, we regard the feature map which has the largest average activation value as the most activated feature map. It can be seen that the three Convnets focus on different visual fields in images. This offers clear insights into diverse characteristics of the three Convnets.

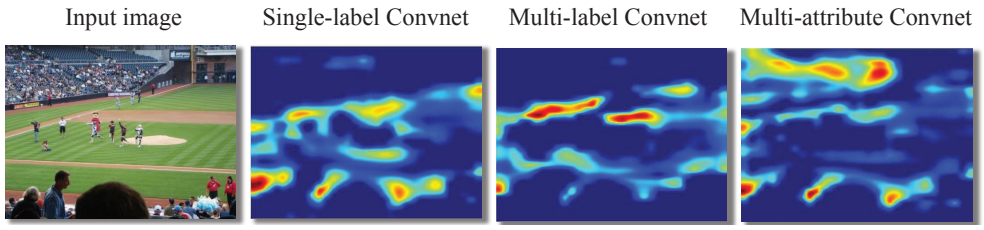

Figure 6.3: Visualization of feature maps for the three Convnets. We select the most activated feature map in conv5_3. We can see that the three Convnets focus on different visual fields in images due to their different classification objectives.

6.3.2 Multi-Convnet Aggregation

Since the three Convnets are trained for different classification objectives and can represent different features given the input image, we propose aggregating them together to compensate the deficiency of any single Convnet feature. Although a multi-attribute Convnet may contain the same objects as in a single-label and multi-label Convnet, the aggregation feature can further improve the accurate prediction of object classes. Figure 6.4 illustrates the pipeline of generating image captions based on multi-Convnet aggregation.

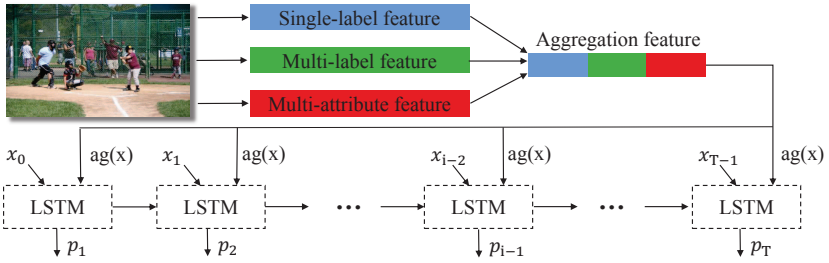

Figure 6.4: The pipeline of Image captioning based on multi-Convnet aggregation. The three Convnet features are concatenated together to generate an aggregation feature $ag(x)$. At each time step, both a word x_i and $ag(x)$ are fed to the LSTM unit whose output is a probability distribution for the next word.

First, the input image x is fed to three pre-trained Convnets to capture separate visual features, denoted as $sc(x), mc(x), ma(x)$. We then concatenate three kinds of features to create an aggregation feature $ag(x)$ (i.e. 1380-Dim vector), where $ag(x) = [sc(x), mc(x), ma(x)]$. Then, we add this aggregation feature to the following RNN unit at each time step. We employ one-layer Long Short-Term Memory (LSTM) [277] that can alleviate the vanishing gradient problem due to its gates mechanism. Finally, at the time step t, the formulation of LSTM units with an aggregation feature can be summarized as below

$$i_t = \sigma(W_{xi}x_t + W_{vi}ag(x) + W_{hi}h_{t-1} + b_i) \tag{6.3}$$

$$f_t = \sigma(W_{xf}x_t + W_{vf}ag(x) + W_{hf}h_{t-1} + b_f) \tag{6.4}$$

$$o_t = \sigma(W_{xo}x_t + W_{vo}ag(x) + W_{ho}h_{t-1} + b_o) \tag{6.5}$$

$$g_t = \phi(W_{xg}x_t + W_{vg}ag(x) + W_{hg}h_{t-1} + b_g) \tag{6.6}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{6.7}$$

$$h_t = o_t \odot \phi(c_t) \tag{6.8}$$

$$p_{t+1} = Softmax(h_t) \tag{6.9}$$

where $W$and b are the weight matrices and bias terms. We refer to x_t as the input word at time step t for image x. σ and ϕ are the sigmoid and tangent activation functions. p_{t+1} is used to predict the probability distribution for the next word. Finally, the objective in LSTMs for language modeling is to minimize the following loss cost

$$-\sum_{t=0}^{T-1} \log p_t(x_{t+1}|x_t, ag(x)) + \lambda||W||_2^2 \tag{6.10}$$

where T is the length of the input sequence of words, and λ indicates the weight decay (In this work, we follow the configuration of [266] and set λ equals 0). For notational simplicity, we just give the computation of one input image and drop the mini-batch size in the formulation. Following the hyper-parameters in [266], both the word embedding size and hidden state size are set to 1000. We use a mini-batch size of 100 image-sentence pairs. The learning rate is initialized with 0.01 and decreases to one tenth of current rate after 20,000 iterations. The whole training will be terminated after 80,000 iterations. In addition, we use a momentum of 0.9 and clip gradients of 10.

6.3.3 Multi-scale Testing

During the test phase, we intend to use a multi-scale augmentation approach to capture a more robust image representation, as shown in Figure 6.5. We first extract a feature vector by inputting a 224×224 image to CNNs. Then, we convert one CNN model to a fully convolutional networks (FCN) [60]. FCN is quite efficient to compute regions based predictions without decreasing the ease of testing. Following [24], we set a smaller side to S and isotropically resize the other side. Here we use two scales of images, including $S = 256$ and 320, and perform

average pooling over the topmost layer of FCN. Finally, the multi-scale feature is computed by averaging one CNN feature and two FCN features. Notably, the multi-scale testing can be used for both single Convnet and multi-Convnet aggregation. We also test more scales such as $S = 384, 512$, but no significant improvement is obtained.

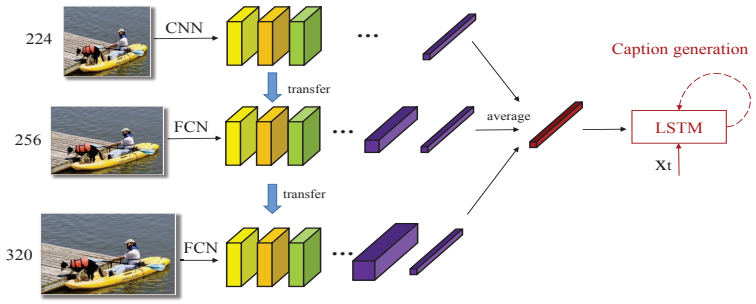

Figure 6.5: The pipeline of multi-scale testing approach. Apart from the basic CNN feature, we use two extra scales based on FCNs. We compute the average over three feature vectors and feed it to LSTM units for caption generation.

6.4 Experiments

In this section, we evaluate our approach on the well-known MS COCO dataset [305]. MS COCO consists of 82783 training images, 40504 validation images and 40775 testing images. Each image is annotated by at least five human-written captions. Following most recent works [266, 302, 306, 312] , we use 5000 images as validation set to tune hyper-parameters, and another 5000 images as test set to report results. We use the vocabulary dictionary in [266] (containing 8800 words). This dictionary is used to encode the input sequence of words. We implemented our approach based on the Caffe framework [218].

6.4.1 Evaluation Configuration

We evaluate our approaches on two tasks: caption generation and image-sentence retrieval. For caption generation task, we evaluate our method with four metrics:

BLEU [313], METEOR [314], ROUGE-L [315] and CIDEr [316]. For image-sentence retrieval task, we divide it into two parts: image-to-sentence retrieval and sentence-to-image retrieval. Following previous works [266, 302], we adopt the evaluation metrics: R@K and Med r. All metrics are computed with the MS COCO evaluation code [317].

We denote the three single Convnets as **SL-Net, ML-Net and MA-Net**. **MA_ML-Net** is the combination of MA-Net and ML-Net, and **MA_ML_SL-Net** indicates the method that aggregates the three Convnets.

We utilize BeamSearch when generating the sentences: iteratively consider the k best sentences up to timestep t when generating sentences of timestep $t+1$. Most of our results use a beam search of size 1 for fast evaluating. For fair comparison with the state-of-the-art, we give the results by using a beam of size 5.

6.4.2 Results on Caption Generation

We evaluate our approach on caption generation with 5000 test images. Table 6.1 shows the single-scale and multi-scale testing of the three Convnets. We list the dimension of the feature since it is closely related with the number of LSTM parameters. It is interesting to see that, SL-Net, which utilizes the largest dimension feature, performs the worst among the three Convnets. This demonstrates that increasing the number of system parameters would not necessarily improve the performance.

For single-scale testing, ML-Net brings about 1% boost over the SL-Net for most evaluation metrics. This improvement is marginal compared to the MA-Net, which outperforms the SL-Net significantly over all the evaluation metrics. Notably, the increase of CIDEr reaches 0.093, from 0.703 to 0.796. On the other hand, the multi-scale testing using FCN shows considerable improvement over the corresponding single-scale testing, with the same feature dimension. This is promising, especially considering the high efficiency of FCN.

In addition to evaluating the three Convnets individually, we also explore the effect of aggregating the Convnets, as shown in Table 6.2. We build the multi-

Table 6.1: MS COCO results on caption generation by comparing three Convnets. Both single-scale and multi-scale testing are shown. Here we use a beam search of size 1.

Method	Dim	B-1	B-2	B-3	B-4	M	R	C
Single-scale Testing:								
SL-Net	1000	0.651	0.474	0.333	0.229	0.214	0.483	0.703
ML-Net	80	0.664	0.487	0.345	0.241	0.213	0.487	0.717
MA-Net	300	0.686	0.516	0.374	0.266	0.228	0.506	0.796
Multi-scale Testing:								
SL-Net	1000	0.666	0.489	0.345	0.239	0.219	0.489	0.735
ML-Net	80	0.679	0.496	0.351	0.245	0.219	0.49	0.75
MA-Net	300	0.697	0.528	0.384	0.274	0.231	0.511	0.81

Convnet based on MA-Net since it is the best individual Convnet. Overall, both MA_ML-Net and MA_MC_SC-Net perform better than the individual MA-Net, indicating that aggregating the Convnet is beneficial for the caption generation. This is reasonable given the fact that different Convnets would learn different contents, and aggregating them generally lead to a more comprehensive prediction. Furthermore, we also evaluate the multi-scale performance using FCN. Similarly, the multi-scale scheme improves the accuracy of the evaluation metric remarkably. Finally, MA_MC_SC-Net can yield a quite competitive result, such as 0.704 B-1 and 0.846 CIDEr.

Table 6.2: MS COCO results on caption generation by multi-Convnet aggregation. The results are based on BLEU, METEOR (M), ROUGE-L (R) and CIDEr (C) metrics. Here we use a beam search of size 1.

Method	B-1	B-2	B-3	B-4	M	R	C
Single-scale Testing:							
MA-Net	0.686	0.516	0.374	0.266	0.228	0.506	0.796
MA_ML-Net	0.687	0.519	0.376	0.268	0.229	0.507	0.797
MA_ML_SL-Net	0.688	0.52	0.379	0.27	0.229	0.507	0.803
Multi-scale Testing:							
MA-Net	0.697	0.528	0.384	0.274	0.231	0.511	0.81
MA_ML-Net	0.703	0.537	0.393	0.282	0.234	0.516	0.846
MA_ML_SL-Net	0.704	0.54	0.398	0.287	0.236	0.519	0.848

Comparison with the state-of-the-art We compare our MA_MC_SC-Net

result with current state-of-the-art methods in Table 6.3. It can be seen that our results delivered better results than most existing methods. Compared to [303], our method obtained the same result on Bleu-1 with the soft-attention model, slightly worse than the more sophisticated hard-attention model. But for all the other evaluation metrics, our method achieved considerably better results. Similar situation comes with [268], with which we also achieved overall competitive performance. It is worthwhile to say that, our method is not inherently conflicted with these methods, and we can incorporate them together for a better achievement. Note that [312] further improved their results by extracting, clustering and selecting a large number of region proposals. Therefore, their great gains are achieved at the expense of algorithm complexity. In contrast, benefited from the high efficiency of FCN, our multi-scale testing strategy brings negligible extra cost compared to the single-scale testing. We argue that a sophisticated region detection approach [44] is also applicable to our system, but it is out of the scope of this work. Figure 6.6 shows some captioning examples.

Table 6.3: Comparison with current state-of-the-art on MS COCO caption generation. Here we use a beam search of size 5.

Method	B-1	B-2	B-3	B-4	M	C
Karpathy et al. [302]	0.625	0.450	0.321	0.230	0.195	0.66
mRNN [304]	0.670	0.490	0.350	0.250	-	-
NIC [267]	-	-	-	0.277	0.237	0.855
LRCN [266]	0.669	0.489	0.349	0.249	-	-
gLSTM [307]	0.670	0.491	0.358	0.264	0.227	0.813
Bi-LSTM [306]	0.672	0.492	0.352	0.244	0.208	0.666
VNet-ft-LSTM [312]	0.680	0.500	0.370	0.250	0.220	0.730
Soft-Attention [303]	**0.707**	0.492	0.344	0.243	0.239	-
Hard-Attention [303]	**0.718**	0.504	0.357	0.250	0.230	-
Jin et al. [308]	0.697	0.519	0.381	0.282	0.235	0.838
ATT-FCN [268]	0.709	0.537	0.402	**0.304**	**0.243**	-
Ours	0.707	**0.548**	**0.410**	**0.304**	0.238	**0.895**

Figure 6.6: The caption generation results for some MS COCO examples by our MA_MC_SC-Net method. We show both the positive and negative examples.

6.4.3 Results on Image-sentence Retrieval

We report the image-to-sentence and sentence-to-image results in Table 6.4. There are 5000 test images and 25,000 captions in total. Overall, MA_MC_SC-Net outperforms other state-of-the-art works on both R@K and Med r.

Table 6.4: Image-sentence retrieval results on MS COCO dataset. R@K: higher is better; Med r: lower is better.

	Image to Sentence				Sentence to Image			
Method	R@1	R@5	R@10	Med r	R@1	R@5	R@10	Med r
Karpathy et al. [302]	16.5	39.2	52.0	9.0	10.7	29.6	42.2	14.0
Bi-LSTM [306]	16.6	39.4	52.4	9.0	11.6	30.9	43.4	13.0
Ours	**16.9**	**39.8**	**53.1**	**8.0**	**12.4**	**31.5**	**44.0**	**12.0**

6.5 Conclusion

In this work, we studied the effects of Convnets for the image captioning task. We employed three Convnets based on single-label, multi-label, multi-attribute classification. In addition, we integrated the three Convnets for an richer aggregation feature. During the test stage, we employed an efficient multi-scale augmentation approach. Experiments on MS COCO dataset demonstrated that our approach achieved competitive results for both caption generation and image-sentence retrieval as compared to the state-of-the-art. In the future work, we will strive to make use of the attention mechanism.