# Hierarchical Multimodal LSTM for Dense Visual-Semantic Embedding

Zhenxing Niu[1] Mo Zhou[2] Le Wang[3] Xinbo Gao[2] Gang Hua[4]
[1]Alibaba Group, [2]Xidian University, [3]Xi'an Jiaotong University, [4]Microsoft Research
{zhenxingniu,ganghua}@gmail.com, lewang@mail.xjtu.edu.cn, xinbogao@mail.xidian.edu.cn

## Abstract

*We address the problem of dense visual-semantic embedding that maps not only full sentences and whole images but also* phrases within sentences *and* salient regions within images *into a multimodal embedding space. Such dense embeddings, when applied to the task of image captioning, enable us to produce several region-oriented and detailed phrases rather than just an overview sentence to describe an image. Specifically, we present a hierarchical structured recurrent neural network (RNN), namely Hierarchical Multimodal LSTM (HM-LSTM). Compared with* chain *structured RNN, our proposed model exploits the* hierarchical relations *between sentences and phrases, and between whole images and image regions, to* jointly *establish their representations. Without the need of any supervised labels, our proposed model automatically learns the fine-grained correspondences between phrases and image regions towards the dense embedding. Extensive experiments on several datasets validate the efficacy of our method, which compares favorably with the state-of-the-art methods.*

## 1. Introduction

Visual-semantic embedding is to map both images and their captions into a common space, so that we can retrieve/rank captions given images or retrieve/rank images given captions. Particularly, it has been broadly used for *image captioning* which aims to describe images with sentences. Recently, the advances in deep learning have made significant progress on visual-semantic embedding. Generally, image representations are produced by Convolutional Neural Networks (CNN), and caption representations are produced by Recurrent Neural Networks (RNN). A ranking loss is subsequently optimized to make the corresponding representations as close as possible in the embedding space [11] [6] [29] [15].

Most previous methods only map *full* sentences and *whole* images into an embedding space. As a result, they are only able to describe an image with a general and overview sentence, *i.e.*, coarsely and generally depict the image content. However, different users may be interested in distinct objects/regions in an image, and hence it is desired to individually depict them with specific descriptions. As shown in Fig. 1, some users may be interested in '*the man with sunglasses*' while others may be interested in '*the girl in red jacket*'. Therefore, it is desired to produce several region-oriented and detailed phrases (*e.g.*, 'a man with a blue hat and sunglasses') rather than just an overview sentence (*e.g.*, 'A man is standing in front of towers') to describe an image.


Figure 1. Region-oriented, detailed, and phrase-level image captioning. It is desired to produce several region-oriented and detailed phrases rather than just an overview sentence for describing an image.

An intuitive solution is to map not only the full sentences but also the *phrases within the sentences* into a common space. As such, for a given image after detecting salient image regions, detailed phrases can be retrieved to describe those image regions. Since long sentences are decomposed as short phrases, many diverse and subtle phrases could be produced. Besides, since more diverse phrases are mapped into the embedding space, we can learn a much *denser* embedding space so that it is possible to find a better and more expressive phrase to describe an image or an image region.

However, most previous methods cannot naturally represent the phrases within sentences, and hence cannot map them into the embedding space. The main reason is that the neural networks (*e.g.*, RNN [10] [9]) adopted for building sentence representations often have a *chain* structure, *i.e.*, a basic unit is unfolded one by one through a chain structure. Therefore, the full sentences are naturally represented with the last hidden state of the chain structured neural network since it encodes all the words within the sentence. But it is difficult to directly build representations for phrases within sentences.

Moreover, previous methods are only able to utilize the correspondences between whole images and full sentences. But there are many fine-grained correspondences between image regions and short phrases, which can be utilized to boost the learning of the embedding space [15]. As shown in Fig. 2, besides the sentence-level correspondence between the sentence 'a cat sat on a mat.' and the whole image, there is a correspondence between the phrase 'a cat' and the corresponding image region, *etc*. Therefore, it is beneficial to exploit and utilize those fine-grained 'phrase-region' correspondences to boost the embedding learning.

To address the two problems above, we propose a Hierarchical Multimodal LSTM (HM-LSTM) model. In particular, our HM-LSTM model has a *hierarchical structure*, where the intermediate nodes represent phrases and regions, while the root nodes represent the full sentences and whole images, as shown in Fig. 4. Thus, our model can naturally and *jointly* learn the embeddings of all sentences, phrases, images and image regions. More importantly, there are *hierarchical relations* between sentences and phrases, and between whole images and image regions. For example, a 'parent' phrase (*e.g.*, 'a cat sat on the mat') is related to its 'children' phrases (*e.g.*, 'a cat' and 'the mat', meanwhile the 'parent' image region covers the two 'children' image regions. Since our model has a hierarchical structure, we can explicitly exploit such hierarchical relations when jointly learning their embeddings. Compared with previous visual-semantic models, our model can map phrases as well as image regions into the embedding space, and hence we can learn a *dense* embedding space, as shown in Fig. 2.

When building representations for phrases, the syntax of phrases is explicitly considered in our model. This is due to that image descriptions often make frequent references to objects, therefore *noun phrases* in a sentence are often more important than the other phrases (*e.g.*, verb phrases). Therefore, noun phrases and the other phrases are distinctively modeled in our HM-LSTM model, *i.e.*, our HM-LSTM model is a syntax-aware model, which is more suitable for the image captioning task.

Note that the fine-grained 'phrase-region' correspondences can be automatically established along with the embedding learning. In other words, we conduct dense visual-semantic embedding in an *unsupervised* fashion, *i.e.*, without the need of manually annotating the correspondences between image regions and phrases. Recently, the DenseCap [13] has been proposed for region-oriented captioning. However, they address this problem in a supervised fashion, *i.e.*, the 'phrase-region' correspondences are given for each training image. Obviously, it is much more expensive to annotate such fine-grained correspondences especially for a large scale dataset. In addition, the phrases annotated in the DenseCap are independent of the full sentences, whereas there are relations among sentences and phrases in our method since the phrases are extracted from the given sentences.

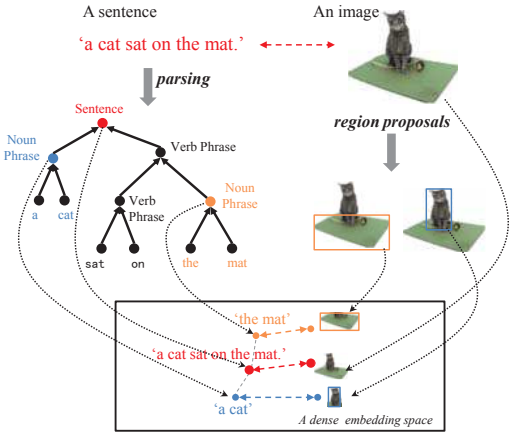



Figure 2. Hierarchical Multimodal Embedding: each sentence is decomposed as some phrases by a tree parser, meanwhile some salient image regions are detected from the image. Then, all of *full sentences*, *phrases*, *whole images*, and *image regions* are mapped into a common space, resulting in a dense embedding space.

Besides, the experimental results turn out that the performance of general image captioning can also be significantly improved due to learning a dense embedding space. This is attributed to the joint embedding of full sentences and their phrases. Since there are hierarchical relations among full sentences and their phrases, such relations could benefit both their embedding learning when they are jointly mapped into the embedding space.

Briefly, our contributions are three-fold:

1. A hierarchical multimodal LSTM model is proposed for dense visual-semantic embedding, which is able to jointly learn the embeddings of all the sentences, phrases, images, and image regions. Moreover, the hierarchical relations among them can be explicitly exploited in our model.
2. The fine-grained correspondences between phrases and image regions can be automatically learned and utilized to boost the learning of the embedding space.
3. Our model is a syntax-aware model where noun phrases and the other phrases are distinctively modeled towards the task of image captioning.

## 2. Related Work

Visual-semantic embedding is closely related to the image captioning. Generally, the methods of image captioning can be roughly grouped into two categories: image caption ranking and image caption generation. Visual-semantic embedding is often regarded as a kind of methods for image caption ranking, *i.e.*, to rank captions given images [6] [29] [14] [15]. DeViSE [6] is a simple model for image caption ranking, where sentences are represented

as the mean of their word embeddings. After that, some sophisticated models such as the SDT-RNN [29] are proposed to learn sentence embedding representations. Recently, Deep Structure-Preserving (DeepSP) [34] is proposed for image-text embedding and achieves the state-of-the-art performance.

For dense embedding, the most related works are the DeepVS [14] and the DeFrag [15], which also align words and short phrases within sentences to bounding boxes. In DeepVS [14], in order to build phrase representations, they additionally apply a Markov Random Field (MRF) to connect neighboring words as a phrase. On the contrary, our hierarchical model can naturally generate syntax-correct phrases and naturally build their representations. In DeFrag [15], although the tree parsing is leveraged for phrase representation, the phrases are independently represented and hence the tree structure is actually discarded in favor of a simpler model. On the contrary, the hierarchical relations among phrases can be explicitly modeled by our method. Moreover, the phrases are jointly instead of independently modeled in our approach.

**Image Caption Generation.** Many methods are proposed for image caption generation [22] [17] [33] [5] [30]. They aim to generate descriptions by sampling from conditional neural language models. Particularly, an 'encoder-decoder' framework [17] [3] is often adopted by those methods, where a CNN is used to represent an image, and an RNN is used to generate descriptions conditioned on the image representation.

## 3. Our Approach

We attempt to map all of *full sentences*, *phrases*, *whole images*, and *image regions* into a common space. Therefore, our approach needs not only to learn the phrase-level correspondences (*i.e.*, the correspondences between phrases and image regions) but also to learn a multimodal embedding space containing all the sentences, phrases, images, and image regions.

Specifically, each sentence is first represented as a Constituency Tree with the Stanford Parser [18], where each intermediate node in the tree indicates a phrase while the root node indicates the full sentence. Meanwhile, for each image, the Region Convolutional Neural Network (R-CNN) [7] is adopted to extract a feature representation for the image region which is generated by using object proposal methods [32].

Next, if the phrase-level correspondences are known, our HM-LSTM model can utilize such correspondences to conduct the embedding learning. In particular, each loss layer is introduced to connect a noun phrase node to an image region, as shown in Fig. 4. At last, all the losses (including 'phrase–region' losses and 'sentence-image' losses) are simultaneously minimized to learn the embedding space.

Input: the 'sentence–image' pairs in the dataset $\{(S_d, I_d)\}_{d=1}^{D}$

1. Initialization stage: **coarse-grained embedding learning**. Only the known sentence-level correspondences are utilized to learn a simplified HM-LSTM model. And then, the initial representations for phrases and image regions are estimated.
2. Loop for $t = 1, ..., T$:
   (a) **Phrase-level correspondences learning**. Given the learned representations of phrases and regions, we establish some 'phrase–region' correspondences $\{(S_{d,k}, I_{d,k})\}_{k=1}^{K_d}$ for each image by measuring their similarity (refers to Section 3.3).
   (b) **Fine-grained embedding learning**. Given the previous phrase-level correspondences, the HM-LSTM model is learned to update the phrase and region representations (refers to Section 3.2.2).

Output: the representations of sentences, phrases, images, and image regions, *i.e.*, $\{(h_{d,k}, v_{d,k})\}_{d=1,k=0}^{d=D,k=K_d}$.

Figure 3. The iterative learning procedure for the hierarchical multimodal embedding.

However, only the sentence-level (rather than the phrase-level) correspondences are known at the beginning. But if we have the representations of all phrases and image regions, it is easy to establish their correspondences, *e.g.*, by measuring the similarities between their representations. Thus, in our approach we take an alternative learning procedure for the embedding learning, *i.e.*, to learn the multimodal embedding space and those phrase-level correspondences alternatively.

In particular, we have an initial learning stage, where only the 'sentence–image' losses are minimized to learn a *simplified HM-LSTM model*. As a result, we are able to produce the initial representations for all the phrases and image regions, which can be further used to construct the initial phrase-level correspondences. After that, a full version of HM-LSTM model (both sentence-level and phrase-level losses are minimized) is learned, and the embedding learning and the correspondences learning can be conducted iteratively, as shown in Fig. 3.

### 3.1. Images Embedding

We follow the work of [14] to represent images. In particular, some object proposals are extracted using the selective search method [32], and they are represented with an R-CNN [7]. Following Karpathy *et al*. [14], we adopt the top 19 detected locations in addition to the whole image, and compute the representations based on the pixels $I_b$ inside each bounding box as follows:

$$v_m = W_m[\text{CNN}_{\theta_c}(I_b)] + b_m \tag{1}$$

where $\text{CNN}(I_b)$ transforms the pixels inside the bounding box $I_b$ into 4096-dimensional activations of the fully connected layer immediately before the classifier.

### 3.2. Hierarchical Multimodal Embedding

Given the phrase-level correspondences, our HM-LSTM model is able to learn a dense embedding space containing all the sentences, phrases, images, and image regions. In particular, we first review the Tree-LSTM model [31] which was recently proposed for sentence embedding. Then it is extended to a syntax-aware model, namely Hierarchical LSTM (H-LSTM) model, where noun phrases and the other phrases are distinctively modeled. At last, our HM-LSTM model is proposed based on the H-LSTM model, which is a multimodal model for joint embedding of sentences, phrases, images, and image regions.

#### 3.2.1 Hierarchical LSTM

Recently, the Tree-LSTM model [31] has been proposed to explicitly model the hierarchical structure of sentences. In particular, a sentence is parsed as a tree, where the root indicates the full sentence and the intermediate nodes indicate the phrases within the sentence.

In Tree-LSTM, children nodes are *equally* treated when connected to their parent node without considering their syntax type – *noun phrase* children and *the other phrase* children (*e.g.*, verb phrase) are equally treated. However, since our task mostly focuses on objects, noun phrases and the other phrases are modeled with different emphasis, *i.e.*, the noun phrase children should have larger contributions than the other phrase children.

To this end, we extend the Tree-LSTM as a syntax-aware model, namely Hierarchical LSTM (H-LSTM) model. Specifically, each unit of H-LSTM (indexed by $j$) contains an input gate $i_j$, an output gate $o_j$, a memory cell $c_j$, and a hidden state $h_j$. Suppose there are $N(j)$ *noun phrase* children for $j$, and $\overline{N}(j)$ *the other phrase* children for $j$, each H-LSTM unit will have $N(j)$ forget gates $\widehat{f}_{jk}, k \in N(j)$ and $\overline{N}(j)$ forget gates $\overline{f}_{jl}, l \in \overline{N}(j)$, as in Eq (3) and Eq (4).

For a parent node $j$, the hidden state of its noun phrase children $h_k, k \in N(j)$ and the other phrase children $h_l, l \in \overline{N}(j)$ are respectively summed up (denoted as $\widehat{h}_j$ and $\overline{h}_j$) before impacting the parent node $j$, as in Eq (2). Furthermore, the $\widehat{h}_j$ and $\overline{h}_j$ have different effects on the input gate $i_j$ by using distinct parameters $\widehat{U}^{(o)}$ and $\overline{U}^{(o)}$, as shown in Eq (5). It is similar for the output gate $o_j$ and memory cell $c_j$, as shown in Eq (6), and Eq (7). This allows the H-LSTM to sufficiently consider the syntax type of children nodes.

$$\widehat{h}_j = \sum_{k \in N(j)} h_k; \quad \overline{h}_j = \sum_{l \in \overline{N}(j)} h_l \tag{2}$$

$$\widehat{f}_{jk} = \sigma(W^{(f)} x_j + \widehat{U}^{(f)} h_k + b^{(f)}), \quad k \in N(j) \tag{3}$$

$$\overline{f}_{jl} = \sigma(W^{(f)} x_j + \overline{U}^{(f)} h_l + b^{(f)}), \quad l \in \overline{N}(j) \tag{4}$$

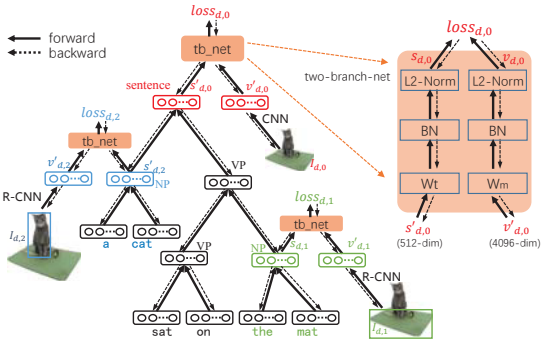



Figure 4. The structure of our HM-LSTM. Each sentence is parsed as a tree, where the intermediate nodes indicate the phrases within the sentence. Some noun phrases (NP) $h_{d,k}$ are associated to the corresponding image regions $v_{d,k}$ by specific loss layers $loss_{d,k}$.

$$i_j = \sigma(W^{(i)} x_j + \widehat{U}^{(i)} \widehat{h}_j + \overline{U}^{(i)} \overline{h}_j + b^{(i)}) \tag{5}$$

$$o_j = \sigma(W^{(o)} x_j + \widehat{U}^{(o)} \widehat{h}_j + \overline{U}^{(o)} \overline{h}_j + b^{(o)}) \tag{6}$$

$$u_j = \tanh(W^{(u)} x_j + \widehat{U}^{(u)} \widehat{h}_j + \overline{U}^{(u)} \overline{h}_j + b^{(u)})$$

$$c_j = i_j \odot u_j + \sum_{k \in N(j)} \widehat{f}_{jk} c_k + \sum_{l \in \overline{N}(j)} \overline{f}_{jl} c_l \tag{7}$$

$$h_j = o_j \odot \tanh(c_j) \tag{8}$$

As the standard LSTM, each H-LSTM leaf node takes an input vector $x_j$. In our applications, each $x_j$ is a vector representation of a word, which is determined as $x_j = W_w \mathbb{I}_t$, where $\mathbb{I}_t$ is an indicator column vector that has a single one at the index of the $t$-th word in a word vocabulary. The weights $W_w$ specify a word embedding matrix that we initialize with 300-dimensional word2vec [24] weights and keep fixed due to overfitting concerns. In addition, as the Tree-LSTM model, the hidden state $h_j$ of node $j$ is regarded as the representation of the corresponding phrase.

#### 3.2.2 Hierarchical Multimodal LSTM

Based on the H-LSTM, we propose a Hierarchical Multimodal LSTM (HM-LSTM) to jointly embed all of images, image regions, sentences, and phrases into a common space.

Let $I_{d,k}$ denote the $k$-th image region in the $d$-th image, $S_{d,k}$ denote the corresponding phrase. And let $I_{d,0}$ denote the $d$-th full image, and $S_{d,0}$ denote the corresponding full sentence. If all the 'phrase-region' pairs $\{(S_{d,k}, I_{d,k})\}_{d=1,k=0}^{D,K_d}$ are known, we learn the HM-LSTM as follows: a H-LSTM model is first constructed for each sentence, and for each 'phrase-region' pair $(S_{d,k}, I_{d,k})$ a loss layer $loss_{d,k}$ is introduced. Inspired by DeepSP [34], we introduce a two-branch-network instead of a simple loss layer for each 'phrase-region' pair. Specifically, each branch is composed of one fully connected layers ($W_t$ for text and $W_m$ for images), one Batch Normalization (BN) layer [12], and one L2-normalization layer, as shown in

Fig 4. Note that the batch normalization could accelerate the training and also make gradient updates more stable.

Let $v_{d,k}$ indicate the representation of the $I_{d,k}$, and $h_{d,k}$ indicate the representation of the $S_{d,k}$. We can define a scoring function $s(v_{d,k}, h_{d,k}) = v_{d,k} \cdot h_{d,k}$ to measure their similarity. Therefore, for each 'phrase-region' pair $(S_{d,k}, I_{d,k})$, we define a contrastive loss to measure the distance between their representations, as the following,

$$\begin{aligned} loss_{d,k} = &\sum_{l} max\{0, m - s(v_{d,k}, h_{d,k}) + s(v_{d,k}, h_{d,l})\} \\ &+ \sum_{l} max\{0, m - s(h_{d,k}, v_{d,k}) + s(h_{d,k}, v_{d,l})\} \end{aligned} \tag{9}$$

where $m$ is the margin, $h_{d,l}$ is a contrastive phrase for image region $v_{d,k}$, and vice-versa with $v_{d,l}$.

Next, the total loss can be defined by the weighted sum of all losses, as the following,

$$Loss = \sum_{d=1}^{D} \sum_{k=0}^{K_d} w_{d,k} loss_{d,k} \tag{10}$$

where $w_{d,k}$ is the weight for the $k$-th 'phrase-image region' pair. The $loss_{d,0}$ indicates the loss at the root layer for the $d$-th image, and $loss_{d,k}, k = 1, \ldots, K_d$ indicates the loss at the intermediate layer, as shown in Fig. 4.

The weight $w_{d,k}$ can be determined from the learning of phrase-level correspondences, *e.g.*, the $w_{d,k}$ is determined according to the confidence of the correspondence for the $k$-th 'phrase-region' pair.

Note that our HM-LSTM model is learned with the Back-propagation Through Structure (BPTS) algorithm [8], where the errors of different loss functions are respectively injected to the corresponding loss layers, and back propagated from root node to leaf nodes along the tree structure.

### 3.3. Phrase-level Correspondences

Before the learning of the HM-LSTM, we need to obtain the phrase-level correspondences. We can address this problem by measuring the representation similarities among phrase candidates and image region candidates.

Specifically, given the image region candidates (*i.e.*, the top-19 object proposals), their representations can be easily obtained according to Eq (1). Meanwhile, each sentence is parsed as a tree, where each intermediate node in the tree represents a phrase. Due to that we are just interested in objects in an image, only noun phrases are selected as the phrase candidates. Such selection is trivial since the syntax type of each phrase (*i.e.*, noun phrase, verb phrase, adjective phrase, *etc.*) is available after parsing.

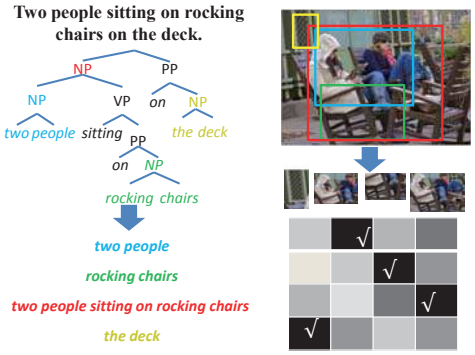



Figure 5. Correspondences between phrases and image regions.

With those image region and phrase candidates, we can establish the 'phrase-region' correspondences according to their representations. In particular, we compute a matrix $S$ to measure the similarities of representations for those candidates, where each element $s_{ij} = v_i \cdot h_j$ indicates the similarity score between the image region $v_i$ and the phrase $h_j$. Therefore, for each phrase we select the best matched image region, and thus we can establish those 'phrase-image region' pairs, as shown in Fig 5. Besides, for each generated 'phrase-region' pair $(v_i, h_j)$, their similarity score $s_{ij}$ is regarded as the confidence of their correspondence, which is used to determine the weight of this correspondence, as shown in Eq (10).

### 3.4. Initialization and Optimization

**Initialization**. At the initial learning stage, the initial representations for all of sentences, phrases, images, and image regions are obtained by learning a *simplified HM-LSTM model* – only the losses at the root $\{loss_{d,0}\}_{d=1}^{D}$ are minimized and the other losses $\{loss_{d,k}\}_{d=1,k=1}^{D,K_d}$ are neglected. Obviously, only the sentence-level correspondences are used to learn the simplified HM-LSTM.

**Optimization**. The CNN part of our model comes from Karpathy *et al*. [14], which is pre-trained on ImageNet [4] and fine tuned on the 200 classes of the ImageNet Detection Challenge [28]. We use Adam [16] to optimize the HM-LSTM with a learning rate of $8 \times 10^{-3}$. In particular, we use mini-batches of 64 paired image-sentences for training.

## 4. Image Caption Ranking

With the learned hierarchical multimodal embedding model, we can describe a new image with a full sentence, *i.e.*, *image-sentence ranking*. In particular, we first extract image features by using the CNN and retrieve the nearest sentence vector $h_{d^*,0} \in \{h_{d,0}\}_{d=1}^{d=D}$ in the embedding space, which is regarded as the caption for the image.

More importantly, our method can produce region-oriented phrase-level description for a new image. In par-

Table 1. Flickr8K experiments. **R@K** is Recall@K (high is good). **Med r** is the median rank (low is good).

| Flickr8K | | | | | | |
|---|---|---|---|---|---|---|
| Model | Image Annotation | | | Image Search | | |
| | **R@1** | **R@10** | **Med $r$** | **R@1** | **R@10** | **Med $r$** |
| Random | 0.1 | 1.1 | 631 | 0.1 | 1.0 | 500 |
| SDT-RNN [29] | 6.0 | 34.0 | 23 | 6.6 | 31.7 | 25 |
| DeViSE [6] | 4.8 | 27.3 | 28 | 5.9 | 29.6 | 29 |
| DeFrag [15] | 12.6 | 44.0 | 14 | 9.7 | 42.5 | 15 |
| SC-NLM [17] | 13.5 | 45.7 | 13 | 10.4 | 43.7 | 14 |
| DeepVS [14] | 16.5 | 54.2 | 7.6 | 11.8 | 44.7 | 12.4 |
| m-RNN [22] | 14.5 | 48.5 | 11 | 11.5 | 42.4 | 15 |
| NIC [33] | 20 | 61 | 6 | 19 | 64 | 5 |
| HM-LSTM | **27.7** | **68.6** | **5** | **24.4** | **68.1** | **4** |

Table 2. Flickr30K experiments. **R@K** is Recall@K (high is good). **Med r** is the median rank (low is good).

| Flickr30K | | | | | | |
|---|---|---|---|---|---|---|
| Model | Image Annotation | | | Image Search | | |
| | **R@1** | **R@10** | **Med $r$** | **R@1** | **R@10** | **Med $r$** |
| Random | 0.1 | 1.1 | 631 | 0.1 | 1.0 | 500 |
| SDT-RNN [29] | 9.6 | 41.1 | 16 | 8.9 | 41.1 | 16 |
| DeViSE [6] | 4.5 | 29.2 | 26 | 6.7 | 32.7 | 25 |
| DeFrag [15] | 14.2 | 51.3 | 10 | 10.2 | 44.2 | 14 |
| SC-NLM [17] | 14.8 | 50.9 | 10 | 11.8 | 46.3 | 13 |
| DeepVS [14] | 22.2 | 61.4 | 4.8 | 15.2 | 50.5 | 9.2 |
| m-RNN [22] | 18.4 | 50.9 | 10 | 12.6 | 41.5 | 16 |
| NIC [33] | 17.0 | 56.0 | 7 | 17.0 | 57.0 | 7 |
| m-RNN-vgg [23] | 35.4 | 73.7 | **3** | 22.8 | 63.1 | 5 |
| DeepSP [34] | 35.7 | 74.4 | N/A | 25.1 | 66.5 | N/A |
| HM-LSTM | **38.1** | **76.5** | **3** | **27.7** | **68.8** | **4** |

Table 3. MS-COCO experiments. **R@K** is Recall@K (high is good). **Med r** is the median rank (low is good).

| MS-COCO | | | | | | |
|---|---|---|---|---|---|---|
| Model | Image Annotation | | | Image Search | | |
| | **R@1** | **R@10** | **Med $r$** | **R@1** | **R@10** | **Med $r$** |
| Random | 0.1 | 1.1 | 631 | 0.1 | 1.0 | 500 |
| DeepVS [14] | 36.4 | 80.9 | 3 | 28.1 | 76.1 | **3** |
| m-RNN-vgg [23] | 41.0 | 83.5 | **2** | 29.0 | 77.0 | **3** |
| DeepSP [34] | 40.7 | 85.3 | N/A | 33.5 | 83.2 | N/A |
| HM-LSTM | **43.9** | **87.8** | **2** | **36.1** | **86.7** | **3** |

ticular, after detecting some salient image regions/object proposals, we can extract the visual features from them, and retrieve specific and detailed phrases to describe them, namely *region-phrase ranking* in this paper.

## 5. Experiments

We use the Flickr8K [11], Flickr30K [35] [25] and MS-COCO [20] [2] datasets in our experiments. These datasets contain $8,000$, $31,000$ and $123,000$ images respectively and each is annotated with $5$ sentences using AMT. For Flickr8K and Flickr30K, we use $1,000$ images for validation, $1,000$ for testing and the rest for training, which is consistent with [11][14]. For MS-COCO we follow [14] to use $5,000$ images for both validation and testing.

### 5.1. Image-Sentence Ranking

We first evaluate the proposed method on the task of image-sentence ranking. We adopt Recall@K as the metric for evaluation, namely the mean number of images for which the correct caption is ranked within the top-$K$ retrieved results (and vice-versa for sentences).

We compare our method with some visual-semantic embedding methods (*i.e.*, ranking-based methods) including DeViSE, SDT-RNN, and DeFrag. For DeViSE [6], sentences are represented as the mean of their word embeddings. The recursive neural network is used to learn sentence representations in SDT-RNN [29]. For DeFrag [15], sentences are represented as a bag of dependency parses.

In addition, some generation-based methods are also involved in comparison. The m-RNN [22] and m-RNN-vgg [23] are methods that do not use a ranking loss and instead optimizes the log-likelihood of predicting the next word in a sequence conditioned on an image. The DeepVS [14] is proposed to first learn an embedding space with a bidirectional-RNN, and then train an RNN sentence generator based on the embedding space. Similarly, the NIC [33] is another method that provides the visual input directly to the RNN model. Recently, Deep Structure-Preserving (DeepSP) [34] is proposed for image-text embedding and achieves the state-of-the-art performance, where the captions for the same image are encouraged to be close to each other.

#### 5.1.1 Results on Flickr8K and Flickr30K

We evaluate our approach on the Flickr8K and Flickr30K. Particularly, the dimension of the embedding space is set as $512$, *i.e.*, $h_j$ and $v_i$ are 512-dimensional vectors.

The *R@K* and *Med r* of different methods are shown in Table 1 and Table 2. Our model outperforms the ranking-based methods by a large margin. Besides, our method also compares favorably with the state-of-the-art methods.

The results of DeepSP [34] in Table 2 are based on the mean vector representations, *i.e.*, a sentence is represented as the mean of their word embeddings. This is a fair comparison since both our model and this version of DeepSP are based on the same word embeddings – word2vec representation [24]. Note that if more sophisticated sentence representations such as Fisher vector (FV) are utilized, the performance of DeepSP could be further improved [34]. However, the memory cost is huge and hence it is not well-suited to a large scale image-sentence ranking task.

#### 5.1.2 Results on MS-COCO

On the dataset of MS-COCO, we follow the experimental setting of [14] to randomly sample $1,000$ images for testing. Specifically, the dimension of the embedding space is set as $512$, and the Multiscale Combinatorial Grouping

(MCG) [1] is adopted to replace the R-CNN to generate object proposals.

The results of the ranking tasks are shown in Table 3. Obviously, we can see that our method significantly outperforms the ranking-based methods. Even for the state-of-the-art methods such as m-RNN-vgg [23] and DeepSP [34], our approach still compares favorably with them.

From the results of image-sentence ranking on all three datasets, we have a conclusion that the performance of general image captioning could be significantly improved by learning a dense embedding space. This is attributed to the joint embedding of full sentences and their phrases. Since there are hierarchical relations among full sentences and their phrases, such relations could benefit both their embedding learning when they are jointly represented and mapped into the embedding space.

## 5.2. Region-Phrase Ranking

Our method can produce region-oriented phrase-level description for a new image. Generally, after detecting some salient image regions/object proposals, our model can retrieve subtle and detailed phrases to describe them. For easier evaluation, the image regions are manually annotated instead of being automatically detected in this experiments.

For quantitative evaluation, we publish a new dataset based on the MS-COCO, namely **MS-COCO-region** dataset. Specifically, 1000 images and corresponding sentences are randomly selected from the MS-COCO validation set. And then, AMT workers [27] are asked to annotate image regions in those images and associate them to the phrases within the sentences. Although some phrase-level captioning datasets such as Visual Genome [19] and Flickr30k-Entities [26] have been proposed, their phrases either are freely annotated by workers or have no relations with the sentences. On the contrary, the phrases in MS-COCO-region dataset are *automatically* extracted from the given sentences, and there are *hierarchical relations* between sentences and phrases.

Specifically, for each sentence, $1 \sim 5$ noun phrases are automatically extracted by using Stanford Parser. For each image, some AMT workers are asked to annotate $1 \sim 8$ regions and associate them to those extracted phrases. As a result, 4467 salient regions and 18724 corresponding phrases are collected in total.

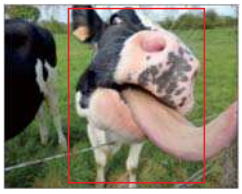
For comparison, DeepVS and m-RNN-vgg are adopted as baselines, where each region-phrase pair is independently fed to those models to obtain their embeddings. The results of region-phrase ranking are shown in Table 4. Obviously, our method outperforms both DeepVS and m-RNN-vgg. It is mainly because (1) the relations among phrases are better utilized due to the hierarchical structure of our model, and (2) the chain structured RNN is good at representing long sequences (*i.e.*, full sentences) instead of short sequences (*i.e.*, phrases). So we have a conclusion that our model can jointly represent short phrases along with long sentences, and better utilize their relations as well.

Table 4. Region-Phrase Ranking. **R@K** is Recall@K (high is good). **Med r** is the median rank (low is good).

| Region Annotation | | | | |
|---|---|---|---|---|
| Model | **R@1** | **R@5** | **R@10** | **Med** $r$ |
| Random | 0.02 | 0.12 | 0.24 | 3133 |
| DeepVS [14] | 7.2 | 18.1 | 26.8 | 64 |
| m-RNN-vgg [23] | 8.1 | 20.6 | 28.2 | 56 |
| HM-LSTM | **10.8** | **22.6** | **30.7** | **42** |

(1) *a white and gray cat with a striped tail*
(2) *a close up of a cat laying next to a mouse*
(3) *a white and gray cat*
(4) *a cat with an intent look*
(5) *his cat below*

(a) a region of 'cat'



(1) *a cow standing in the grass with a tag in its ear*
(2) *a cow with a black face*
(3) *a cow staring into the camera*
(4) *mother cow laying next to her baby on the grass*
(5) *a close up of a black and white cow*

(b) a region of 'cow'

Figure 6. Our approach can produce subtle and detailed descriptions for an image region. Besides, many descriptions are diverse so that they can describe different aspects of an object.

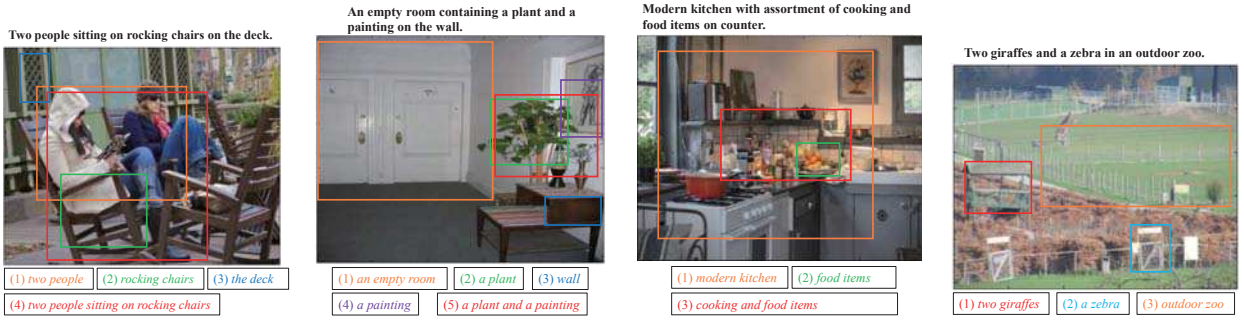
**Qualitative results**. Our method can describe image regions with detailed and subtle phrases. For example, for the Fig. 6(a) previous methods tend to describe it with a general and overview description, *e.g.*, '*A cat* sitting under an umbrella'. In contrast, our method targets a salient image region (*e.g.*, which is marked by red box), and produce detailed and subtle descriptions such as 'a white and gray cat with a strip tail'. Compared to the coarse description 'a cat', our description is more informative and expressive.

In addition, our approach can produce some diverse descriptions for a given image region. As shown in Fig. 6(b), for the image region containing a 'cow', the top-5 retrieved phrases diversely describe the 'cow', *e.g.*, 'a cow standing in the grass with a tag in its ear' focuses on the ear of the cow, while 'a cow staring into the camera' focuses on the action of the cow. In other words, our approach can diversely describe different aspects of an object of interest.

## 5.3. Discussion

### 5.3.1 Learned Embedding Space

To intuitively and qualitatively check the properties of the learned embedding space, we visualize the learned embedding vectors in a 2-D space by using t-SNE [21]. Specifically, we randomly sample 60 images and corresponding sentences from our MS-COCO testing dataset. And their embedding vectors are visualized in a 2-D space, as shown in Fig. 8. Particularly, we connect each image embedding to 5 corresponding sentence embeddings by lines. We can see




(a) (b) (c) (d)

Figure 7. Four examples of the learned correspondences between phrases and image regions. For image (a), we obtain 4 phrases after sentence parsing: (1) 'two people', (2) 'rocking chairs', (3) 'the deck', and (4) 'two people sitting on rocking chairs', meanwhile some salient image regions are obtained. The learned correspondences between phrases and image regions are indicated by their color, *e.g.*, the phrase 'two people' corresponds to the orange box. Obviously, our approach can learn correct correspondences in most cases. Note that (d) is a failure example, it is mainly due to that the salient regions do not cover the objects mentioned in its caption.

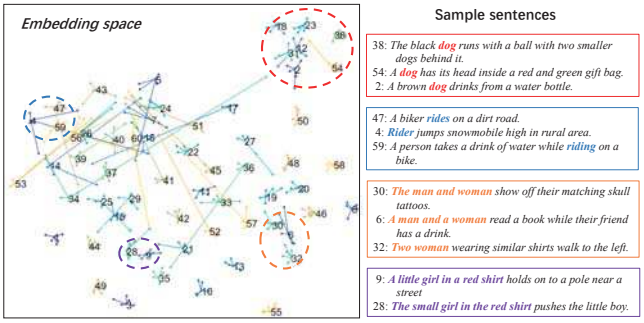



Figure 8. The visualization of the learned embedding space. Each image is connected to 5 corresponding sentences by lines. Obviously, the image and the corresponding sentences are very close to each other in most cases. Besides, images/sentences with similar semantics are also close to each other, *e.g.*, the 38-th, 54-th, and 2-nd images are all related to '*Dog*', and their embeddings are exactly neighbors in the embedding space (within the red circle).

that the learned image embedding is very close to its sentence embeddings in most cases, which demonstrates the effectiveness of our approach.

Moreover, from Fig. 8 we can see that our model can learn a *semantic* embedding space, where images/sentences with similar semantics will be mapped close to each other. For example, the 38-th, 54-th, and 2-th images are all related to '*Dog*' (as shown by their descriptions). And their learned embedding vectors are exactly neighbors in the embedding space (within the red circle).

### 5.3.2 Learned Phrase-level Correspondences

When learning the dense embedding space, our approach can automatically find the 'phrase-region' correspondences in the training data. We evaluate the quality of those learned correspondences here. Since it is expensive to obtain the ground truth phrase-level correspondences, we only make an evaluation on a subset of training data. In practice, we randomly sample 2000 'phrase-region' pairs from all learned phrase-level correspondences, and ask 10 users to judge whether each pair is correct. After a majority voting among those users, we find out that 82% learned correspondences are correct.

Fig. 7 illustrates four examples of the learned correspondences between phrases and image regions. In most cases, our approach is able to find correct correspondences. Moreover, there are consistent mappings between the phrases' as well as the regions' hierarchical structures, *e.g.*, the phrase 'two people sitting on rocking chairs' is on top of two phrases 'two people' and 'rocking chairs', meanwhile the red box for 'two people sitting on rocking chairs' exactly cover the orange box for 'two people' and the green box for 'rocking chairs', *etc*.

## 6. Conclusion

In this paper, a Hierarchical Multimodal LSTM model is proposed for dense visual-semantic embedding, which can jointly learn the embeddings of all the sentences, their phrases, images, and salient image regions. Due to the hierarchical structure, we can naturally build representations for all phrases and image regions, and exploit their hierarchical relations as well. The experimental results turn out that the performance of general image captioning can be significantly improved due to learning a dense embedding space. Bedsides, our method can produce detailed and diverse phrases to describe image salient regions.

## 7. Acknowledgement

This work was supported by NSFC Grant 61432014, U1605252, 61402348, 61672402, 61602355, and 61503296, by Key Industrial Innovation Chain 2016KTZDGY-02 and National High-Level Talents CS31117200001. Dr. Gang Hua was supported by NSFC Grant 61629301.

## References


[1] P. Arbelaez, J. Tuset, J. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. *CVPR*, 2014.

[2] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollar, and C. Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv:1504.00325*, 2015.

[3] K. Cho, B. Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv:1406.1078*, 2014.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. *CVPR*, 2009.

[5] J. Devlin, H. Cheng, H. Fang, S. Gupta, L. Deng, X. He, G. Zweig, and M. Mitchell. Language models for image captioning: The quirks and what works. *arXiv:1505.01809*, 2015.

[6] A. Frome, G. Corrado, J. Shlens, S. Bengio, J. Dean, and T. Ranzato. Devise: A deep visual-semantic embedding model. *NIPS*, 2013.

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *CVPR*, 2014.

[8] C. Goller and A. Kuchler. Learning task-dependent distributed representations by backpropagation through structure. *ICNN*, 1996.

[9] A. Graves. Generating sequences with recurrent neural networks. *arXiv:1308.0850*, 2013.

[10] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 1997.

[11] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *JAIR*, 2013.

[12] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ICML*, 2015.

[13] J. Johnson, A. Karpathy, and L. Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. *CVPR*, 2016.

[14] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. *CVPR*, 2015.

[15] A. Karpathy, A. Joulin, and L. Fei-Fei. Deep fragment embeddings for bidirectional image-sentence mapping. *NIPS*, 2014.

[16] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *ICLR*, 2015.

[17] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *NIPS*, 2014.

[18] D. Klein and C. Manning. Accurate unlexicalized parsing. *ACL*, 2003.

[19] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L. Li, D. Shamma, M. Bernstein, and L. Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *https://arxiv.org/abs/1602.07332*, 2016.

[20] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. Zitnick. Microsoft coco: Common objects in context. *arXiv:1405.0312*, 2014.

[21] V. Maaten and G. Hinton. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9(11):2579–2605, 2008.

[22] J. Mao, W. Xu, Y. Yang, J. Wang, and A. Yuille. Explain images with multimodal recurrent neural networks. *NIPS*, 2014.

[23] J. Mao, W. Xu, Y. Yang, J. Wang, and A. Yuille. Deep captioning with multimodal recurrent neural networks. *ICLR*, 2015.

[24] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. *NIPS*, 2013.

[25] B. Plummer, L. Wang, C. Cervantes, J. Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *ICCV*, 2015.

[26] B. Plummer, L. Wang, C. Cervantes, J. Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *IJCV*, 2016.

[27] C. Rashtchian, P. Young, M. Hodosh, and J. Hockenmaier. Collecting image annotations using amazons mechanical turk. *NAACL-HLT workshop*, 2010.

[28] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *CVPR*, 2014.

[29] R. Socher, Q. Le, C. Manning, and A. Ng. Grounded compositional semantics for finding and describing images with sentences. *TACL*, 2014.

[30] I. Sutskever, O. Vinyals, and Q. Le. Sequence to sequence learning with neural networks. *NIPS*, 2014.

[31] K. Tai, R. Socher, and C. Manning. Improved semantic representations from tree-structured long short-term memory networks. *ACL*, 2015.

[32] J. Uijlings, K. Sande, T. Gevers, and A. Smeulders. Selective search for object recognition. *IJCV*, 2013.

[33] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. *arXiv:1411.4555*, 2014.

[34] L. Wang, Y. Li, and S. Lazebnik. Learning deep structure-preserving image-text embeddings. *CVPR*, 2016.

[35] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2014.