From Captions to Visual Concepts and Back

Hao Fang* Saurabh Gupta* Forrest Iandola* Rupesh Srivastava*
Li Deng Piotr Dollár† Jianfeng Gao Xiaodong He
Margaret Mitchell John C. Platt C. Lawrence Zitnick Geoffrey Zweig

Microsoft Research

Abstract

This paper presents a novel approach for automatically generating image descriptions: visual detectors and language models learn directly from a dataset of image captions. We use Multiple Instance Learning to train visual detectors for words that commonly occur in captions, including many different parts of speech such as nouns, verbs, and adjectives. The word detector outputs serve as conditional inputs to a maximum-entropy language model. The language model learns from a set of over 400,000 image descriptions to capture the statistics of word usage. We capture global semantics by re-ranking caption candidates using sentence-level features and a deep multimodal similarity model. When human judges compare the system captions to ones written by other people, the system captions have equal or better quality over 23% of the time.

1. Introduction

When does a machine "understand" an image? One definition is when it can generate a novel caption that summarizes the salient content within an image. This content may include objects that are present, their attributes, or relations between objects. Determining the salient content requires not only knowing the contents of an image, but also deducing which aspects of the scene may be interesting or novel through commonsense knowledge [50, 4, 7].

This paper describes a novel approach for generating image captions from samples. We train our caption generator from a data set of images and corresponding image descriptions. Previous approaches to generating image captions relied on object, attribute, and relation detectors learned from separate hand-labeled training data [46, 22].

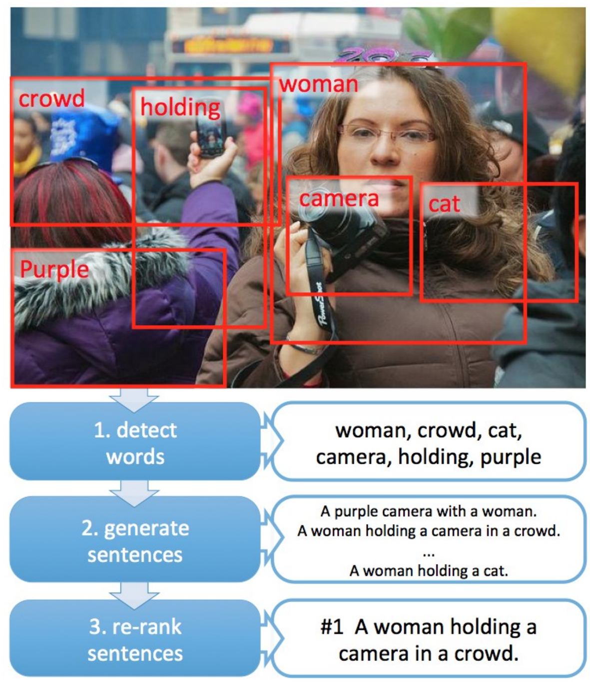

Figure 1. An illustrative example of our approach's pipeline.

The direct use of captions in training has two distinct advantages. First, captions only contain information that is inherently salient. For example, a dog detector trained from images with captions containing the word "dog" will be biased towards detecting dogs that are salient and not those that are in the background. Image descriptions also contain a wide variety of words, including nouns, verbs, and adjectives. As a result, we learn detectors that relate to a wide variety of concepts. While some concepts, such as "riding" or "beautiful", may be difficult to learn in the abstract, these terms may be highly correlated to specific visual patterns (such as a person on a horse or mountains at sunset).

Second, training a language model (LM) on image cap-

*H. Fang, S. Gupta, F. Iandola and R. Srivastava contributed equally to this work while doing internships at Microsoft Research. Current affiliations are H. Fang: University of Washington; S. Gupta and F. Iandola: University of California at Berkeley; R. Srivastava: IDSIA, USI-SUPSI.

†P. Dollár is currently at Facebook AI Research.

tions captures commonsense knowledge about a scene. A language model can learn that a person is more likely to sit on a chair than to stand on it. This information disambiguates noisy visual detections.

The pipeline of our approach is shown in Figure 1. First, we use weakly-supervised learning to create detectors for a set of words commonly found in image captions. Learning directly from image captions is difficult, because the system does not have access to supervision signals, such as object bounding boxes, that are found in other data sets [10, 5]. Many words, e.g., "crowded" or "inside", do not even have well-defined bounding boxes. To overcome this difficulty, we use three ideas. First, the system finds image regions using object proposal generators [49, 45]. Next, we featurize each of these regions using pre-trained ImageNet convolutional neural network (CNN) features [21]. Finally, we map the features of each region to a set of words likely to be contained in the caption. We train this map using Multiple Instance Learning (MIL) [28, 48, 43] to determine the image regions that are most informative of each word.

Generating novel image descriptions from a bag of likely words requires an effective LM. In this paper, we view caption generation as an optimization problem. In this view, the core task is to take the set of word detection scores, and find the highest likelihood sentence that covers each word exactly once. We train a maximum entropy (ME) LM from a set of training image descriptions [46]. This training captures commonsense knowledge about the world through language statistics [3]. An explicit search over word sequences is effective at finding a set of high-likelihood sentences.

The final stage of the system (Figure 1) re-ranks a set of high-likelihood sentences by a linear weighting of sentence features. These weights are learned using Minimum Error Rate Training (MERT) [33]. In addition to several common sentence features, we introduce a new feature based on a Deep Multimodal Similarity Model (DMSM). The DMSM learns two neural networks that map images and text fragments to a common vector representation in which the similarity between sentences and images can be easily measured. As we demonstrate, the use of the DMSM significantly improves the selection of quality sentences.

To evaluate the quality of our automatic captions, we use three easily computable metrics and *better/worse/equal* comparisons by human subjects on Amazon's Mechanical Turk (AMT). The evaluation was performed on the challenging Microsoft COCO dataset [26] containing complex images with multiple objects. Each of the 82,783 training images has 5 human annotated captions. For measuring the quality of our sentences we use the popular BLEU [36], METEOR [1] and perplexity (PPLX) metrics. Surprisingly, we find our generated captions outperform humans based on the BLEU metric. When evaluated by human subjects, the subjects found our captions of the same quality or better than humans 23% of the time. We also compare to previous work on the PASCAL sentence dataset [37]. Our results demonstrate the utility of training both visual detectors and LMs directly on image captions, as well as using a global semantic model for re-ranking the caption candidates.

2. Related Work

There are two well-studied approaches to automatic image captioning: retrieval of existing human-written captions, and generation of novel captions. Recent retrieval-based approaches have used neural networks to map images and text into a common vector representation [12, 44, 42]. Other retrieval based methods use similarity metrics that take pre-defined image features [15, 35]. Farhadi et al. [11] represents both images and text as linguistically-motivated semantic triples, and computes similarity in that space. A similar fine-grained analysis of sentences and images is done for retrieval in the context of neural networks [19].

Retrieval-based methods always return well-formed human-written captions, but these captions may not be able to describe new combinations of objects or novel scenes. This limitation has motivated a large body of work on generative approaches, where the image is first analyzed and objects are detected, and then a novel caption is generated. Previous work utilizes syntactic and semantic constraints in the generation process [30, 47, 25, 23, 22, 46], and we compare against prior state of the art in this line of work. We focus on the Midge system [30], which combines syntactic structures using maximum likelihood estimation to generate novel sentences; and compare qualitatively against the Baby Talk system [22], which generates descriptions by filling sentence template slots with words selected from a conditional random field that predicts the most likely image labeling. Both of these previous systems use the same set of test sentences, making direct comparison possible.

Recently, researchers explored purely statistical approaches to guiding language model from images. Kiros et al. [20] use a log-bilinear model with bias features derived from the image to model text conditional on the image. Kiros' model both ranks and generates the captions via sampling. Mao et al. [27] explore the use of an image-conditional recurrent neural network. Neither of these approaches use an explicit initial object detection step.

Similarly to the last two approaches, our work focuses on language generation guided by image content. Unlike these approaches, we detect words from images by applying a CNN to image regions [13, 21] and integrating the information with MIL [48]. We also minimize *a priori* assumptions about how sentences should be structured by training directly from captions. Finally, in contrast to [20, 27], we formulate the problem of generation as an optimization problem and search for the most likely sentence [39].

Figure 2. Multiple Instance Learning detections for cat, baseball, red, looking and flying.

3. Word Detection

The first step in our caption generation pipeline detects a set of words that are likely to be part of the image's description. These words may belong to any part of speech, including nouns, verbs, and adjectives. We determine our vocabulary $\mathcal{V}$ using the 1000 most common words in the training captions. This list covers over 92% of the word occurrences in the training data. The complete set of words may be viewed in the appendix.

3.1. Training Word Detectors

Given a vocabulary of words, our next goal is to detect the words from images. We cannot use standard supervised learning techniques for learning detectors, since we do not know the image bounding boxes corresponding to the words. In fact, many words relate to concepts for which bounding boxes may not be easily defined, such as "open" or "beautiful". One possible approach is to use image classifiers that take as input the entire image. As we show in Section 6, this leads to worse performance since many words or concepts only apply to image sub-regions. Instead, we learn our detectors using the weakly-supervised approach of Multiple Instance Learning (MIL) [43, 48].

For each word $w \in \mathcal{V}$, MIL takes as input sets of "positive" and "negative" bags of bounding boxes. Each bag corresponds to one image. For word w and image i, a bag b_i is said to be positive if w is in image i's description, and negative otherwise. Intuitively, MIL performs training by iteratively finding positive instances within the positive bags, followed by retraining the detector using the updated positive labels. For computational efficiency, a set of 64 bounding boxes per image is generated using the object proposal generator Edge Boxes 70 from Zitnick and Dollár [49].

We use a noisy-OR version of MIL [48], where the probability of bag b_i containing word w is calculated from the probabilities of individual instances in the bag:

$$1 - \prod_{j \in b_i} \left(1 - p_{ij}^w\right) \tag{1}$$

where p_{ij}^w is the probability that bounding box j in image i corresponds to word w. For each bounding box, we compute features from the `fc6` layer of the convolutional neural network [21] trained on the ImageNet dataset [5]. These features have been shown to generalize beyond just the ImageNet classification task and perform well for a variety of vision tasks [8, 13, 34]. We compute p_{ij}^w from `fc6` features $\phi\left(b_{ij}\right)$ for box j in image i using a logistic function:

$$\frac{1}{1 + \exp\left(-v_w \phi(b_{ij}) - u_w\right)}, \tag{2}$$

where v_w are the weights associated with word w and u_w is the bias.

3.2. Generating Word Scores for a Test Image

Given a novel test image i, we compute n object bounding boxes $b_{ij}, j \in [1, \ldots, n]$. We compute the scores p_{ij}^w using Equation (2) for each bounding box b_{ij}. The score p_i^w of a word w appearing in the image's description corresponds to an aggregation of p_{ij}^w across all j, computed from the noisy-OR function. We do this for all words w in the vocabulary $\mathcal{V}$. Note that all the word detectors have been trained independently and hence their outputs need to be calibrated. To calibrate the output of different detectors, we use the image level likelihood p_i^w to compute precision on a held-out subset of the training data [14]. We threshold this precision value at a global threshold τ, and output all words $\tilde{\mathcal{V}}$ with a precision of τ or higher along with the image level probability p_i^w.

Figure 2 shows some sample MIL detections. For each image, we draw the box with the highest activation. Note that the method has not used any bounding box annotations for training, but is still able to reliably localize objects and also associate image regions with more abstract concepts.

4. Language Generation

We cast the generation process as a search for the likeliest sentence conditioned on the set of visually detected words. The language model is at the heart of this process because it defines the probability distribution over word sequences. Note that despite being a statistical model, the LM can encode very meaningful information, for instance that "running" is more likely to follow "horse" than "talk-

Table 1. Features used in the maximum entropy language model.

Feature	Type	Definition	Description
Attribute	0/1	$\bar{w}_l \in \tilde{\mathcal{V}}_{l-1}$	Predicted word is in the attribute set, i.e. has been visually detected and not yet used.
N-gram +	0/1	$\bar{w}_{l-N+1}, \cdots, \bar{w}_l = \kappa$ and $\bar{w}_l \in \tilde{\mathcal{V}}_{l-1}$	N-gram ending in predicted word is κ and the predicted word is in the attribute set.
N-gram -	0/1	$\bar{w}_{l-N+1}, \cdots, \bar{w}_l = \kappa$ and $\bar{w}_l \notin \tilde{\mathcal{V}}_{l-1}$	N-gram ending in predicted word is κ and the predicted word is not in the attribute set.
End	0/1	$\bar{w}_l = \kappa$ and $\tilde{\mathcal{V}}_{l-1} = \emptyset$	The predicted word is κ and all attributes have been mentioned.
Score	$\mathbb{R}$	score($\bar{w}_l$) when $\bar{w}_l \in \tilde{\mathcal{V}}_{l-1}$	The log-probability of the predicted word when it is in the attribute set.

ing." This information can help identify false word detections and encodes a form of commonsense knowledge about the world.

4.1. Statistical Model

To generate candidate captions for an image, we use a maximum entropy (ME) LM conditioned on the set of visually detected words. The ME LM estimates the probability of a word w_l conditioned on the preceding words $w_1, w_2, \cdots, w_{l-1}$, as well as the set of words with high likelihood detections $\tilde{\mathcal{V}}_l \subset \tilde{\mathcal{V}}$ that have yet to be mentioned in the sentence. The motivation of conditioning on the unused words is to encourage all the words to be used, while avoiding repetitions. The top 15 most frequent closed-class words[1] are removed from the set $\tilde{\mathcal{V}}$ since they are detected in nearly every image (and are trivially generated by the LM). It should be noted that the detected words are usually somewhat noisy. Thus, when the end of sentence token is being predicted, the set of remaining words may still contain some words with a high confidence of detection.

Following the definition of an ME LM [2], the word probability conditioned on preceding words and remaining objects can be written as:

$$\Pr(w_l = \bar{w}_l | \bar{w}_{l-1}, \cdots, \bar{w}_1, \text{<s>}, \tilde{\mathcal{V}}_{l-1}) = \frac{\exp\left[\sum_{k=1}^{K} \lambda_k f_k(\bar{w}_l, \bar{w}_{l-1}, \cdots, \bar{w}_1, \text{<s>}, \tilde{\mathcal{V}}_{l-1})\right]}{\sum_{v \in \mathcal{V} \cup \text{</s>}} \exp\left[\sum_{k=1}^{K} \lambda_k f_k(v, \bar{w}_{l-1}, \cdots, \bar{w}_1, \text{<s>}, \tilde{\mathcal{V}}_{l-1})\right]} \quad (3)$$

where <s> denotes the start-of-sentence token, $\bar{w}_j \in \mathcal{V} \cup \text{</s>}$, and $f_k(w_l, \cdots, w_1, \tilde{\mathcal{V}}_{l-1})$ and λ_k respectively denote the k-th max-entropy feature and its weight. The basic discrete ME features we use are summarized in Table 1. These features form our "baseline" system. It has proven effective to extend this with a "score" feature, which evaluates to the log-likelihood of a word according to the corresponding visual detector. We have also experimented with distant bigram features [24] and continuous space log-bilinear features [31, 32], but while these improved PPLX significantly, they did not improve BLEU, METEOR or human preference, and space restrictions preclude further discussion.

To train the ME LM, the objective function is the log-likelihood of the captions conditioned on the corresponding set of detected objects, i.e.:

$$L(\Lambda) = \sum_{s=1}^{S} \sum_{l=1}^{\#(s)} \log \Pr(\bar{w}_l^{(s)} | \bar{w}_{l-1}^{(s)}, \cdots, \bar{w}_1^{(s)}, \text{<s>}, \tilde{\mathcal{V}}_{l-1}^{(s)}) \quad (4)$$

where the superscript (s) denotes the index of sentences in the training data, and $\#(s)$ denotes the length of the sentence. The noise contrastive estimation (NCE) technique is used to accelerate the training by avoiding the calculation of the exact denominator in (3) [32]. In the generation process, we use the unnormalized NCE likelihood estimates, which are far more efficient than the exact likelihoods, and produce very similar outputs. However, all PPLX numbers we report are computed with exhaustive normalization. The ME features are implemented in a hash table as in [29]. In our experiments, we use N-gram features up to 4-gram and 15 contrastive samples in NCE training.

4.2. Generation Process

During the generation stage, we perform a left-to-right beam search similar to the one used in [38]. This maintains a stack of length l partial hypotheses. At each step in the search, every length l path on the stack is extended with a set of likely words, and the resulting length $l+1$ paths are stored. The top k length $l+1$ paths are retained and the others pruned away.

We define the possible extensions to be the "end of sentence" token </s> , the 100 most frequent words, the set of attribute words that remain to be mentioned, and all the words in the training data that have been observed to follow the last word in the hypothesis. Pruning is based on the likelihood of the partial path. When </s> is generated, the full path to </s> is removed from the stack and set aside as a completed sentence. The process continues until a maximum sentence length L is reached.

After obtaining the set of completed sentences $\mathcal{C}$, we form an M-best list as follows. Given a target number of T image attributes to be mentioned, the sequences in $\mathcal{C}$ covering at least T objects are added to the M-best list, sorted in descending order by the log-likelihood. If there are less than M sequences covering at least T objects found in $\mathcal{C}$, we reduce T by 1 until M sequences are found.

[1] The top 15 frequent closed-class words are "a", "on", "of", "the", "in", "with", "and", "is", "to", "an", "at", "are", "next", "that", and "it".

Table 2. Features used by MERT.

1. The log-likelihood of the sequence.
2. The length of the sequence.
3. The log-probability per word of the sequence.
4. The logarithm of the sequence's rank in the log-likelihood.
5. 11 binary features indicating whether the number of mentioned objects is x ($x = 0, \dots, 10$).
6. The DMSM score between the sequence and the image.

5. Sentence Re-Ranking

Our LM produces an M-best set of sentences. Our final stage uses MERT [33] to re-rank the sentences. MERT uses a linear combination of features computed over an entire sentence that cannot be used in the original generation process, such as sentence length. The MERT model is trained on the M-best lists for the validation set using the BLEU metric, and applied to the M-best lists for the test set. Finally, the best sequence after the re-ranking is selected as the caption of the image.

Along with the use of several standard features, we introduce a new multimodal similarity model, discussed below. The full list of features used by MERT are given in Table 2.

5.1. Deep Multimodal Similarity Model

To model global similarity between images and text, we develop a Deep Multimodal Similarity Model (DMSM). The DMSM learns two neural networks that map images and text fragments to a common vector representation. We measure similarity between images and text by measuring dot-product similarity between their corresponding vectors. This dot product is used by MERT to re-rank the sentences. The DMSM is closely related to the unimodal Deep Structured Semantic Model (DSSM) [16, 41], but extends it to the multimodal setting. The DSSM was initially proposed to model the semantic relevance between textual search queries and documents, and is extended in this work to replace the query vector in the original DSSM by the image vector computed from the deep convolutional network.

The DMSM consists of a pair of neural networks, one for mapping each input modality to a common semantic space, which are trained jointly. In training, the data consists of a set of image/caption pairs. The loss function minimized during training represents the negative log posterior probability of the caption given the corresponding image.

Image model: We map images to semantic vectors via a deep convolutional neural network consisting of several convolutional, max-pooling, response-normalization and fully connected layers. This architecture has been very successful for large-scale image classification [21] and the learned features have shown to transfer to a broad variety of vision tasks [40]. Motivated by these results, and to simplify training, we do not relearn all the weights for our image model. Instead, we initialize several layers using a network pre-trained [17] on the ILSVRC 2012 image classification dataset and only train a few fully connected weight layers on top of them. Thus, although this model is quite deep (12 layers), we do not train the weights in the first seven layers. We cross-validated the number of additional layers based on performance on a validation set. The representation at the last fully connected layer of the model is the semantic vector for the given image, and must be of the same size as the last layer of the text model, so that the cosine similarity score between them can be computed.

Text model: The text part of the DMSM maps text fragments to semantic vectors, in the same manner as in the original DSSM. In general, the text fragments can be a full caption which when treated as a bag of words can be represented using a fixed size word-count vector. Following [16] we convert the word-count vector to a letter-trigram count vector, which uses the count distribution of context-dependent letters to represent a word. This representation has the advantage of reducing the size of the input layer while generalizing well to infrequent, unseen and incorrectly spelled words. This representation is then forward propagated through a deep fully connected network to produce the semantic vector at the last layer.

Objective and training: We define the relevance R as the cosine similarity between an image or query (Q) and a text fragment or document (D) based on their representations y_Q and y_D obtained using the image and text models:

$$R(Q, D) = \text{cosine}(y_Q, y_D) = \frac{{y_Q}^T y_D}{\|y_Q\|\|y_D\|} \tag{5}$$

For a given image-text pair, we can compute the posterior probability of the text being relevant to the image via:

$$P(D|Q) = \frac{\exp(\gamma R(Q, D))}{\Sigma_{D' \in \mathbb{D}} \exp(\gamma R(Q, D'))}. \tag{6}$$

Here γ is a smoothing factor determined using the validation set. $\mathbb{D}$ denotes the set of all candidate documents (captions) which should be compared to the query (image). We found that restricting $\mathbb{D}$ to one matching document D_+ and a fixed number N of randomly selected non-matching documents D^- worked reasonably well, although using noise-contrastive estimation could further improve results. Thus, for each image we select one relevant text fragment and N non-relevant fragments to compute the posterior probability. During training, we adjust the model parameters Λ to minimize the negative log posterior probability that the relevant captions are matched to the images:

$$L(\Lambda) = -\log \prod_{(Q, D^+)} P(D^+|Q) \tag{7}$$

Additional details are given in the appendix.

Figure 3. Qualitative results for several randomly chosen images on the Microsoft COCO dataset. Word detections are shown by color boxes with corresponding words and scores. Our generated caption (black) and a human caption (blue) are shown for each image.

6. Experimental Results

We next describe the datasets used for testing, followed by an evaluation of our approach for word detection and experimental results on sentence generation.

6.1. Datasets

Most of our results are reported on the Microsoft COCO dataset [26]. The dataset contains 82,783 training images and 40,504 validation images. The images create a challenging testbed for image captioning since most images contain multiple objects and significant contextual information. The COCO dataset provides 5 human-annotated captions per image, for a total of over 400k captions. The test annotations are not available, so we split the validation set into validation and test sets.

For experimental comparison with prior papers, we also report results on the PASCAL sentence dataset [37], which contains 1000 images from the 2008 VOC Challenge [10], with 5 human captions each.

6.2. Word Detection

To gain insight into our weakly-supervised approach for word detection using MIL, we measure its accuracy on the word classification task. Note that this is an extremely challenging task, since conceptually similar words are classified separately; for example, the words cat/cats/kitten, or run/ran/running all correspond to different classes. Note that attempts at adding further supervision, e.g., in the form of lemmas, did not result in significant gains. If a word is used in at least one ground truth caption it is in the positive class for the image.

Average Precision (AP) and Precision at Human Recall (PHR) results for different parts of speech are shown in Table 3. We observe that humans often disagree on which words to use, and human agreement (as measured by benchmarking words from one caption against the rest) varies for different words (see appendix). Human precision increases as we consider more captions but human recall remains fairly constant. Therefore, we measure precision of the proposed algorithm at human recall (denoted PHR) and com-

Table 3. Average precision (AP) and Precision at Human Recall (PHR) for words with different parts of speech. Results are shown using a chance classifier, full image classification, and Noisy OR multiple instance learning.

	Metric	Noun	Verb	Adjective	Determiner	Pronoun	Preposition	Others	all
Count		616	176	119	10	11	38	30	1000
Chance	AP	1.95	2.29	2.49	23.63	4.66	11.87	7.65	2.87
Classification	AP	30.42	14.66	18.65	29.97	14.25	19.16	13.62	25.13
MIL NOR	AP	31.44	14.87	19.68	31.05	14.78	20.24	14.07	25.99
Classification	PHR	36.28	23.53	31.94	34.85	22.50	26.64	20.42	32.51
MIL NOR	PHR	38.32	23.69	34.79	37.85	21.60	28.91	20.32	34.24
Human Agreement	PHR	63.83	35.07	35.86	43.12	32.51	34.32	31.60	52.80

pare it to human precision to measure similarity between human and machine word classification performance.

We report two baselines. The first (Chance) is the result of randomly classifying each word. The second (Classification) is the result of training a whole image classifier with a single window using the same features (`fc6` layer of the CNN [21]) as used by our MIL approach.

As shown in Table 3, the MIL NOR approach improves over both baselines for all parts of speech, demonstrating that better localization can help predict words. In fact, we observe the largest improvement for nouns and adjectives, which often correspond to concrete objects in an image subregion. Results for both classification and MIL NOR are lower for parts of speech that may be less visually informative and difficult to detect, such as adjectives (e.g., *few*, which has an AP of 1.94), pronouns (e.g., *himself*, with an AP of 1.71), and prepositions (e.g., *before*, with an AP of 0.68). In comparison words with high AP scores are typically either visually informative (*red*: AP 62.4, *her*: AP 34.7) or associated with specific objects (*polar*: AP 78.6, *stuffed*: AP 60.3). Qualitative results demonstrating word localization are shown in Figures 2 and 3.

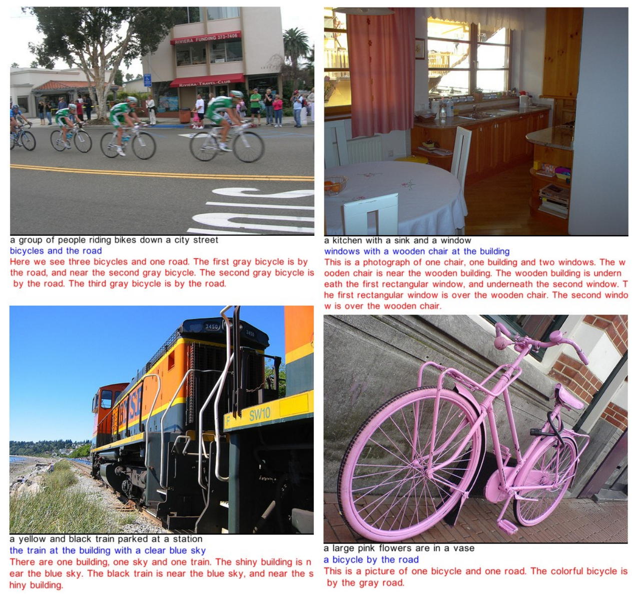

Figure 4. Qualitative results for images on the PASCAL sentence dataset. Captions using our approach (black), Midge [30] (blue) and Baby Talk [22] (red) are shown.

6.3. Caption Generation

We next describe our caption generation results, beginning with a short discussion of evaluation metrics.

Metrics: The sentence generation process is measured using both automatic metrics and human studies. We use three different automatic metrics: PPLX, BLEU [36], and METEOR [1]. PPLX (perplexity) measures the uncertainty of the language model, corresponding to how many bits on average would be needed to encode each word given the language model. Hence, a lower PPLX indicates a better score. BLEU [36] is widely used in machine translation and measures the fraction of N-grams (up to 4-gram) that are in common between a hypothesis and a reference or set of references. METEOR [1] measures unigram precision and recall, extending exact word matches to include similar words based on WordNet synonyms and stemmed tokens.

All of these automatic metrics are known to only roughly correlate with human judgment [9]. We therefore include crowdsourcing experiments to further explore the quality of our models. Each task presents a crowd-worker with an image and two captions. One is automatically generated, and the other is a human caption. The worker is asked to select which caption better describes the image, or to choose a "same" option when they are of equal quality. In each experiment, 250 Turkers were asked to compare 20 caption pairs each, and 5 Turkers judged each caption pair. We use Crowdflower, which automatically filters out spammers. The ordering of the captions was randomized to avoid bias, and we included four check-cases where the answer was known and obvious; annotators who missed any of these were excluded. The final judgment is the majority vote of the judgment of the 5 Turkers. In case of a tie, one-half of a count is distributed to the two best answers.

Table 4. Overall system performance on the Microsoft COCO dataset. We show three conditions: the full system (Baseline+Score+DMSM), leaving out the DMSM, and leaving out both the DMSM and the floating-point score for each word. We report results using Perplexity (PPLX), BLEU and METEOR. Results from Amazon Turk studies of subjective performance are also shown, with error bars in parentheses.

System	PPLX	BLEU	METEOR	≈human	>human	≥human
No vision conditioning	24.1	1.18%	6.76%			
Randomly-selected human-written captions	–	1.68%	7.25%			
Baseline	20.9	16.94%	18.85%	9.9% (±1.9%)	2.4% (±1.0%)	12.3% (±2.1%)
Baseline+Score	20.2	20.12%	20.52%	16.9% (±2.4%)	3.9% (±1.2%)	20.8% (±2.6%)
Baseline+Score+DMSM	20.2	21.05%	20.71%	18.7% (±2.4%)	4.6% (±1.2%)	**23.3%** (±2.6%)
Human-written captions	–	19.32%	24.07%			

Generation results: Table 4 summarizes our caption generation results on the Microsoft COCO dataset. For experimental comparison, we provide several baselines. These include two baselines that measure the complexity of the dataset: Unconditioned Generation, which generates sentences by sampling an N-gram LM without knowledge of the visual word detectors; and Shuffled Human, which randomly picks another human generated caption from another image. Both the BLEU and METEOR scores are very low for these approaches, demonstrating the large variation and complexity of the Microsoft COCO dataset.

We provide results on three variants of our algorithm, Baseline, Baseline+Score, and Baseline+Score+DMSM. Baseline uses the ME LM with all discrete features described in Table 1. Baseline+Score adds the feature for the word detector score into the ME LM. Both of them use the same set of sentence features (excluding the DMSM score) described in Section 5 when re-ranking the captions using MERT. Baseline+Score+DMSM uses the same ME LM as Baseline+Score, but adds the DMSM score as a feature for re-ranking. As shown in Table 4, the PPLX of the ME LM with and without the word detector score feature is roughly the same. BLEU and METEOR are improved with addition of the word detector scores in the ME LM; and further improved with addition of the DMSM scores in re-ranking. Surprisingly, the BLEU scores are actually above those produced by human generated captions (21.05% vs. 19.32%).

To gain a better understanding of the perceived quality of our captions, we report the percentage of cases where humans judged them to be the "same" quality as a human caption, "better", and "same or better", deriving error bars from binomial distribution standard errors. We see that the Baseline+Score+DMSM approach produces captions that are judged to be of the same or better quality than human-written descriptions 23.3% of the time, which is a significant improvement over the Baseline result. Note that the given error bars assume that the samples are independent: if we use the McNemar paired test, which compares the results of each of the three systems on the same set of images, then we find that adding DMSM to our system makes the majority vote significantly better, with a p value of 0.024.

Table 5. Results on the PASCAL sentence dataset comparing Midge [30] and our approach using BLEU and METEOR.

System	BLEU	METEOR
Midge [30]	1.74%	8.79%
Baseline+Score	14.5%	17.25%

Qualitative results are shown in Figure 3.

To enable direct comparison with previous work on automatically generating image captions, we also test on the PASCAL sentence dataset [37], which was used in both the Midge [30] and Baby Talk [22] systems. We show significantly improved results over the Midge [30] system, as measured by both BLEU and METEOR (Table 5). Baby Talk generates long captions with multiple sentences, making comparison by BLEU and METEOR difficult. However, to give a basic qualitative sense of the progress quickly being made in this field, Figure 4 shows output from our system, the Midge system, and the Baby Talk system.[2]

7. Conclusion

This paper presents a new system for generating novel captions from images. Our system trains on images and corresponding human-written captions. The system learns to extract nouns, verbs, and adjectives from regions in the image. These detected words then guide a language model to generate text that reads well and includes the detected words. Finally, we use a global similarity model to re-rank candidate captions based on overall similarity of the image to the caption in a common vector representation.

Our system exceeds human captioning performance, as measured by the (admittedly limited) BLEU translation metric. Our system's captions have been judged by Turkers to be equal to or better than human-written captions 23.3% of the time.

[2] Images were selected visually, without viewing system captions.

8. Appendix

The appendix covers the following topics: Section 8.1 provides a discussion of human agreement for word prediction. Additional analysis of our automatically detected words is presented in Section 8.2 and further details for the Deep Multimodal Similarity Model (DMSM) and BLEU metric are given in Sections 8.3 and 8.4.

8.1. Human Agreement for Word Detection

When examining human agreement on captions, it becomes clear that there are many equivalent ways to say essentially the same thing. We can compute the human precision and recall for a given word w by benchmarking words used in the $k+1$ human caption with respect to words used in the first k reference captions. Note that we use weighted versions of precision and recall, where each negative image has a weight of 1 and each positive image has a weight equal to the number of captions containing the word w. Human precision (H_p) and human recall (H_r) can be computed from the counts of how many subjects out of k use the word w to describe a given image over the whole dataset.

We plot H_p versus H_r for a set of nouns, verbs and adjectives, and all 1000 words considered in Figure 5. Nouns referring to animals like 'elephant' have a high recall, which means that if an 'elephant' exists in the image, a subject is likely to talk about it (which makes intuitive sense, given 'elephant' images are somewhat rare, and there are no alternative words that could be used instead of 'elephant'). On the other hand, an adjective like 'bright' is used inconsistently and hence has low recall. Interestingly, words with high recall also have high precision. Indeed, all the points of human agreement appear to lie on a one-dimensional curve in the two-dimension precision-recall space.

This observation motivates us to propose a simple model for when subjects use a particular word w for describing an image. Let o denote an object or visual concept associated with word w, n be the total number of images, and k be the number of reference captions. Next, let $q = P(o=1)$ be the probability that object o exists in an image. For clarity these definitions are summarized in Table 6. We make two simplifications. First, we ignore *image level saliency* and instead focus on *word level saliency*. Specifically, we only model $p = P(w=1|o=1)$, the probability a subject uses w given that o is in the image, without conditioning on the image itself. Second, we assume that $P(w=1|o=0)=0$, i.e. that a subject does not use w unless o is in the image. As we will show, even with these simplifications our model suffices to explain the empirical observations in Figure 5 to a reasonable degree of accuracy.

Given these assumptions, we can model human precision $\widetilde{H_p}$ and recall $\widetilde{H_r}$ for a word w given only p and k. First, given k captions per image, we need to compute the

Table 6. Model defintions.

o	=	object or visual concept
w	=	word associated with o
n	=	total number of images
k	=	number of captions per image
q	=	$P(o=1)$
p	=	$P(w=1\|o=1)$

expected number of (1) captions containing w (cw), (2) true positives (tp), and (3) false positives (fp). Note that in our definition there can be up to k true positives per image (if $cw=k$, i.e. each of the k captions contains word w) but at most 1 false positive (if none of the k captions contains w). The expectations, in terms of k, p, and q are:

$$\begin{aligned}
E[cw] &= \Sigma_{i=1}^{k} P(w^i=1) \\
&= \Sigma_i P(w^i=1|o=1)P(o=1) \\
&\quad +\Sigma_i P(w^i=1|o=0)P(o=0) \\
&= kpq + 0 = \boxed{kpq} \\
E[tp] &= \Sigma_{i=1}^{k} P(w^i=1 \wedge w^{k+1}=1) \\
&= \Sigma_i P(w^i=1 \wedge w^{k+1}=1|o=1)P(o=1) \\
&\quad +\Sigma_i P(w^i=1 \wedge w^{k+1}=1|o=0)P(o=0) \\
&= kppq + 0 = \boxed{kp^2q} \\
E[fp] &= P(w^1 \dots w^k = 0 \wedge w^{k+1}=1) \\
&= P(o=1 \wedge w^1 \dots w^k = 0 \wedge w^{k+1}=1) \\
&\quad +P(o=0 \wedge w^1 \dots w^k = 0 \wedge w^{k+1}=1) \\
&= q(1-p)^k p + 0 = \boxed{q(1-p)^k p}
\end{aligned}$$

In the above $w^i = 1$ denotes that w appeared in the i^{th} caption. Note that we are also assuming independence between subjects conditioned on o. We can now define model precision and recall as:

$$\begin{aligned}
\widetilde{H_p} &:= \frac{nE[tp]}{nE[tp]+nE[fp]} = \frac{pk}{pk+(1-p)^k} \\
\widetilde{H_r} &:= \frac{nE[tp]}{nE[cw]} = p
\end{aligned}$$

Note that these expressions are independent of q and only depend on p. Interestingly, because of the use of weighted precision and recall, the recall for a category comes out to be exactly equal to p, the probability a subject uses w given that o is in the image.

We set $k=4$ and vary p to plot $\widetilde{H_p}$ versus $\widetilde{H_r}$, getting the curve as shown in blue in Figure 5 (bottom left). The curve explains the observed data quite well, closely matching the precision-recall tradeoffs of the empirical data (although not perfectly). We can also reduce the number of captions from four, and look at how the empirical and predicted precision and recall change. Figure 5 (bottom right), shows this variation as we reduce the number of reference captions per image from four to one annotations. We see that the points

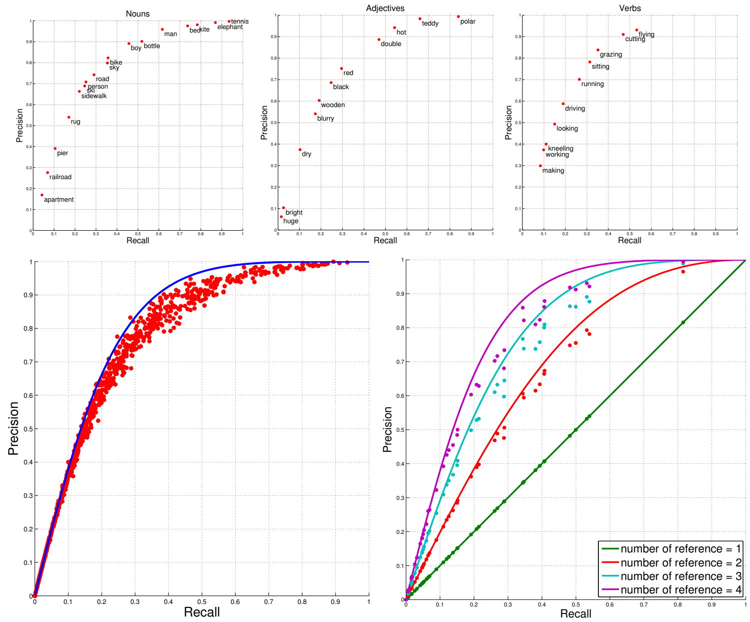

Figure 5. Precision-recall points for human agreement: we compute precision and recall by treating one human caption as prediction and benchmark it against the others to obtain points on the precision recall curve. We plot these points for example nouns (top left), adjectives (top center), and verbs (top right), and for all words (bottom left). We also plot the fit of our model for human agreement with the empirical data (bottom left) and show how the human agreement changes with different number of captions being used (bottom right). We see that the human agreement point remains at the same recall value but dips in precision when using fewer captions.

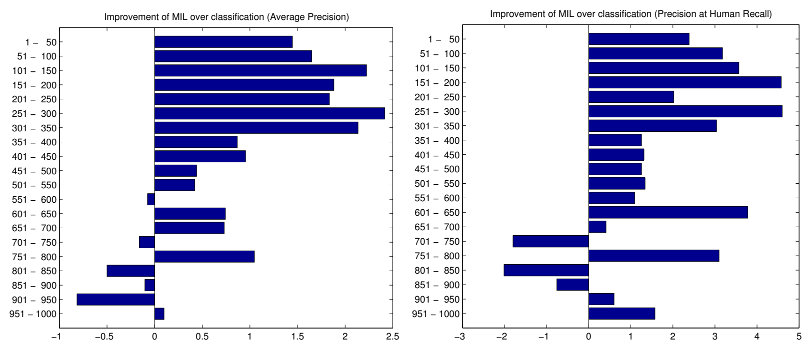

Figure 6. MIL versus classification: we show performance differences in word prediction between MIL and classification as measured by average precision and precision at human recall. First, we sort the 1000 words by their frequency and divide them into 20 groups of 50. Next, for each group of 50 words, we plot the average difference in accuracy between MIL and classification. For frequent words MIL shows a clear advantage using both metrics. For less frequent words classification and MIL perform similarly.

of human agreement remain at the same recall value, but decrease in their precision, which is consistent with what the model predicts. Also, the human precision at infinite subjects will approach one, which is again reasonable given that a subject will only use the word w if the corresponding object is in the image (and in the presence of infinite subjects someone else will also use the word w).

In fact, the fixed recall value can help us recover p, the probability that a subject will use the word w in describing the image given the object is present. Nouns like 'elephant' and 'tennis' have large p, which is reasonable. Verbs and adjectives, on the other hand, have smaller p values, which can be justified from the fact that a) subjects are less likely to describe attributes of objects and b) subjects might use a different word (synonym) to describe the same attribute.

This analysis of human agreement also motivates using a different metric for measuring performance. We propose Precision at Human Recall (PHR) as a metric for measuring performance of a vision system performing this task. Given that human recall for a particular word is fixed and precision varies with the number of annotations, we can look at system precision at human recall and compare it with human precision to report the performance of the vision system.

8.2. Word Detection Analysis

In Figure 6, we show the improvement provided by the use of MIL over that of classification with respect to word frequency. We observe that for frequently occurring words, MIL provides a larger boost in performance. For less frequent words, MIL and classification provide similar results.

A list of words used by our system is shown in Table 7.

8.3. DMSM Additional Details

In this work, we use our Deep Multimodal Similarity Model (DMSM) to rank a set a candidate captions for an image in order of relevance. Here we describe some details of the training and validation procedures for this model.

The DMSM is trained to maximize the relevance score between the images and the corresponding captions in the training set, and then used to re-rank the set of captions for test images generated by the language model. The trainable weights in the query and document models (denoted by Λ in the main text) are learned using mini-batch gradient descent. Our initial experiments suggested that the DMSM objective function could be reliably optimized using a constant learning rate. We fixed N and γ to 40 and 10 respectively in our experiments, based on an initial pilot study.

A validation set score was used to select the best model during training. We measured the Harmonic mean Best Rank (HBR) for the model after each training epoch on the validation set. This was calculated as follows: for each image we first compute the *Best Rank* which is the highest rank obtained by any of its true captions when all validation set captions are ranked in order of their relevance. We then computed the harmonic mean of the Best Ranks for all images. The harmonic mean was found to be a more stable score than the Arithmetic mean of Best Ranks (ABR) which is sensitive to a few low ranks.

To find the best performing model, we first identified effective values of the hyperparameters using initial experiments on a subset of data. We then trained several models with different architectures and for each training run, selected the model with the best validation score. Using these experiments, we were able to observe the effect of network architectures on model performance for both query and document networks. Since representations from both the sixth weight layer (often known as "FC6") and the seventh weight layer ("FC7") of the convolutional network trained on ILSVRC 2012 have been shown to perform well under different settings in the past, we trained models with either six or seven pre-trained weight layers which remained fixed. Based on the above validation study, the final model which produced the best ranking performance (HBR) was used to re-rank the captions generated by the language model.

Table 8 summarizes the results of our validation experiments. The best performing architecture contains ten weight layers in the query model and five layers in the document model. The first seven layers of the query model were initialized using the Caffe reference model and remained fixed during training, so another way to describe our query model is to have three fully connected layers on top of FC7 features obtained from a network trained on the ILSVRC 2012 image classification dataset.

It is notable that in general we obtained better performance when using FC7 features than when using FC6 features. Since these features are from the layer closest to the softmax classification layer of the ImageNet model, we can say that having highly discriminative features is beneficial in our particular experimental setting.

The most striking observation from the results is the clear advantage of network depth for both query and document models. This is not surprising for the query model – network depth has proven to be crucial for many computer vision tasks. However, we see a stark contrast between the best document model obtained here and the models used for tasks such as web search in the past [16], where search queries and web document titles are matched instead of images and text fragments. Both search queries and web page titles are typically short in length and their semantic representations are less entangled, compared to descriptions of natural images. As a result, we found that increasing the document model's depth up to 5 or 6 layers helped performance. Although our document model currently does not take into account the relative ordering of words in the captions, it is able to capture the relationships between them quite well without the use of explicit dependency parsing. This is consistent with recent studies [41, 18, 6] in the area of natural language processing with neural networks.

8.4. BLEU Metric

BLEU was first proposed by Papineni, et al. [36] to measure the quality of machine translation output. The BLEU-4 score is commonly used in the machine translation community and also in this paper. It is computed by:

$$\text{BLEU-4} = BP \cdot \exp\left(\frac{1}{4}\sum_{n=1}^{4}\log p_n\right). \tag{8}$$

The precision of $ngrams$ in the output is computed by $p_n = \frac{\#(ngram\ matched)}{\#(ngram)}$ and the Brevity Penalty (BP) is computed by $BP = min(1, e^{(1-\frac{r}{c})})$, where r is the length of the reference and c is the length of the candidate output. Note that p_n, r, and c are computed over the whole testing corpus. When multiple references per sentence are used, the length of the reference that is closest (longer or shorter) to the length of the candidate is used to compute r, and when computing p_n, the maximum $\#(ngram\ matched)$ at each sentence is limited to the maximum number of $ngram$ in one of the references of that sentence.

Table 7. List of 1000 words used in our caption generating systems.

Noun [NN]: man, people, woman, table, street, person, group, field, tennis, front, plate, room, train, dog, cat, water, baseball, bathroom, sign, food, kitchen, grass, bus, pizza, side, building, snow, bed, ball, beach, couple, boy, men, toilet, city, road, skateboard, player, clock, game, girl, bear, picture, bench, area, laptop, cake, horse, phone, sink, board, giraffe, computer, frisbee, living, air, truck, window, desk, car, trees, umbrella, motorcycle, tree, wall, close, park, elephant, fire, stop, sky, court, child, kite, bat, skis, surfboard, background, bowl, sheep, photo, back, airplane, boat, couch, chair, bunch, view, ocean, light, glass, cell, traffic, bird, zebra, hydrant, plane, mirror, counter, fence, women, sandwich, shirt, hand, horses, sidewalk, wave, giraffes, lot, floor, flowers, tracks, vase, cars, parking, baby, racket, ground, vegetables, elephants, bananas, tie, tower, day, zebras, dirt, middle, image, hill, bike, slope, station, signs, head, skiing, wine, piece, cows, luggage, snowy, broccoli, wii, hat, refrigerator, ski, glasses, display, suit, mountain, fruit, herd, kites, cow, children, camera, buildings, corner, pole, pair, trick, keyboard, airport, chairs, umbrellas, television, track, stove, box, boats, door, video, animals, crowd, soccer, tv, lady, plates, surf, banana, birds, body, wood, coffee, dogs, lots, guy, runway, motorcycles, cheese, someone, paper, players, house, skateboarder, bedroom, river, cup, something, night, lights, restaurant, walk, meat, bears, snowboard, brick, jet, home, metal, bicycle, shower, skier, ramp, items, decker, face, racquet, passenger, hands, surfer, line, animal, book, intersection, slice, mouth, tray, cut, suitcase, bottle, scissors, batter, store, screen, bag, number, zoo, enclosure, knife, half, jacket, carrots, donuts, bridge, microwave, row, sand, way, tub, kids, silver, lake, meal, pile, buses, toy, adult, forest, cabinets, skiers, oranges, boys, furniture, mouse, swing, bread, girls, seat, cloudy, kid, photograph, chocolate, waves, hair, drinking, dining, monitor, drink, fork, scene, fruits, salad, apples, cats, rocks, shelf, apple, office, meter, birthday, walls, stone, market, pan, fries, rain, flower, blanket, windows, teeth, snowboarder, tables, dish, books, pictures, sun, uniform, mountains, police, bikes, donut, helmet, edge, surfboards, rail, platform, rock, dress, bath, slices, yard, base, statue, cellphone, shot, time, controller, pitch, catcher, path, branch, vases, sauce, computers, pieces, motor, vehicle, doughnut, shore, family, case, country, surface, cart, boards, town, dinner, basket, cooking, tarmac, plant, types, lamp, lap, hotel, pizzas, doughnuts, laptops, others, toppings, pitcher, trains, guys, distance, rice, variety, engine, jump, trucks, benches, appliances, bathtub, tricks, passengers, phones, beer, pen, woods, end, post, shop, candles, gear, plants, place, pasture, curb, cattle, poles, railroad, graffiti, drinks, carriage, brush, chicken, toothbrush, center, match, bags, bottles, fireplace, shoes, school, bar, sandwiches, sofa, planes, tile, steel, neck, dock, pot, boxes, feet, fridge, beds, pillows, trunk, control, tomatoes, skateboards, equipment, clothes, bushes, arm, rack, container, church, sinks, suitcases, space, bicycles, foods, towel, vehicles, mother, work, airplanes, potatoes, legs, style, dishes, run, dessert, cabinet, cream, hay, subway, shorts, breakfast, cement, surfers, wire, christmas, spoon, trail, show, painting, highway, outdoors, backpack, couches, round, swings, games, reflection, pool, soup, business, umpire, sunglasses, blender, team, assortment, controllers, lawn, hillside, stairs, swimming, adults, garden, van, clocks, bun, rug, shelves, wedding, hotdog, ledge, onions, stall, flag, skies, gate, pillow, rackets, toddler, trash, arms, desktop, cups, fish, event, foot, restroom, things, eyes, clouds, floors, landing, land, closeup, ice, eggs, glove, flock, kitten, party, desert, doors, steps, lunch, coat, race, ceiling, cakes, construction, trailer, friends, monitors, turn, sunset, curtain, boarder, type, machine, kind, pastries, rider, towels, cage, bottom, pots, roof, pond, sale, chips, bite, ear, picnic, pie, pasta, sea, toilets, cloth, transit, veggies, walkway, foreground, sides, photos, device, pants, doorway, tour, toys, fighter, pepperoni, colors, signal, smoke, action, gold, suits, object, clothing, shade, rest, leash, scooter, carrot, grill, kinds, houses, leather, hole, winter, ties, pastry, structure, pedestrians, peppers, steam, commuter, ship, vanity, palm, bow, tomato, pier, papers, railing, containers, officer, island, lettuce, streets, outfit, cap, farm, t, apartment, beans, produce, stack, rainy

Verb [VB]: is, sitting, standing, are, holding, riding, has, walking, playing, parked, looking, flying, wearing, laying, sits, eating, s, covered, filled, stands, stand, being, sit, driving, taking, grazing, doing, hanging, holds, cutting, going, talking, jumping, swinging, watching, looks, posing, smiling, topped, traveling, getting, hit, using, set, be, carrying, waiting, running, lying, dressed, preparing, surfing, rides, pulling, shown, made, surrounded, look, play, sleeping, showing, decorated, walks, lined, colored, throwing, displayed, have, attached, coming, crossing, leaning, making, seen, hitting, watch, resting, painted, catch, leaves, been, setting, working, trying, moving, stopped, ride, having, skateboarding, eat, placed, shows, passing, eaten, brushing, perched, reading, enjoying, watches, bowls, performing, gathered, mounted, feeding, flies, left, containing, fenced, fly, take, including, pulled, striped, takes, reaching, served, catching, floating, plays, putting, shaped, pose, poses, taken, staring, sticking, prepares, serving, smiles, lays, serve, stacked, cooked, throw, drives, seated, boarding, docked, get, go, eats, jumps, drawn, facing, snowboarding, skating, see, says, closed, pointing, arranged, kneeling, graze, tied, does, overlooking, reads, hold, fashioned, appears, loaded, gets, wrapped, was, giving, petting, held, make, features, contains, lies, leading, prepared, racing, used, opened

Adjective [JJ]: next, white, large, top, small, black, red, young, blue, green, other, several, wooden, brown, yellow, open, many, old, little, big, teddy, orange, different, tall, grassy, full, ready, stuffed, skate, long, pink, hot, double, empty, oven, colorful, various, high, remote, busy, male, gray, dark, few, older, purple, past, grey, outdoor, lush, female, lit, plastic, cross, sunny, bright, clean, public, beautiful, clear, nice, sandy, wet, nintendo, single, modern, tiled, multiple, pretty, cute, fresh, dirty, right, vintage, concrete, crowded, sliced, huge, flat, dry, giant, commercial, polar, professional, lone, wild, asian, square, same, cluttered, french, military, low, mid, wooded, rocky, stainless, assorted, broken, electronic, tan, new, vegetable, wide, messy, overhead, urban, narrow, net, antique, fancy, plain, short, baked, blurry, decorative, electric, hard, residential

Determiner [DT]: a, the, an, some, each, this, another, all, no, both

Pronoun [PRP]: it, his, her, their, its, them, he, him, they, she, himself

Preposition [IN]: on, of, in, with, at, near, by, while, for, over, from, through, around, outside, as, behind, under, into, inside, along, above, across, beside, during, against, about, like, between, onto, towards, underneath, below, after, atop, toward, among, before, beneath

Others: and, to, two, that, down, up, there, three, out, one, very, together, off, four, who, can, or, just, five, well, alone, nearby, not, away, where, which, still, what, partially, six

Table 8. DMSM caption retrieval results on the COCO dataset. The models are listed top to bottom in increasing order of performance.

Untrained Query Layers	Trained Query Layers	Trained Doc. Layers	Trained Query Layer Sizes	Trained Doc. Layer Sizes	ABR	HBR
6	2	2	1000-300	1000-300	1331.07	33.77
6	3	3	1000-500-300	1000-500-300	1399.41	33.76
6	5	6	2000-1000-800-500-300	4000-2000-1000-500-300	1357.96	28.6
7	2	2	1000-300	1000-300	1090.45	28.43
6	3	7	1000-500-300	4000-2000-1000-800-500-300-300	1171.54	27.45
6	4	6	2000-1000-500-300	4000-2000-1000-500-300	1259.57	27.43
6	3	5	1000-500-300	4000-1000-500-300-300	1190.38	27.35
6	3	3	1000-500-300	4000-2000-300	1216.74	26.75
6	3	6	1000-500-300	4000-2000-1000-500-300-300	1159.47	26.45
7	3	6	1000-500-300	4000-2000-1000-500-300-300	1069.83	24.4
7	3	7	1000-500-300	4000-2000-1000-800-500-300-300	1026.04	23.63
7	3	5	2000-1000-300	4000-2000-1000-500-300	938.71	21.68

References

[1] S. Banerjee and A. Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 2005. 2, 7

[2] A. L. Berger, S. A. D. Pietra, and V. J. D. Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 1996. 4

[3] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. Hruschka Jr, and T. M. Mitchell. Toward an architecture for never-ending language learning. In *AAAI*, 2010. 2

[4] X. Chen, A. Shrivastava, and A. Gupta. Neil: Extracting visual knowledge from web data. In *ICCV*, 2013. 1

[5] 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. 2, 3

[6] M. Denil, A. Demiraj, N. Kalchbrenner, P. Blunsom, and N. de Freitas. Modelling, visualising and summarising documents with a single convolutional neural network. *arXiv preprint arXiv:1406.3830*, 2014. 11

[7] S. Divvala, A. Farhadi, and C. Guestrin. Learning everything about anything: Webly-supervised visual concept learning. In *CVPR*, 2014. 1

[8] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014. 3

[9] D. Elliott and F. Keller. Comparing automatic evaluation measures for image description. In *ACL*, 2014. 7

[10] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, June 2010. 2, 6

[11] 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. 2

[12] 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. 2

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2, 3

[14] G. Gkioxari, B. Hariharan, R. Girshick, and J. Malik. Using k-poselets for detecting people and localizing their keypoints. In *CVPR*, 2014. 3

[15] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *JAIR*, 47:853–899, 2013. 2

[16] P. Huang, X. He, J. Gao, L. Deng, A. Acero, and L. Heck. Learning deep structured semantic models for web search using clickthrough data. In *CIKM*, 2013. 5, 11

[17] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, June 2014. 5

[18] N. Kalchbrenner, E. Grefenstette, and P. Blunsom. A convolutional neural network for modelling sentences. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, June 2014. 11

[19] A. Karpathy, A. Joulin, and L. Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. *arXiv preprint arXiv:1406.5679*, 2014. 2

[20] R. Kiros, R. Zemel, and R. Salakhutdinov. Multimodal neural language models. In *NIPS Deep Learning Workshop*, 2013. 2

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, 2012. 2, 3, 5, 7

[22] 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. 1, 2, 7, 8

[23] P. Kuznetsova, V. Ordonez, A. C. Berg, T. L. Berg, and Y. Choi. Collective generation of natural image descriptions. In *ACL*, 2012. 2

[24] R. Lau, R. Rosenfeld, and S. Roukos. Trigger-based language models: A maximum entropy approach. In *ICASSP*, 1993. 4

[25] 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. 2

[26] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 2, 6

[27] 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. 2

[28] O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. *NIPS*, 1998. 2

[29] T. Mikolov, A. Deoras, D. Povey, L. Burget, and J. Cernocky. Strategies for training large scale neural network language models. In *ASRU*, 2011. 4

[30] 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. 2, 7, 8

[31] A. Mnih and G. Hinton. Three new graphical models for statistical language modelling. In *ICML*, 2007. 4

[32] A. Mnih and Y. W. Teh. A fast and simple algorithm for training neural probabilistic language models. In *ICML*, 2012. 4

[33] F. J. Och. Minimum error rate training in statistical machine translation. In *ACL*, 2003. 2, 5

[34] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, 2014. 3

[35] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*, 2011. 2

[36] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002. 2, 7, 11

[37] C. Rashtchian, P. Young, M. Hodosh, and J. Hockenmaier. Collecting image annotations using Amazon's mechanical

turk. In *NAACL HLT Workshop Creating Speech and Language Data with Amazon's Mechanical Turk*, 2010. 2, 6, 8

[38] A. Ratnaparkhi. Trainable methods for surface natural language generation. In *NAACL*, 2000. 4

[39] A. Ratnaparkhi. Trainable approaches to surface natural language generation and their application to conversational dialog systems. *Computer Speech & Language*, 16(3):435–455, 2002. 2

[40] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. CNN features off-the-shelf: an astounding baseline for recognition. *arXiv preprint arXiv:1403.6382*, Mar. 2014. 5

[41] Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil. A latent semantic model with convolutional-pooling structure for information retrieval. In *CIKM*, 2014. 5, 11

[42] R. Socher, Q. Le, C. Manning, and A. Ng. Grounded compositional semantics for finding and describing images with sentences. In *NIPS Deep Learning Workshop*, 2013. 2

[43] H. O. Song, R. Girshick, S. Jegelka, J. Mairal, Z. Harchaoui, and T. Darrell. On learning to localize objects with minimal supervision. In *ICML*, 2014. 2, 3

[44] N. Srivastava and R. Salakhutdinov. Multimodal learning with deep boltzmann machines. In *NIPS*, 2012. 2

[45] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders. Selective search for object recognition. *IJCV*, 104(2):154–171, 2013. 2

[46] Y. Yang, C. L. Teo, H. Daumé III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *EMNLP*, 2011. 1, 2

[47] 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. 2

[48] C. Zhang, J. C. Platt, and P. A. Viola. Multiple instance boosting for object detection. In *NIPS*, 2005. 2, 3

[49] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014. 2, 3

[50] C. L. Zitnick and D. Parikh. Bringing semantics into focus using visual abstraction. In *CVPR*, 2013. 1