Deep Visual-Semantic Alignments for Generating Image Descriptions

Andrej Karpathy Li Fei-Fei
Department of Computer Science, Stanford University
{karpathy,feifeili}@cs.stanford.edu

Abstract

We present a model that generates natural language descriptions of images and their regions. Our approach leverages datasets of images and their sentence descriptions to learn about the inter-modal correspondences between language and visual data. Our alignment model is based on a novel combination of Convolutional Neural Networks over image regions, bidirectional Recurrent Neural Networks over sentences, and a structured objective that aligns the two modalities through a multimodal embedding. We then describe a Multimodal Recurrent Neural Network architecture that uses the inferred alignments to learn to generate novel descriptions of image regions. We demonstrate that our alignment model produces state of the art results in retrieval experiments on Flickr8K, Flickr30K and MSCOCO datasets. We then show that the generated descriptions significantly outperform retrieval baselines on both full images and on a new dataset of region-level annotations.

1. Introduction

A quick glance at an image is sufficient for a human to point out and describe an immense amount of details about the visual scene [14]. However, this remarkable ability has proven to be an elusive task for our visual recognition models. The majority of previous work in visual recognition has focused on labeling images with a fixed set of visual categories and great progress has been achieved in these endeavors [45, 11]. However, while closed vocabularies of visual concepts constitute a convenient modeling assumption, they are vastly restrictive when compared to the enormous amount of rich descriptions that a human can compose.

Some pioneering approaches that address the challenge of generating image descriptions have been developed [29, 13]. However, these models often rely on hard-coded visual concepts and sentence templates, which imposes limits on their variety. Moreover, the focus of these works has been on reducing complex visual scenes into a single sentence, which we consider to be an unnecessary restriction.

In this work, we strive to take a step towards the goal of generating dense descriptions of images (Figure 1). The primary challenge towards this goal is in the design of a model that is rich enough to simultaneously reason about contents of images and their representation in the domain of natural language. Additionally, the model should be free of assumptions about specific hard-coded templates, rules or categories and instead rely on learning from the training data. The second, practical challenge is that datasets of image captions are available in large quantities on the internet [21, 58, 37], but these descriptions multiplex mentions of several entities whose locations in the images are unknown.

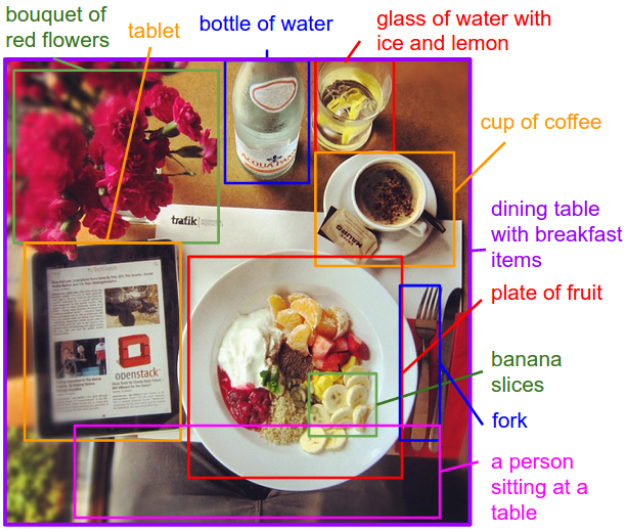


Figure 1. Motivation/Concept Figure: Our model treats language as a rich label space and generates descriptions of image regions.

Our core insight is that we can leverage these large image-sentence datasets by treating the sentences as weak labels, in which contiguous segments of words correspond to some particular, but unknown location in the image. Our approach is to infer these alignments and use them to learn a generative model of descriptions. Concretely, our contributions are twofold:

- We develop a deep neural network model that infers the latent alignment between segments of sentences and the region of the image that they describe.

Our model associates the two modalities through a common, multimodal embedding space and a structured objective. We validate the effectiveness of this approach on image-sentence retrieval experiments in which we surpass the state-of-the-art.

- We introduce a multimodal Recurrent Neural Network architecture that takes an input image and generates its description in text. Our experiments show that the generated sentences significantly outperform retrieval-based baselines, and produce sensible qualitative predictions. We then train the model on the inferred correspondences and evaluate its performance on a new dataset of region-level annotations.

We make code, data and annotations publicly available. [1]

2. Related Work

Dense image annotations. Our work shares the high-level goal of densely annotating the contents of images with many works before us. Barnard et al. [2] and Socher et al. [48] studied the multimodal correspondence between words and images to annotate segments of images. Several works [34, 18, 15, 33] studied the problem of holistic scene understanding in which the scene type, objects and their spatial support in the image is inferred. However, the focus of these works is on correctly labeling scenes, objects and regions with a fixed set of categories, while our focus is on richer and higher-level descriptions of regions.

Generating descriptions. The task of describing images with sentences has also been explored. A number of approaches pose the task as a retrieval problem, where the most compatible annotation in the training set is transferred to a test image [21, 49, 13, 43, 23], or where training annotations are broken up and stitched together [30, 35, 31]. Several approaches generate image captions based on fixed templates that are filled based on the content of the image [19, 29, 13, 55, 56, 9, 1] or generative grammars [42, 57], but this approach limits the variety of possible outputs. Most closely related to us, Kiros et al. [26] developed a log-bilinear model that can generate full sentence descriptions for images, but their model uses a fixed window context while our Recurrent Neural Network (RNN) model conditions the probability distribution over the next word in a sentence on all previously generated words. Multiple closely related preprints appeared on Arxiv during the submission of this work, some of which also use RNNs to generate image descriptions [38, 54, 8, 25, 12, 5]. Our RNN is simpler than most of these approaches but also suffers in performance. We quantify this comparison in our experiments.

Grounding natural language in images. A number of approaches have been developed for grounding text in the visual domain [27, 39, 60, 36]. Our approach is inspired by Frome et al. [16] who associate words and images through a semantic embedding. More closely related is the work of Karpathy et al. [24], who decompose images and sentences into fragments and infer their inter-modal alignment using a ranking objective. In contrast to their model which is based on grounding dependency tree relations, our model aligns contiguous segments of sentences which are more meaningful, interpretable, and not fixed in length.

Neural networks in visual and language domains. Multiple approaches have been developed for representing images and words in higher-level representations. On the image side, Convolutional Neural Networks (CNNs) [32, 28] have recently emerged as a powerful class of models for image classification and object detection [45]. On the sentence side, our work takes advantage of pretrained word vectors [41, 22, 3] to obtain low-dimensional representations of words. Finally, Recurrent Neural Networks have been previously used in language modeling [40, 50], but we additionally condition these models on images.

3. Our Model

Overview. The ultimate goal of our model is to generate descriptions of image regions. During training, the input to our model is a set of images and their corresponding sentence descriptions (Figure 2). We first present a model that aligns sentence snippets to the visual regions that they describe through a multimodal embedding. We then treat these correspondences as training data for a second, multimodal Recurrent Neural Network model that learns to generate the snippets.

3.1. Learning to align visual and language data

Our alignment model assumes an input dataset of images and their sentence descriptions. Our key insight is that sentences written by people make frequent references to some particular, but unknown location in the image. For example, in Figure 2, the words *"Tabby cat is leaning"* refer to the cat, the words *"wooden table"* refer to the table, etc. We would like to infer these latent correspondences, with the eventual goal of later learning to generate these snippets from image regions. We build on the approach of Karpathy et al. [24], who learn to ground dependency tree relations to image regions with a ranking objective. Our contribution is in the use of bidirectional recurrent neural network to compute word representations in the sentence, dispensing of the need to compute dependency trees and allowing unbounded interactions of words and their context in the sentence. We also substantially simplify their objective and show that both modifications improve ranking performance.

We first describe neural networks that map words and image regions into a common, multimodal embedding. Then we introduce our novel objective, which learns the embedding

[1] cs.stanford.edu/people/karpathy/deepimagesent

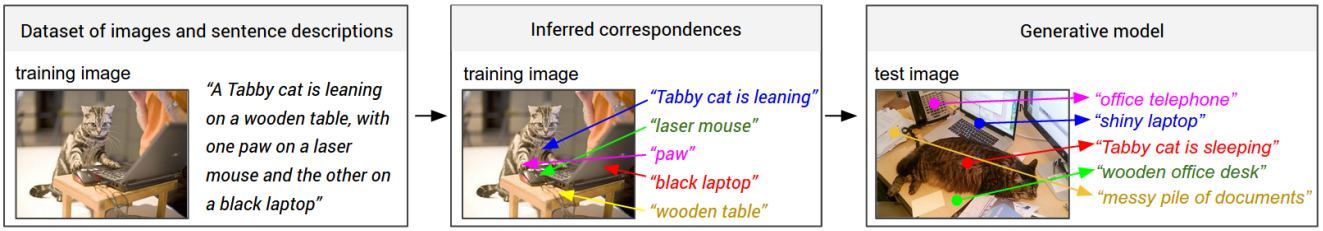


Figure 2. Overview of our approach. A dataset of images and their sentence descriptions is the input to our model (left). Our model first infers the correspondences (middle, Section 3.1) and then learns to generate novel descriptions (right, Section 3.2).

representations so that semantically similar concepts across the two modalities occupy nearby regions of the space.

3.1.1 Representing images

Following prior work [29, 24], we observe that sentence descriptions make frequent references to objects and their attributes. Thus, we follow the method of Girshick et al. [17] to detect objects in every image with a Region Convolutional Neural Network (RCNN). The CNN is pre-trained on ImageNet [6] and finetuned on the 200 classes of the ImageNet Detection Challenge [45]. Following Karpathy et al. [24], we use 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:

$$v = W_m[CNN_{\theta_c}(I_b)] + b_m, \tag{1}$$

where $CNN(I_b)$ transforms the pixels inside bounding box I_b into 4096-dimensional activations of the fully connected layer immediately before the classifier. The CNN parameters θ_c contain approximately 60 million parameters. The matrix W_m has dimensions $h \times 4096$, where h is the size of the multimodal embedding space (h ranges from 1000-1600 in our experiments). Every image is thus represented as a set of h-dimensional vectors $\{v_i \mid i = 1 \dots 20\}$.

3.1.2 Representing sentences

To establish the inter-modal relationships, we would like to represent the words in the sentence in the same h-dimensional embedding space that the image regions occupy. The simplest approach might be to project every individual word directly into this embedding. However, this approach does not consider any ordering and word context information in the sentence. An extension to this idea is to use word bigrams, or dependency tree relations as previously proposed [24]. However, this still imposes an arbitrary maximum size of the context window and requires the use of Dependency Tree Parsers that might be trained on unrelated text corpora.

To address these concerns, we propose to use a Bidirectional Recurrent Neural Network (BRNN) [46] to compute the word representations. The BRNN takes a sequence of N words (encoded in a 1-of-k representation) and transforms each one into an h-dimensional vector. However, the representation of each word is enriched by a variably-sized context around that word. Using the index $t = 1 \dots N$ to denote the position of a word in a sentence, the precise form of the BRNN is as follows:

$$x_t = W_w \mathbb{I}_t \tag{2}$$
$$e_t = f(W_e x_t + b_e) \tag{3}$$
$$h_t^f = f(e_t + W_f h_{t-1}^f + b_f) \tag{4}$$
$$h_t^b = f(e_t + W_b h_{t+1}^b + b_b) \tag{5}$$
$$s_t = f(W_d(h_t^f + h_t^b) + b_d). \tag{6}$$

Here, $\mathbb{I}_t$ 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 [41] weights and keep fixed due to overfitting concerns. However, in practice we find little change in final performance when these vectors are trained, even from random initialization. Note that the BRNN consists of two independent streams of processing, one moving left to right (h_t^f) and the other right to left (h_t^b) (see Figure 3 for diagram). The final h-dimensional representation s_t for the t-th word is a function of both the word at that location and also its surrounding context in the sentence. Technically, every s_t is a function of all words in the entire sentence, but our empirical finding is that the final word representations (s_t) align most strongly to the visual concept of the word at that location ($\mathbb{I}_t$).

We learn the parameters W_e, W_f, W_b, W_d and the respective biases b_e, b_f, b_b, b_d. A typical size of the hidden representation in our experiments ranges between 300-600 dimensions. We set the activation function f to the rectified linear unit (ReLU), which computes $f : x \mapsto max(0, x)$.

3.1.3 Alignment objective

We have described the transformations that map every image and sentence into a set of vectors in a common h-dimensional space. Since the supervision is at the level of entire images and sentences, our strategy is to formulate an

image-sentence score as a function of the individual region-word scores. Intuitively, a sentence-image pair should have a high matching score if its words have a confident support in the image. The model of Karpathy et a. [24] interprets the dot product $v_i^T s_t$ between the i-th region and t-th word as a measure of similarity and use it to define the score between image k and sentence l as:

$$S_{kl} = \sum_{t \in g_l} \sum_{i \in g_k} max(0, v_i^T s_t). \quad (7)$$

Here, g_k is the set of image fragments in image k and g_l is the set of sentence fragments in sentence l. The indices k, l range over the images and sentences in the training set. Together with their additional Multiple Instance Learning objective, this score carries the interpretation that a sentence fragment aligns to a subset of the image regions whenever the dot product is positive. We found that the following reformulation simplifies the model and alleviates the need for additional objectives and their hyperparameters:

$$S_{kl} = \sum_{t \in g_l} max_{i \in g_k} v_i^T s_t. \quad (8)$$

Here, every word s_t aligns to the single best image region. As we show in the experiments, this simplified model also leads to improvements in the final ranking performance. Assuming that $k = l$ denotes a corresponding image and sentence pair, the final max-margin, structured loss remains:

$$\mathcal{C}(\theta) = \sum_k \Big[\underbrace{\sum_l max(0, S_{kl} - S_{kk} + 1)}_{\text{rank images}} + \underbrace{\sum_l max(0, S_{lk} - S_{kk} + 1)}_{\text{rank sentences}} \Big]. \quad (9)$$

This objective encourages aligned image-sentences pairs to have a higher score than misaligned pairs, by a margin.

3.1.4 Decoding text segment alignments to images

Consider an image from the training set and its corresponding sentence. We can interpret the quantity $v_i^T s_t$ as the unnormalized log probability of the t-th word describing any of the bounding boxes in the image. However, since we are ultimately interested in generating snippets of text instead of single words, we would like to align extended, contiguous sequences of words to a single bounding box. Note that the naïve solution that assigns each word independently to the highest-scoring region is insufficient because it leads to words getting scattered inconsistently to different regions.

To address this issue, we treat the true alignments as latent variables in a Markov Random Field (MRF) where the binary interactions between neighboring words encourage an alignment to the same region. Concretely, given a sentence with N words and an image with M bounding boxes, we introduce the latent alignment variables $a_j \in \{1 \dots M\}$ for $j = 1 \dots N$ and formulate an MRF in a chain structure along the sentence as follows:

$$E(\mathbf{a}) = \sum_{j=1 \dots N} \psi_j^U(a_j) + \sum_{j=1 \dots N-1} \psi_j^B(a_j, a_{j+1}) \quad (10)$$

$$\psi_j^U(a_j = t) = v_i^T s_t \quad (11)$$

$$\psi_j^B(a_j, a_{j+1}) = \beta \mathbb{1}[a_j = a_{j+1}]. \quad (12)$$

Here, β is a hyperparameter that controls the affinity towards longer word phrases. This parameter allows us to interpolate between single-word alignments ($\beta = 0$) and aligning the entire sentence to a single, maximally scoring region when β is large. We minimize the energy to find the best alignments **a** using dynamic programming. The output of this process is a set of image regions annotated with segments of text. We now describe an approach for generating novel phrases based on these correspondences.

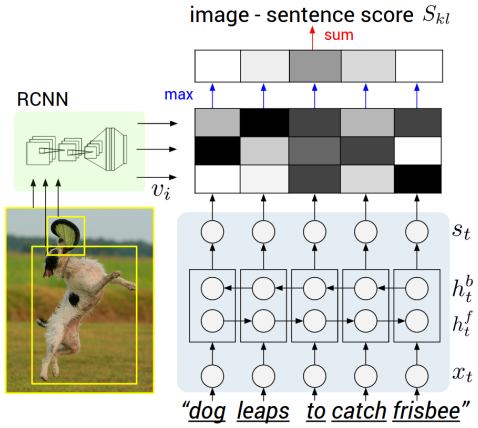


Figure 3. Diagram for evaluating the image-sentence score S_{kl}. Object regions are embedded with a CNN (left). Words (enriched by their context) are embedded in the same multimodal space with a BRNN (right). Pairwise similarities are computed with inner products (magnitudes shown in grayscale) and finally reduced to image-sentence score with Equation 8.

3.2. Multimodal Recurrent Neural Network for generating descriptions

In this section we assume an input set of images and their textual descriptions. These could be full images and their sentence descriptions, or regions and text snippets, as inferred in the previous section. The key challenge is in the design of a model that can predict a variable-sized sequence of outputs given an image. In previously developed language models based on Recurrent Neural Networks (RNNs) [40, 50, 10], this is achieved by defining a probability distribution of the next word in a sequence given the current word and context from previous time steps. We explore a simple

but effective extension that additionally conditions the generative process on the content of an input image. More formally, during training our Multimodal RNN takes the image pixels I and a sequence of input vectors $(x_1, \ldots, x_T)$. It then computes a sequence of hidden states $(h_1, \ldots, h_t)$ and a sequence of outputs $(y_1, \ldots, y_t)$ by iterating the following recurrence relation for $t = 1$ to T:

$$b_v = W_{hi}[CNN_{\theta_c}(I)] \quad (13)$$

$$h_t = f(W_{hx}x_t + W_{hh}h_{t-1} + b_h + \mathbb{1}(t = 1) \odot b_v) \quad (14)$$

$$y_t = softmax(W_{oh}h_t + b_o). \quad (15)$$

In the equations above, $W_{hi}, W_{hx}, W_{hh}, W_{oh}, x_i$ and b_h, b_o are learnable parameters, and $CNN_{\theta_c}(I)$ is the last layer of a CNN. The output vector y_t holds the (unnormalized) log probabilities of words in the dictionary and one additional dimension for a special END token. Note that we provide the image context vector b_v to the RNN only at the first iteration, which we found to work better than at each time step. In practice we also found that it can help to also pass both $b_v, (W_{hx}x_t)$ through the activation function. A typical size of the hidden layer of the RNN is 512 neurons.

RNN training. The RNN is trained to combine a word (x_t), the previous context (h_{t-1}) to predict the next word (y_t). We condition the RNN's predictions on the image information (b_v) via bias interactions on the first step. The training proceeds as follows (refer to Figure 4): We set $h_0 = \vec{0}$, x_1 to a special START vector, and the desired label y_1 as the first word in the sequence. Analogously, we set x_2 to the word vector of the first word and expect the network to predict the second word, etc. Finally, on the last step when x_T represents the last word, the target label is set to a special END token. The cost function is to maximize the log probability assigned to the target labels (i.e. Softmax classifier).

RNN at test time. To predict a sentence, we compute the image representation b_v, set $h_0 = 0$, x_1 to the START vector and compute the distribution over the first word y_1. We sample a word from the distribution (or pick the argmax), set its embedding vector as x_2, and repeat this process until the END token is generated. In practice we found that beam search (e.g. beam size 7) can improve results.

3.3. Optimization

We use SGD with mini-batches of 100 image-sentence pairs and momentum of 0.9 to optimize the alignment model. We cross-validate the learning rate and the weight decay. We also use dropout regularization in all layers except in the recurrent layers [59] and clip gradients elementwise at 5 (important). The generative RNN is more difficult to optimize, party due to the word frequency disparity between rare words and common words (e.g. "a" or the END token). We achieved the best results using RMSprop [52], which is an adaptive step size method that scales the update of each weight by a running average of its gradient norm.

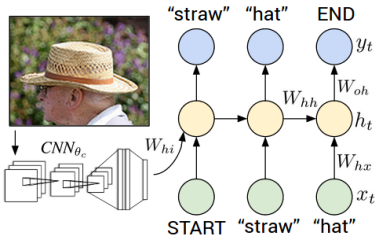


Figure 4. Diagram of our multimodal Recurrent Neural Network generative model. The RNN takes a word, the context from previous time steps and defines a distribution over the next word in the sentence. The RNN is conditioned on the image information at the first time step. START and END are special tokens.

4. Experiments

Datasets. We use the Flickr8K [21], Flickr30K [58] and MSCOCO [37] datasets in our experiments. These datasets contain 8,000, 31,000 and 123,000 images respectively and each is annotated with 5 sentences using Amazon Mechanical Turk. For Flickr8K and Flickr30K, we use 1,000 images for validation, 1,000 for testing and the rest for training (consistent with [21, 24]). For MSCOCO we use 5,000 images for both validation and testing.

Data Preprocessing. We convert all sentences to lowercase, discard non-alphanumeric characters. We filter words to those that occur at least 5 times in the training set, which results in 2538, 7414, and 8791 words for Flickr8k, Flickr30K, and MSCOCO datasets respectively.

4.1. Image-Sentence Alignment Evaluation

We first investigate the quality of the inferred text and image alignments with ranking experiments. We consider a withheld set of images and sentences and retrieve items in one modality given a query from the other by sorting based on the image-sentence score S_{kl} (Section 3.1.3). We report the median rank of the closest ground truth result in the list and Recall@K, which measures the fraction of times a correct item was found among the top K results. The result of these experiments can be found in Table 1, and example retrievals in Figure 5. We now highlight some of the takeaways.

Our full model outperforms previous work. First, our full model ("Our model: BRNN") outperforms Socher et al. [49] who trained with a similar loss but used a single image representation and a Recursive Neural Network over the sentence. A similar loss was adopted by Kiros et al. [25], who use an LSTM [20] to encode sentences. We list their performance with a CNN that is equivalent in power (AlexNet [28]) to the one used in this work, though similar to [54] they outperform our model with a more powerful CNN (VGGNet [47], GoogLeNet [51]). "DeFrag" are the results reported by Karpathy et al. [24]. Since we use different word vectors, dropout for regularization and different cross-validation ranges and larger embedding sizes, we reimplemented their loss for a fair comparison ("Our imple-

Model	Image Annotation				Image Search			
	R@1	**R@5**	**R@10**	**Med *r***	**R@1**	**R@5**	**R@10**	**Med *r***
Flickr30K								
SDT-RNN (Socher et al. [49])	9.6	29.8	41.1	16	8.9	29.8	41.1	16
Kiros et al. [25]	14.8	39.2	50.9	10	11.8	34.0	46.3	13
Mao et al. [38]	18.4	40.2	50.9	10	12.6	31.2	41.5	16
Donahue et al. [8]	17.5	40.3	50.8	9	-	-	-	-
DeFrag (Karpathy et al. [24])	14.2	37.7	51.3	10	10.2	30.8	44.2	14
Our implementation of DeFrag [24]	19.2	44.5	58.0	6.0	12.9	35.4	47.5	10.8
Our model: DepTree edges	20.0	46.6	59.4	5.4	15.0	36.5	48.2	10.4
Our model: BRNN	**22.2**	**48.2**	**61.4**	**4.8**	**15.2**	**37.7**	**50.5**	**9.2**
Vinyals et al. [54] (more powerful CNN)	23	-	63	5	17	-	57	8
MSCOCO								
Our model: 1K test images	38.4	69.9	80.5	1.0	27.4	60.2	74.8	3.0
Our model: 5K test images	16.5	39.2	52.0	9.0	10.7	29.6	42.2	14.0

Table 1. Image-Sentence ranking experiment results. **R@K** is Recall@K (high is good). **Med *r*** is the median rank (low is good). In the results for our models, we take the top 5 validation set models, evaluate each independently on the test set and then report the average performance. The standard deviations on the recall values range from approximately 0.5 to 1.0.

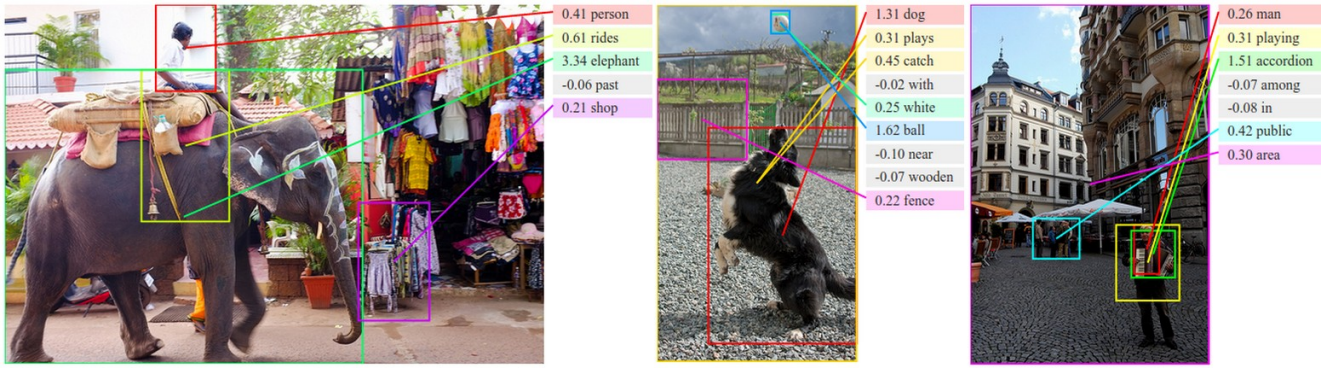


Figure 5. Example alignments predicted by our model. For every test image above, we retrieve the most compatible test sentence and visualize the highest-scoring region for each word (before MRF smoothing described in Section 3.1.4) and the associated scores ($v_i^T s_t$). We hide the alignments of low-scoring words to reduce clutter. We assign each region an arbitrary color.

mentation of DeFrag"). Compared to other work that uses AlexNets, our full model shows consistent improvement.

Our simpler cost function improves performance. We strive to better understand the source of our performance. First, we removed the BRNN and used dependency tree relations exactly as described in Karpathy et al. [24] ("Our model: DepTree edges"). The only difference between this model and "Our reimplementation of DeFrag" is the new, simpler cost function introduced in Section 3.1.3. We see that our formulation shows consistent improvements.

BRNN outperforms dependency tree relations. Furthermore, when we replace the dependency tree relations with the BRNN we observe additional performance improvements. Since the dependency relations were shown to work better than single words and bigrams [24], this suggests that the BRNN is taking advantage of contexts longer than two words. Furthermore, our method does not rely on extracting a Dependency Tree and instead uses the raw words directly.

MSCOCO results for future comparisons. We are not aware of other published ranking results on MSCOCO. Therefore, we report results on a subset of 1,000 images and the full set of 5,000 test images for future comparisons. Note that the 5000 images numbers are lower since Recall@K is a function of test set size.

Qualitative. As can be seen from example groundings in Figure 5, the model discovers interpretable visual-semantic correspondences, even for small or relatively rare objects such as an *"accordion"*. These would be likely missed by models that only reason about full images.

Learned region and word vector magnitudes. An appealing feature of our model is that it learns to modulate the magnitude of the region and word embeddings. Due to their inner product interaction, we observe that representations of visually discriminative words such as *"kayaking, pumpkins"* have embedding vectors with higher magnitudes, which in turn translates to a higher influence on the image-sentence score. Conversely, stop words such as *"now, simply, actually, but"* are mapped near the origin, which reduces their influence. See more analysis in supplementary material.

	Flickr8K				Flickr30K				MSCOCO 2014					
Model	B-1	B-2	B-3	B-4	B-1	B-2	B-3	B-4	B-1	B-2	B-3	B-4	METEOR	CIDEr
Nearest Neighbor	—	—	—	—	—	—	—	—	48.0	28.1	16.6	10.0	15.7	38.3
Mao et al. [38]	58	28	23	—	55	24	20	—	—	—	—	—	—	—
Google NIC [54]	63	41	27	—	66.3	42.3	27.7	18.3	66.6	46.1	32.9	24.6	—	—
LRCN [8]	—	—	—	—	58.8	39.1	25.1	16.5	62.8	44.2	30.4	—	—	—
MS Research [12]	—	—	—	—	—	—	—	—	—	—	—	21.1	20.7	—
Chen and Zitnick [5]	—	—	—	14.1	—	—	—	12.6	—	—	—	19.0	20.4	—
Our model	57.9	38.3	24.5	16.0	57.3	36.9	24.0	15.7	62.5	45.0	32.1	23.0	19.5	66.0

Table 2. Evaluation of full image predictions on 1,000 test images. **B-n** is BLEU score that uses up to n-grams. High is good in all columns. For future comparisons, our METEOR/CIDEr Flickr8K scores are 16.7/31.8 and the Flickr30K scores are 15.3/24.7.

Figure 6. Example sentences generated by the multimodal RNN for test images. We provide many more examples on our project page.

4.2. Generated Descriptions: Fulframe evaluation

We now evaluate the ability of our RNN model to describe images and regions. We first trained our Multimodal RNN to generate sentences on full images with the goal of verifying that the model is rich enough to support the mapping from image data to sequences of words. For these full image experiments we use the more powerful VGGNet image features [47]. We report the BLEU [44], METEOR [7] and CIDEr [53] scores computed with the `coco-caption` code [4] [2]. Each method evaluates a *candidate* sentence by measuring how well it matches a set of five *reference* sentences written by humans.

Qualitative. The model generates sensible descriptions of images (see Figure 6), although we consider the last two images failure cases. The first prediction *"man in black shirt is playing a guitar"* does not appear in the training set. However, there are 20 occurrences of "man in black shirt" and 60 occurrences of "is paying guitar", which the model may have composed to describe the first image. In general, we find that a relatively large portion of generated sentences (60% with beam size 7) can be found in the training data. This fraction decreases with lower beam size; For instance, with beam size 1 this falls to 25%, but the performance also deteriorates (e.g. from 0.66 to 0.61 CIDEr).

Multimodal RNN outperforms retrieval baseline. Our first comparison is to a nearest neighbor retrieval baseline. Here, we annotate each test image with a sentence of the most similar training set image as determined by L2 norm over VGGNet [47] fc7 features. Table 2 shows that the Multimodal RNN confidently outperforms this retrieval method. Hence, even with 113,000 train set images in MSCOCO the retrieval approach is inadequate. Additionally, the RNN takes only a fraction of a second to evaluate per image.

Comparison to other work. Several related models have been proposed in Arxiv preprints since the original submission of this work. We also include these in Table 2 for comparison. Most similar to our model is Vinyals et al. [54]. Unlike this work where the image information is communicated through a bias term on the first step, they incorporate it as a first word, they use a more powerful but more complex sequence learner (LSTM [20]), a different CNN (GoogLeNet [51]), and report results of a model ensemble. Donahue et al. [8] use a 2-layer factored LSTM (similar in structure to the RNN in Mao et al. [38]). Both models appear to work worse than ours, but this is likely in large part due to their use of the less powerful AlexNet [28] features. Compared to these approaches, our model prioritizes simplicity and speed at a slight cost in performance.

4.3. Generated Descriptions: Region evaluation

We now train the Multimodal RNN on the correspondences between image regions and snippets of text, as inferred by the alignment model. To support the evaluation, we used Amazon Mechanical Turk (AMT) to collect a new dataset

[2] `https://github.com/tylin/coco-caption`

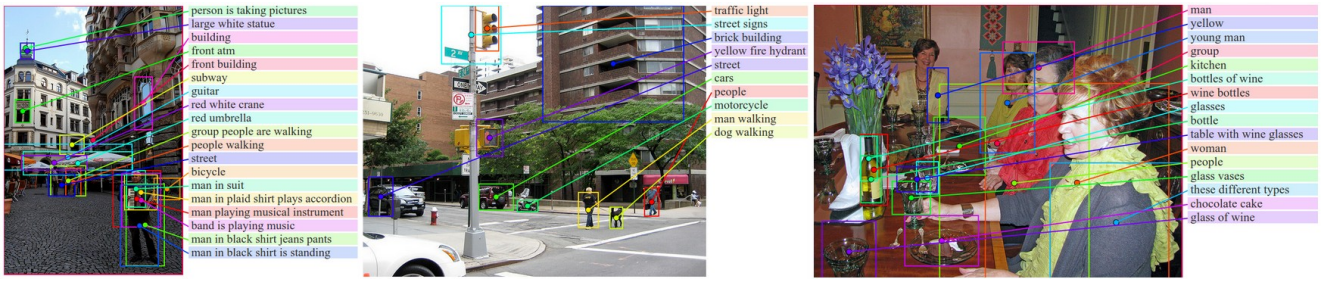


Figure 7. Example region predictions. We use our region-level multimodal RNN to generate text (shown on the right of each image) for some of the bounding boxes in each image. The lines are grounded to centers of bounding boxes and the colors are chosen arbitrarily.

of region-level annotations that we only use at test time. The labeling interface displayed a single image and asked annotators (we used nine per image) to draw five bounding boxes and annotate each with text. In total, we collected 9,000 text snippets for 200 images in our MSCOCO test split (i.e. 45 snippets per image). The snippets have an average length of 2.3 words. Example annotations include *"sports car", "elderly couple sitting", "construction site", "three dogs on leashes", "chocolate cake"*. We noticed that asking annotators for grounded text snippets induces language statistics different from those in full image captions. Our region annotations are more comprehensive and feature elements of scenes that would rarely be considered salient enough to be included in a single sentence sentence about the full image, such as *"heating vent", "belt buckle", and "chimney"*.

Qualitative. We show example region model predictions in Figure 7. To reiterate the difficulty of the task, consider for example the phrase *"table with wine glasses"* that is generated on the image on the right in Figure 7. This phrase only occurs in the training set 30 times. Each time it may have a different appearance and each time it may occupy a few (or none) of our object bounding boxes. To generate this string for the region, the model had to first correctly learn to ground the string and then also learn to generate it.

Region model outperforms full frame model and ranking baseline. Similar to the full image description task, we evaluate this data as a prediction task from a 2D array of pixels (one image region) to a sequence of words and record the BLEU score. The ranking baseline retrieves training sentence substrings most compatible with each region as judged by the BRNN model. Table 3 shows that the region RNN model produces descriptions most consistent with our collected data. Note that the fullframe model was trained only on full images, so feeding it smaller image regions deteriorates its performance. However, its sentences are also longer than the region model sentences, which likely negatively impacts the BLEU score. The sentence length is non-trivial to control for with an RNN, but we note that the region model also outperforms the fullframe model on all other metrics: CIDEr 61.6/20.3, METEOR 15.8/13.3, ROUGE 35.1/21.0 for region/fullframe respectively.

Model	B-1	B-2	B-3	B-4
Human agreement	61.5	45.2	30.1	22.0
Nearest Neighbor	22.9	10.5	0.0	0.0
RNN: Fullframe model	14.2	6.0	2.2	0.0
RNN: Region level model	**35.2**	**23.0**	**16.1**	**14.8**

Table 3. BLEU score evaluation of image region annotations.

4.4. Limitations

Although our results are encouraging, the Multimodal RNN model is subject to multiple limitations. First, the model can only generate a description of one input array of pixels at a fixed resolution. A more sensible approach might be to use multiple saccades around the image to identify all entities, their mutual interactions and wider context before generating a description. Additionally, the RNN receives the image information only through additive bias interactions, which are known to be less expressive than more complicated multiplicative interactions [50, 20]. Lastly, our approach consists of two separate models. Going directly from an image-sentence dataset to region-level annotations as part of a single model trained end-to-end remains an open problem.

5. Conclusions

We introduced a model that generates natural language descriptions of image regions based on weak labels in form of a dataset of images and sentences, and with very few hard-coded assumptions. Our approach features a novel ranking model that aligned parts of visual and language modalities through a common, multimodal embedding. We showed that this model provides state of the art performance on image-sentence ranking experiments. Second, we described a Multimodal Recurrent Neural Network architecture that generates descriptions of visual data. We evaluated its performance on both fullframe and region-level experiments and showed that in both cases the Multimodal RNN outperforms retrieval baselines.

Acknowledgements.
We thank Justin Johnson and Jon Krause for helpful comments and discussions. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the GPUs used for this research. This research is partially supported by an ONR MURI grant, and NSF ISS-1115313.

References

[1] A. Barbu, A. Bridge, Z. Burchill, D. Coroian, S. Dickinson, S. Fidler, A. Michaux, S. Mussman, S. Narayanaswamy, D. Salvi, et al. Video in sentences out. *arXiv preprint arXiv:1204.2742*, 2012.

[2] K. Barnard, P. Duygulu, D. Forsyth, N. De Freitas, D. M. Blei, and M. I. Jordan. Matching words and pictures. *JMLR*, 2003.

[3] Y. Bengio, H. Schwenk, J.-S. Senécal, F. Morin, and J.-L. Gauvain. Neural probabilistic language models. In *Innovations in Machine Learning*. Springer, 2006.

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

[5] X. Chen and C. L. Zitnick. Learning a recurrent visual representation for image caption generation. *CoRR*, abs/1411.5654, 2014.

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

[7] M. Denkowski and A. Lavie. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*, 2014.

[8] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. *arXiv preprint arXiv:1411.4389*, 2014.

[9] D. Elliott and F. Keller. Image description using visual dependency representations. In *EMNLP*, pages 1292–1302, 2013.

[10] J. L. Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[11] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.

[12] H. Fang, S. Gupta, F. Iandola, R. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. Platt, et al. From captions to visual concepts and back. *arXiv preprint arXiv:1411.4952*, 2014.

[13] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences from images. In *ECCV*. 2010.

[14] L. Fei-Fei, A. Iyer, C. Koch, and P. Perona. What do we perceive in a glance of a real-world scene? *Journal of vision*, 7(1):10, 2007.

[15] S. Fidler, A. Sharma, and R. Urtasun. A sentence is worth a thousand pixels. In *CVPR*, 2013.

[16] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov, et al. Devise: A deep visual-semantic embedding model. In *NIPS*, 2013.

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

[18] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1–8. IEEE, 2009.

[19] A. Gupta and P. Mannem. From image annotation to image description. In *Neural information processing*. Springer, 2012.

[20] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[21] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 2013.

[22] R. JeffreyPennington and C. Manning. Glove: Global vectors for word representation. 2014.

[23] Y. Jia, M. Salzmann, and T. Darrell. Learning cross-modality similarity for multinomial data. In *ICCV*, 2011.

[24] A. Karpathy, A. Joulin, and L. Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. *arXiv preprint arXiv:1406.5679*, 2014.

[25] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*, 2014.

[26] R. Kiros, R. S. Zemel, and R. Salakhutdinov. Multimodal neural language models. *ICML*, 2014.

[27] C. Kong, D. Lin, M. Bansal, R. Urtasun, and S. Fidler. What are you talking about? text-to-image coreference. In *CVPR*, 2014.

[28] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[29] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Baby talk: Understanding and generating simple image descriptions. In *CVPR*, 2011.

[30] P. Kuznetsova, V. Ordonez, A. C. Berg, T. L. Berg, and Y. Choi. Collective generation of natural image descriptions. In *ACL*, 2012.

[31] P. Kuznetsova, V. Ordonez, T. L. Berg, U. C. Hill, and Y. Choi. Treetalk: Composition and compression of trees for image descriptions. *Transactions of the Association for Computational Linguistics*, 2(10):351–362, 2014.

[32] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[33] L.-J. Li and L. Fei-Fei. What, where and who? classifying events by scene and object recognition. In *ICCV*, 2007.

[34] L.-J. Li, R. Socher, and L. Fei-Fei. Towards total scene understanding: Classification, annotation and segmentation in an automatic framework. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2036–2043. IEEE, 2009.

[35] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams. In *CoNLL*, 2011.

[36] D. Lin, S. Fidler, C. Kong, and R. Urtasun. Visual semantic search: Retrieving videos via complex textual queries. 2014.

[37] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. *arXiv preprint arXiv:1405.0312*, 2014.

[38] J. Mao, W. Xu, Y. Yang, J. Wang, and A. L. Yuille. Explain images with multimodal recurrent neural networks. *arXiv preprint arXiv:1410.1090*, 2014.

[39] C. Matuszek*, N. FitzGerald*, L. Zettlemoyer, L. Bo, and D. Fox. A Joint Model of Language and Perception for Grounded Attribute Learning. In *Proc. of the 2012 International Conference on Machine Learning*, Edinburgh, Scotland, June 2012.

[40] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur. Recurrent neural network based language model. In *INTERSPEECH*, 2010.

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

[42] M. Mitchell, X. Han, J. Dodge, A. Mensch, A. Goyal, A. Berg, K. Yamaguchi, T. Berg, K. Stratos, and H. Daumé, III. Midge: Generating image descriptions from computer vision detections. In *EACL*, 2012.

[43] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*, 2011.

[44] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

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

[46] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *Signal Processing, IEEE Transactions on*, 1997.

[47] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[48] R. Socher and L. Fei-Fei. Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora. In *CVPR*, 2010.

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

[50] I. Sutskever, J. Martens, and G. E. Hinton. Generating text with recurrent neural networks. In *ICML*, 2011.

[51] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*, 2014.

[52] T. Tieleman and G. E. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude., 2012.

[53] R. Vedantam, C. L. Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. *CoRR*, abs/1411.5726, 2014.

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

[55] Y. Yang, C. L. Teo, H. Daumé III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *EMNLP*, 2011.

[56] B. Z. Yao, X. Yang, L. Lin, M. W. Lee, and S.-C. Zhu. I2t: Image parsing to text description. *Proceedings of the IEEE*, 98(8):1485–1508, 2010.

[57] M. Yatskar, L. Vanderwende, and L. Zettlemoyer. See no evil, say no evil: Description generation from densely labeled images. *Lexical and Computational Semantics*, 2014.

[58] 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.

[59] W. Zaremba, I. Sutskever, and O. Vinyals. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*, 2014.

[60] C. L. Zitnick, D. Parikh, and L. Vanderwende. Learning the visual interpretation of sentences. *ICCV*, 2013.

6. Supplementary Material

6.1. Magnitude modulation

An appealing feature of our alignment model is that it learns to modulate the importance of words and regions by scaling the magnitude of their corresponding embedding vectors. To see this, recall that we compute the image-sentence similarity between image k and sentence l as follows:

$$S_{kl} = \sum_{t \in g_l} max_{i \in g_k} v_i^T s_t. \quad (16)$$

Disciminative words. As a result of this formulation, we observe that representations of visually discriminative words such as *"kayaking, pumpkins"* tend to have higher magnitude in the embedding space, which translates to a higher influence on the final image-sentence scores due to the inner product. Conversely, the model learns to map stop words such as *"now, simply, actually, but"* near the origin, which reduces their influence. Table 4 show the top 40 words with highest and lowest magnitudes $\|s_t\|$.

Disciminative regions. Similarly, image regions that contain discriminative entities are assigned vectors of higher magnitudes by our model. This can be be interpreted as a measure of visual saliency, since these regions would produced large scores if their textual description was present in a corresponding sentence. We show the regions with high magnitudes in Figure 8. Notice the common occurrence of often described regions such as balls, bikes, helmets.

Figure 8. Flickr30K test set regions with high vector magnitude.

Magnitude	Word	Magnitude	Word
0.42	now	2.61	kayaking
0.42	simply	2.59	trampoline
0.43	actually	2.59	pumpkins
0.44	but	2.58	windsurfing
0.44	neither	2.56	wakeboard
0.45	then	2.54	acrobatics
0.45	still	2.54	sousaphone
0.46	obviously	2.54	skydivers
0.47	that	2.52	wakeboarders
0.47	which	2.52	skateboard
0.47	felt	2.51	snowboarder
0.47	not	2.51	wakeboarder
0.47	might	2.50	skydiving
0.47	because	2.50	guitar
0.48	appeared	2.50	snowboard
0.48	therefore	2.48	kitchen
0.48	been	2.48	paraglider
0.48	if	2.48	ollie
0.48	also	2.47	firetruck
0.48	only	2.47	gymnastics
0.48	so	2.46	waterfalls
0.49	would	2.46	motorboat
0.49	yet	2.46	fryer
0.50	be	2.46	skateboarding
0.50	had	2.46	dulcimer
0.50	revealed	2.46	waterfall
0.50	never	2.46	backflips
0.50	very	2.46	unicyclist
0.50	without	2.45	kayak
0.51	they	2.43	costumes
0.51	either	2.43	wakeboarding
0.51	could	2.43	trike
0.51	feel	2.42	dancers
0.51	otherwise	2.42	cupcakes
0.51	when	2.42	tuba
0.51	already	2.42	skijoring
0.51	being	2.41	firewood
0.51	else	2.41	elevators
0.52	just	2.40	cranes
0.52	ones	2.40	bassoon

Table 4. This table shows the top magnitudes of vectors ($\|s_t\|$) for words in Flickr30K. Since the magnitude of individual words in our model is also a function of their surrounding context in the sentence, we report the average magnitude.

6.2. Alignment model

Learned appearance of text snippets. We can query our alignment model with a piece of text and retrieve individual image regions that have the highest score with that snippet. We show examples of such queries in Figure 9 and Figure 10. Notice that the model is sensitive to compound words and modifiers. For example, *"red bus"* and *"yellow bus"* give very different results. Similarly, *"bird flying in the sky"* and *"bird on a tree branch"* give different results. Additionally, it can be seen that the quality of the results deteriorates for less frequently occurring concepts, such as *"roof"* or *"straw hat"*. However, we emphasize that the model learned these visual appearances of text snippets from raw data of full images and sentences, without any explicit correspondences.

Additional alignment visualizations. See additional examples of inferred alignments between image regions and words in Figure 11. Note that one limitation of our model is that it does not explicitly handle or support counting. For instance, the last example we show contains the phrase *"three people"*. These words should align to the three people in the image, but our model puts the bounding box around two of the people. In doing so, the model may be taking advantage of the BRNN structure to modify the "people" vector to preferentially align to regions that contain multiple people. However, this is still unsatisfying because such spurious detections only exist as a result of an error in the RCNN inference process, which presumably failed to localize the individual people.

Web demo. We have published a web demo that displays our alignments for all images in the test set [3].

Additional Flickr8K experiments. We omitted ranking experiment results from our paper due to space constraints, but these can be found in Table 5

Counting. We experimented with losses that perform probabilistic inference in the forward pass that explicitly tried to localize exactly three distinct people in the image. However, this worked poorly because while the RCNN is good at finding people, it is not very good at localizing them. For instance, a single person can easily yield multiple detections (the head, the torso, or the full body, for example). We were not able to come up with a simple approach to collapsing these into a single detection (non-maxim suppression by itself was not sufficient in our experiments). Note that this ambiguity is partly an artifact of the training data. For example, torsos of people can often be labeled alone if the body is occluded. We are therefore lead to believe that this additional modeling step is highly non-trivial and a worthy subject of future work.

[3] http://cs.stanford.edu/people/karpathy/deepimagesent/rankingdemo/

Plug and play use of Natural Language Processing toolkits. Before adopting the BRNN approach, we also tried to use Natural Language Processing toolkits to process the input sentences into graphs of noun phrases and their binary relations. For instance, in the sentence *"a brown dog is chasing a young child"*, the toolkit would infer that there are two noun phrases (*"a brown dog"*, *"young child"*), joined by a binary relationship of *"chasing"*. We then developed a CRF that inferred the grounding of these noun phrases to the detection bounding boxes in the image with a unary appearance model and a spatial binary model. However, this endeavor proved fruitless. First, performing CRF-like inference during the forward pass of a Neural Network proved to be extremely slow. Second, we found that there is surprisingly little information in the relative spatial positions between bounding boxes. For instance, almost any two bounding boxes in the image could correspond to the action of *"chasing"* due to huge amount of possibly camera views of a scene. Hence, we were unable to extract enough signal from the binary relations in the coordinate system of the image and suspect that more complex 3-dimensional reasoning may be required. Lastly, we found that NLP tools (when used out of the box) introduce a large amount of mistakes in the extracted parse trees, dependency trees and parts of speech tags. We tried to fix these with complex rules and exceptions, but ultimately decided to abandon the idea. We believe that part of the problem is that these tools are usually trained on different text corpora (e.g. news articles), so image captions are outside of their domain of competence. In our experience, adopting the BRNN model instead of this approach provided immediate performance improvements and produced significant reductions in code complexity.

6.3. Additional examples: Image annotation

Additional examples of generated captions on the full image level can be found in Figure 12 (and our website). The model often gets the right gist of the scene, but sometimes guesses specific fine-grained words incorrectly. We expect that reasoning not only the global level of the image but also on the level of objects will significantly improve these results. We find the last example (*"woman in bikini is jumping over hurdle"*) to be especially illuminating. This sentence does not occur in the training data. Our general qualitative impression of the model is that it learns certain templates, e.g. *"<noun>in <noun>is <verb>in <noun>"*, and then fills these in based on textures in the image. In this particular case, the volleyball net has the visual appearance of a hurdle, which may have caused the model to insert it as a noun (along with the woman) into one of its learned sentence templates.

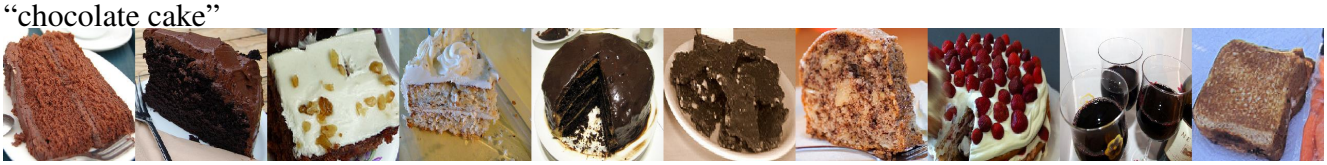
6.4. Additional examples: Region annotation

Additional examples of region annotations can be found in Figure 13. Note that we annotate regions based on the content of each image region alone, which can cause erroneous predictions when not enough context is available in the bounding box (e.g. a generated description that says "container" detected on the back of a dog's head in the image on the right, in the second row). We found that one effective way of using the contextual information and improving the predictions is to concatenate the fullframe feature CNN vector to the vector of the region of interest, giving 8192-dimensional input vector the to RNN. However, we chose to omit these experiments in our paper to preserve the simplicity of the mode, and because we believe that cleaner and more principled approaches to this challenge can be developed.

6.5. Training the Multimodal RNN

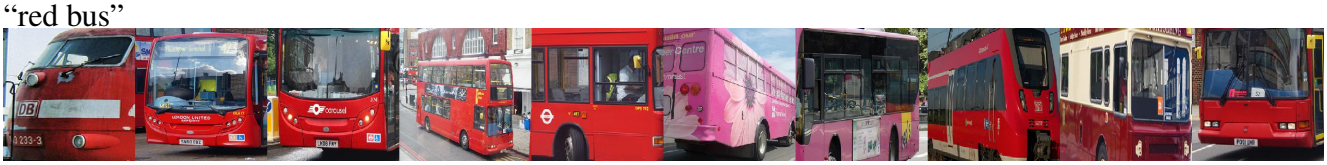
There are a few tricks needed to get the Multimodal RNN to train efficiently. We found that **clipping the gradients** (we only experimented with simple per-element clipping) at an appropriate value consistently gave better results and helped on the validation data. As mentioned in our paper, we experimented with SGD, SGD+Momentum, Adadelta, Adagrad, but found **RMSProp** to give best results. However, some SGD checkpoints usually also converged to nearby validation performance vicinity. Moreover, the distribution of the words in English language are highly non-uniform. Therefore, the model spends the first few iterations mostly learning the biases for the Softmax classifier such that it is predicting every word at random with the appropriate dataset frequency. We found that we could obtain faster convergence early in the training (and nicer loss curves) by explicitly **initializing the biases** of all words in the dictionary (in the Softmax classifier) to log probability of their occurrence in the training data. Therefore, with small weights and biases set appropriately the model right away predicts word at random according to their chance distribution. After submission of our original paper we performed additional experiments with comparing an RNN to an LSTM and found that **LSTMs** consistently produced better results, but took longer to train. Lastly, we initially used word2vec vectors as our word representations x_i, but found that it was sufficient to train these vectors from random initialization without changes in the final performance. Moreover, we found that the word2vec vectors have some unappealing properties when used in multimodal language-visual tasks. For example, all colors (e.g. red, blue, green) are clustered nearby in the word2vec representation because they are relatively interchangeable in most language contexts. However, their visual instantiations are very different.

“chocolate cake”



“glass of wine”

“red bus”



“yellow bus”

“closeup of zebra”

“sprinkled donut”

“wooden chair”

“wooden office desk”

“shiny laptop”

Figure 9. Examples of highest scoring regions for queried snippets of text, on 5,000 images of our MSCOCO test set.

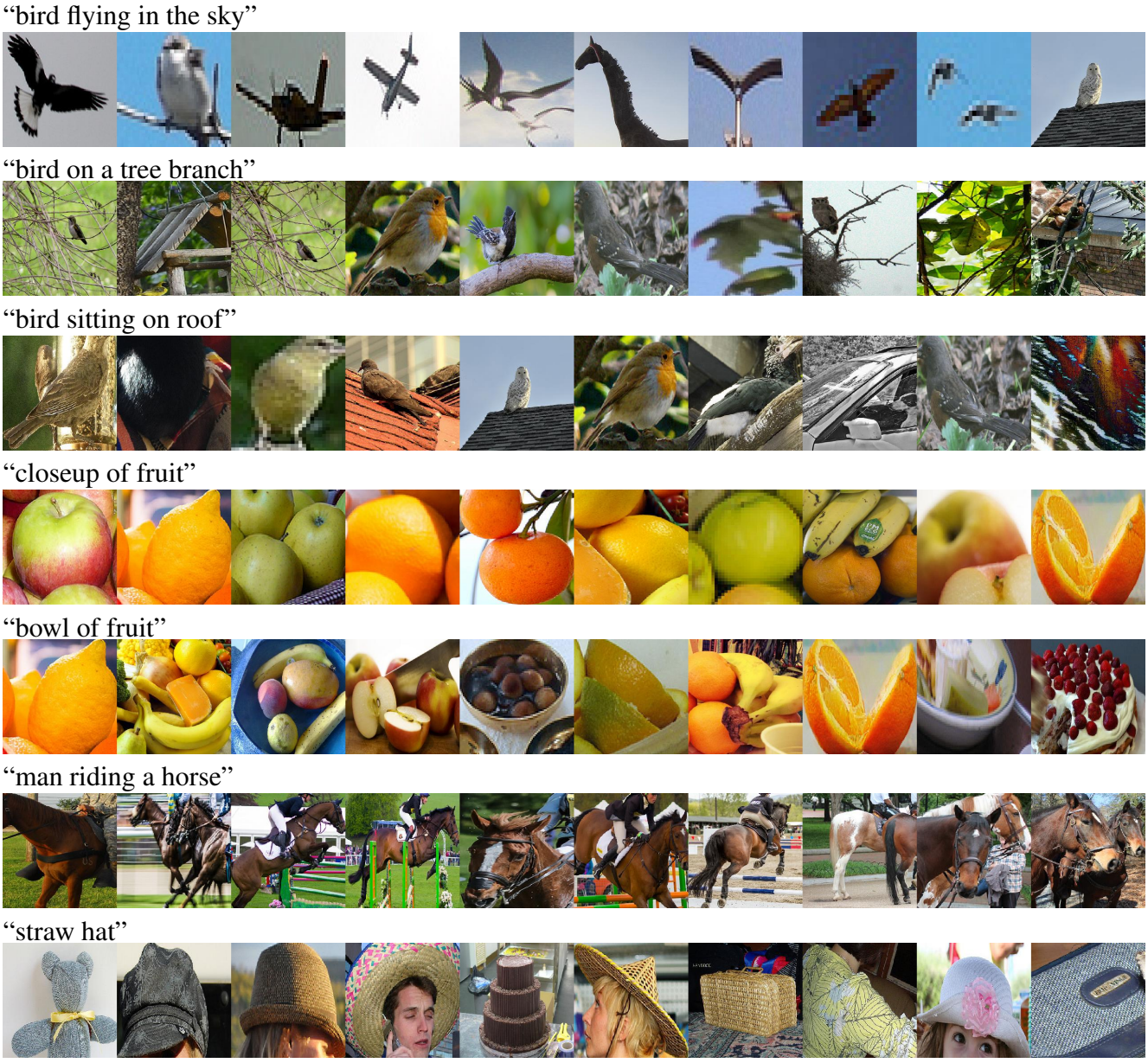


Figure 10. Examples of highest scoring regions for queried snippets of text, on 5,000 images of our MSCOCO test set.

Model	Image Annotation R@1	R@5	R@10	Med r	Image Search R@1	R@5	R@10	Med r
Flickr8K								
DeViSE (Frome et al. [16])	4.5	18.1	29.2	26	6.7	21.9	32.7	25
SDT-RNN (Socher et al. [49])	9.6	29.8	41.1	16	8.9	29.8	41.1	16
Kiros et al. [25]	13.5	36.2	45.7	13	10.4	31.0	43.7	14
Mao et al. [38]	14.5	37.2	48.5	11	11.5	31.0	42.4	15
DeFrag (Karpathy et al. [24])	12.6	32.9	44.0	14	9.7	29.6	42.5	15
Our implementation of DeFrag [24]	13.8	35.8	48.2	10.4	9.5	28.2	40.3	15.6
Our model: DepTree edges	14.8	37.9	50.0	9.4	11.6	31.4	43.8	13.2
Our model: BRNN	**16.5**	**40.6**	**54.2**	**7.6**	**11.8**	**32.1**	**44.7**	**12.4**

Table 5. Ranking experiment results for the Flickr8K dataset.

Figure 11. Additional examples of alignments. For each query test image above we retrieve the most compatible sentence from the test set and show the alignments.

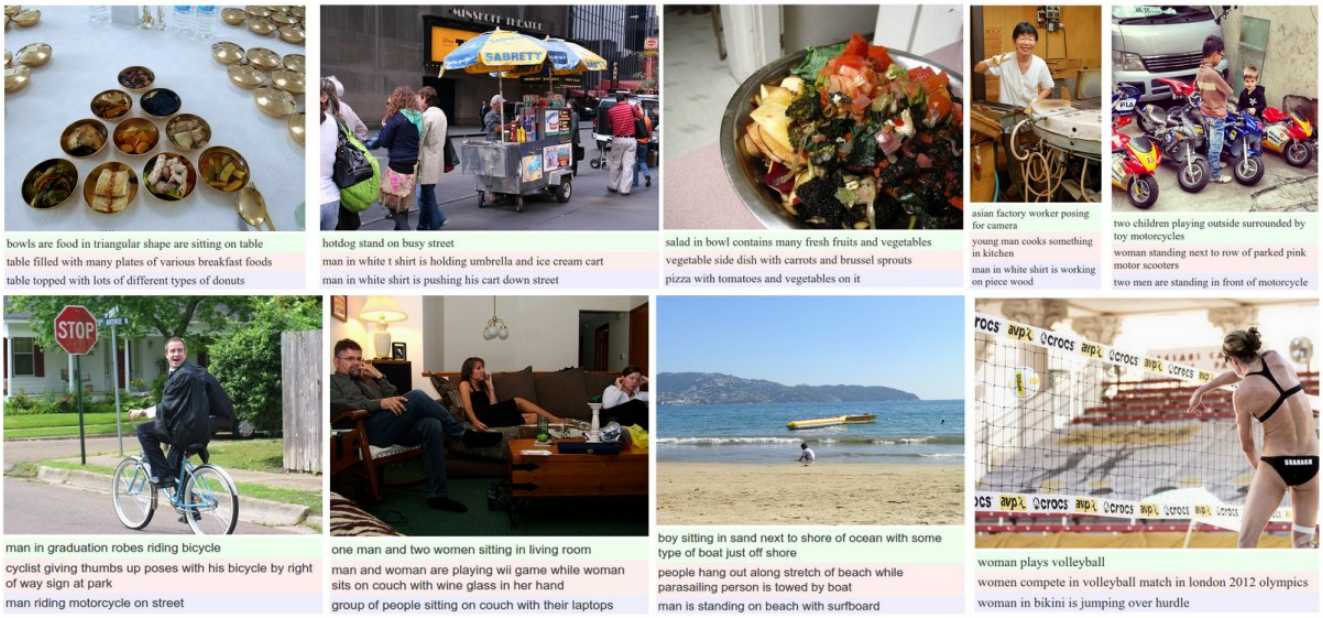


Figure 12. Additional examples of captions on the level of full images. Green: Human ground truth. Red: Top-scoring sentence from training set. Blue: Generated sentence.

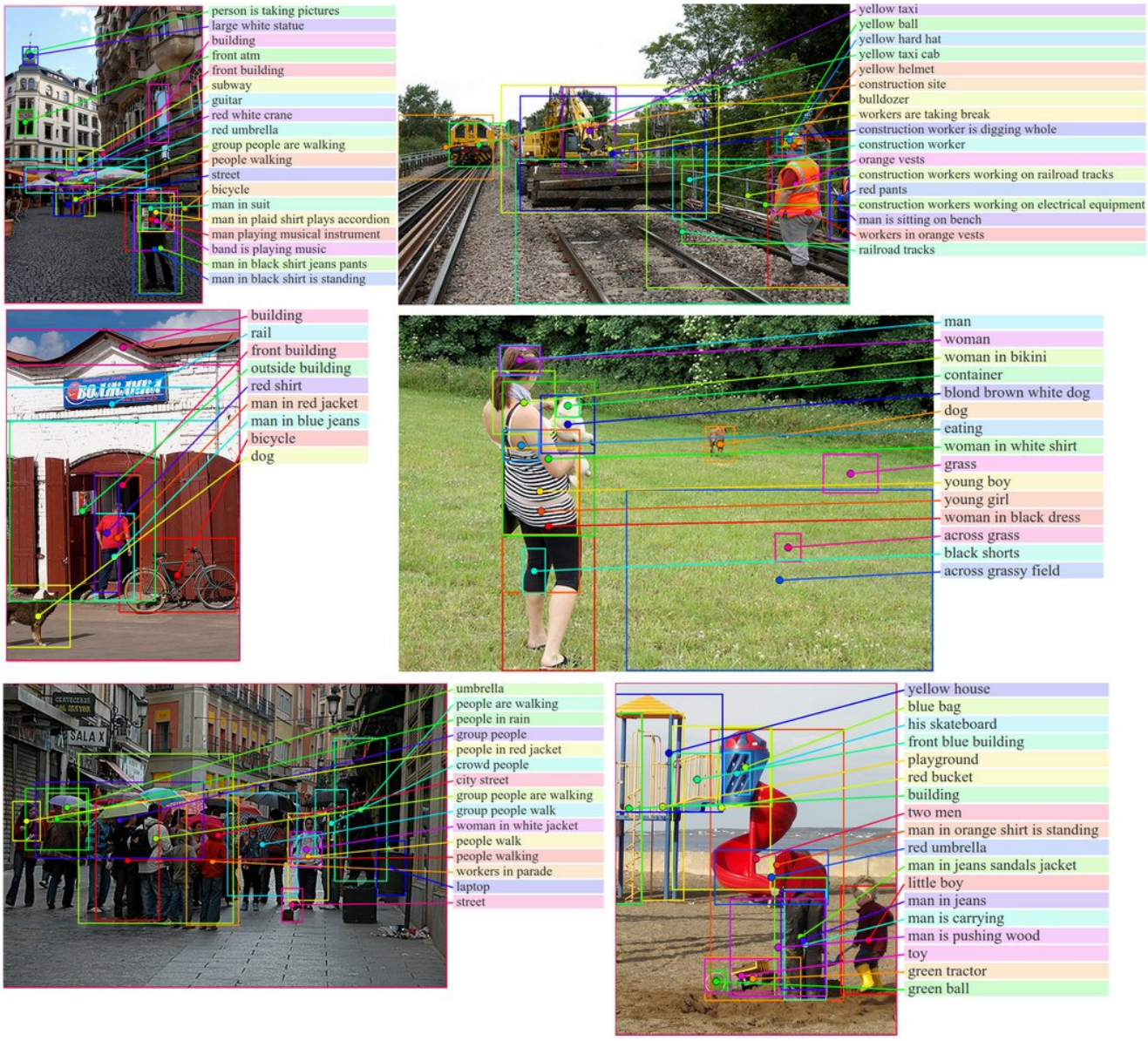


Figure 13. Additional examples of region captions on the test set of Flickr30K.