What Value Do Explicit High Level Concepts Have in Vision to Language Problems?

Qi Wu, Chunhua Shen, Lingqiao Liu, Anthony Dick, Anton van den Hengel
School of Computer Science, The University of Adelaide, Australia
{qi.wu01,chunhua.shen,lingqiao.liu,anthony.dick,anton.vandenhengel}@adelaide.edu.au

Abstract

Much recent progress in Vision-to-Language (V2L) problems has been achieved through a combination of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). This approach does not explicitly represent high-level semantic concepts, but rather seeks to progress directly from image features to text. In this paper we investigate whether this direct approach succeeds due to, or despite, the fact that it avoids the explicit representation of high-level information. We propose a method of incorporating high-level concepts into the successful CNN-RNN approach, and show that it achieves a significant improvement on the state-of-the-art in both image captioning and visual question answering. We also show that the same mechanism can be used to introduce external semantic information and that doing so further improves performance. We achieve the best reported results on both image captioning and VQA on several benchmark datasets, and provide an analysis of the value of explicit high-level concepts in V2L problems.

1. Introduction

Vision-to-Language problems present a particular challenge in Computer Vision because they require translation between two different forms of information. In this sense the problem is similar to that of machine translation between languages. In machine language translation there have been a series of results showing that good performance can be achieved without developing a higher-level model of the state of the world. In [3, 7, 47], for instance, a source sentence is transformed into a fixed-length vector representation by an 'encoder' RNN, which in turn is used as the initial hidden state of a 'decoder' RNN that generates the target sentence.

Despite the supposed equivalence between an image and 1000 words, the manner in which information is represented in each data form could hardly be more different. Human language is designed specifically so as to communicate information between humans, whereas even the most carefully composed image is the culmination of a complex set of physical processes over which humans have little control. Given the differences between these two forms of information, it seems surprising that methods inspired by machine language translation have been so successful. These RNN-based methods which translate directly from image features to text, without developing a high-level model of the state of the world, represent the current state of the art for key Vision-to-Language (*V2L*) problems, such as image captioning and visual question answering.

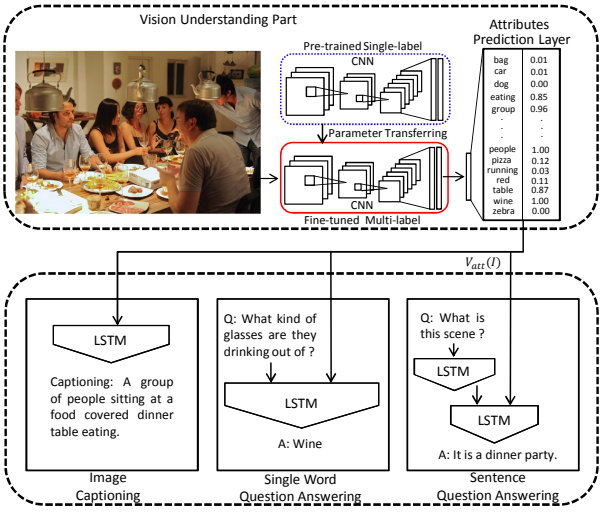

Figure 1. Our attribute based *V2L* framework. The image analysis module learns a mapping between an image and the semantic attributes through a CNN. The language module learns a mapping from the attributes vector to a sequence of words using an LSTM.

This approach is reflected in many recent successful works on image captioning, such as [6, 10, 23, 36, 50, 55]. Current state-of-the-art captioning methods use a CNN as an image 'encoder' to produce a fixed-length vector representation [25, 29, 45, 48], which is then fed into the 'decoder' RNN to generate a caption.

Visual Question Answering (VQA) is a more recent challenge than image captioning. In this *V2L* problem an image and a free-form, open-ended question about the image are

presented to the method which is required to produce a suitable answer [2]. Same as image captioning, the current state of the art in VQA [13, 35, 43] relies on passing CNN features to an RNN language model.

Our main contribution is to consider the question: *what value do explicit high level concepts have in V2L problems?* That is, given that significant performance improvements have been achieved by moving to models which directly pass from image features to text, should we give up on high-level concepts in *V2L* altogether? We investigate particularly the impact that adding high-level information to the CNN-RNN framework has upon performance. We do this by inserting an explicit representation of attributes of the scene which are meaningful to humans. Each semantic attribute corresponds to a word mined from the training image descriptions, and represents higher-level knowledge about the content of the image. A CNN-based classifier is trained for each attribute, and the set of attribute likelihoods for an image forms a high-level representation of image content. An RNN is then trained to generate captions, or answer questions, on the basis of the likelihoods.

Our second contribution is a fully trainable attribute based neural network that can be applied to multiple *V2L* problems which yields significantly better performance than current state-of-the-art approaches. For example, in the Microsoft COCO Captioning Challenge, we produce a BLEU-1 score of 0.73, which is the state of the art on the leaderboard at the time of writing. Our final model also provides the state-of-the-art performance on several recently released VQA datasets. For instance, our system yields a WUPS@0.9 score of 71.15, compared with the current state of the art of 66.78, on the Toronto COCO-QA single word question answering dataset. On the VQA (test-standard), an open-answer task dataset, our method achieves 55.84% accuracy, while the baseline is 54.06%. Moreover, with an expansion from image-sourced attributes to knowledge-sourced through WordNet (see Section 5.3), we further improve the accuracy to 57.62%.

2. Related Work

Image Captioning The problem of annotating images with natural language at the scene level has long been studied in both computer vision and natural language processing. Hodosh *et al*. [17] proposed to frame sentence-based image annotation as the task of ranking a given pool of captions. Similarly, [15, 19, 40] posed the task as a retrieval problem, but based on co-embedding of images and text in the same space. Recently, Socher *et al*. [46] used neural networks to co-embed image and sentences together and Karpathy *et al*. [23] co-embedded image crops and sub-sentences. Neither attempted to generate novel captions.

Attributes have been used in many image captioning methods to fill the gaps in predetermined caption templates. Farhadi *et al*. [12], for instance, used detections to infer a triplet of scene elements which is converted to text using a template. Li *et al*. [30] composed image descriptions given computer vision based inputs such as detected objects, modifiers and locations using web-scale n-grams. A more sophisticated CRF-based method that uses attribute detections beyond triplets was proposed by Kulkarni *et al*. [26]. The advantage of template-based methods is that the resulting captions are more likely to be grammatically correct. The drawback is that they still rely on hard-coded visual concepts and suffer the implied limits on the variety of the output. Instead of using fixed templates, more powerful language models based on language parsing have been developed, such as [1, 27, 28, 39].

Fang *et al*. [11] won the 2015 COCO Captioning Challenge with an approach that is similar to ours in as much as it applies a visual concept (i.e., attribute) detection process before generating sentences. They first learned 1000 independent detectors for visual words based on a multi-instance learning framework and then used a maximum entropy language model conditioned on the set of visually detected words directly to generate captions. Differently, our visual attributes act as a high-level semantic representation for image content which is fed into an LSTM which generates target sentences based on a much larger word vocabulary. More importantly, the success of their model relies on a re-scoring process from a joint image-text embedding space. To what extent the high-level concepts help in image captioning (and other *V2L* tasks) is not discussed in their work. Instead, this is the main focus of this paper.

In contrast to the aforementioned two-stage methods, the recent dominant trend in *V2L* is to use an architecture which connects a CNN to an RNN to learn the mapping from images to sentences directly. Mao *et al*. [36], for instance, proposed a multimodal RNN (m-RNN) to estimate the probability distribution of the next word given previous words and the deep CNN feature of an image at each time step. Similarly, Kiros *et al*. [24] constructed a joint multimodal embedding space using a powerful deep CNN model and an LSTM that encodes text. Karpathy *et al*. [22] also proposed a multimodal RNN generative model, but in contrast to [36], their RNN is conditioned on the image information only at the first time step. Vinyals *et al*. [50] combined deep CNNs for image classification with an LSTM for sequence modeling, to create a single network that generates descriptions of images. Chen *et al*. [6] learned a bi-directional mapping between images and their sentence-based descriptions using RNN. Xu *et al*. [53] proposed a model based on visual attention, as well as You *et al*. [56]. Jia *et al*. [18] applied additional retrieved sentences to guide the LSTM in generating captions. Devlin *et al*. [9] combined both maximum entropy (ME) language model and RNN to generate captions.

Interestingly, this end-to-end CNN-RNN approach ignores the image-to-word mapping which was an essential step in many of the previous image captioning systems detailed above [12, 26, 30, 54]. The CNN-RNN approach has the advantage that it is able to generate a wider variety of captions, can be trained end-to-end, and outperforms the previous approach on the benchmarks. It is not clear, however, what the impact of bypassing the intermediate high-level representation is, and particularly to what extent the RNN language model might be compensating. Donahue *et al*. [10] described an experiment, for example, using tags and CRF models as a mid-layer representation for video to generate descriptions, but it was designed to prove that LSTM outperforms an SMT-based approach [44]. It remains unclear whether the mid-layer representation or the LSTM leads to the success. Our paper provides several well-designed experiments to answer this question.

We thus here show not only a method for introducing a high-level representation into the CNN-RNN framework, and that doing so improves performance, but we also investigate the value of high-level information more broadly in *V2L* tasks. This is of critical importance at this time because *V2L* has a long way to go, particularly in the generality of the images and text it is applicable to.

Visual Question Answering Visual question answering is one of the more challenging, and interesting, *V2L* tasks as it requires answering previously unseen questions about image content [2, 13, 32, 33, 34, 35, 43, 59]. This is as opposed to the vast majority of challenges in Computer Vision in which the question is specified long before the program is written. Both Gao *et al*. [13] and Malinowski *et al*. [35] used RNNs to encode the question and output the answer. Ren *et al*. [43] focused on questions with a single-word answer and formulated the task as a classification problem using an LSTM, and released a single-word answer dataset (Toronto COCO-QA). Ma *et al*. [32] used CNNs to both extract image features and sentence features, and fuse the features together with a multi-modal CNN. Antol *et al*. [2] proposed a large-scale open-ended VQA dataset based on COCO, which is called VQA. They also provided several baseline methods which combined both image features (CNN extracted) and question features (LSTM extracted) to obtain a single embedding and further built a MLP (Multi-Layer Perceptron) to obtain a distribution over answers.

3. An Attribute-based *V2L* Model

Our approach is summarized in Figure 1. The model includes an image analysis part and a language generation part. In the image analysis part, we first use supervised learning to predict a set of attributes, based on words commonly found in image captions. We solve this as a multi-label classification problem and train a corresponding deep CNN by minimizing an element-wise logistic loss function. Secondly, a fixed length vector $V_{att}(I)$ is created for each image I, whose length is the size of the attribute set. Each dimension of the vector contains the prediction probability for a particular attribute. In the language generation part, we apply an LSTM-based sentence generator. Our attribute vector $V_{att}(I)$ is used as an input to this LSTM. For different tasks, we have different language models. For image captioning, we follow [50] to generate sentences from an LSTM; for single-word question answering, as in [43], we use the LSTM as a classifier providing a likelihood for each potential answer; for open-ended question answering, we use an encoder LSTM to encode questions while the second LSTM decoder uses the attribute vector $V_{att}(I)$ to generate a sentence based answer. A baseline model is also implemented for each of the three tasks. In the baseline model, as in [13, 43, 50] we use a pre-trained CNN to extract image features CNN(I) which are fed into the LSTM directly. For the sake of completeness a fine-tuned version of this approach is also implemented. The baseline method is used as a counterpart to verify the effectiveness of the intermediate attribute prediction layer for each task.

3.1. The Attribute Predictor

We first build an attributes vocabulary regardless of the final tasks (*i.e*. image captioning, VQA). Unlike [26, 54], that use a vocabulary from separate hand-labeled training data, our semantic attributes are extracted from training captions and can be any part of speech, including object names (nouns), motions (verbs) or properties (adjectives). The direct use of captions guarantees that the most salient attributes for an image set are extracted. We use the c most common words in the training captions to determine the attribute vocabulary. In contrast to [11], our vocabulary is not tense or plurality sensitive (done manually), for instance, `'ride'` and `'riding'` are classified as the same semantic attribute, similarly `'bag'` and `'bags'`. This significantly decreases the size of our attribute vocabulary. We finally obtain a vocabulary with 256 attributes. Our attributes represent a set of high-level semantic constructs, the totality of which the LSTM then attempts to represent in sentence form. Generating a sentence from a vector of attribute likelihoods exploits a much larger set of candidate words which are learned separately (see Section 3.2 for more details).

Given this attribute vocabulary, we can associate each image with a set of attributes according to its captions. We then wish to predict the attributes given a test image. Because we do not have ground truth bounding boxes for attributes, we cannot train a detector for each using the standard approach. Fang *et al*. [11] solved a similar problem using a Multiple Instance Learning framework [58] to detect visual words from images. Motivated by the relatively small number of times that each word appears in a caption, we instead treat this as a multi-label classification problem.

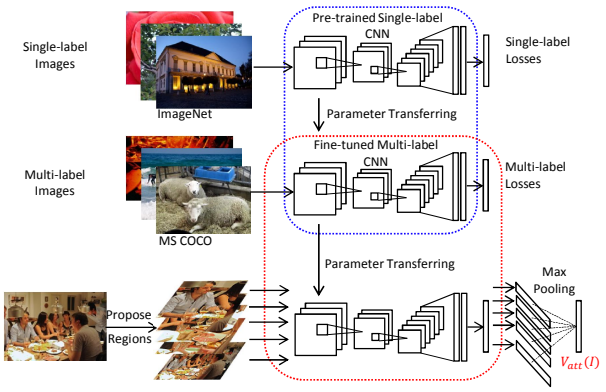

Figure 2. Attribute prediction CNN: the model is initialized from VggNet [45] pre-trained on ImageNet. The model is then fine-tuned on the target multi-label dataset. Given a test image, a set of proposal regions are selected and passed to the shared CNN, and finally the CNN outputs from different proposals are aggregated with max pooling to produce the final multi-label prediction, which gives us the high-level image representation, $V_{att}(I)$

To address the concern that some attributes may only apply to image sub-regions, we follow Wei *et al.* [51] in designing a region-based multi-label classification framework.

Figure 2 summarizes the attribute prediction network. In contrast to [51], which uses AlexNet [25] as the initialization of the shared CNN, we use the more powerful VggNet [45] pre-trained on ImageNet [8]. This model has been widely used in image captioning tasks [6, 11, 22, 36]. The shared CNN is then fine-tuned on the target multi-label dataset (our image-attribute training data). In this step, the output of the last fully-connected layer is fed into a c-way softmax. The $c = 256$ here represents the attribute vocabulary size. In contrast to [51] who employs the squared loss, we find that element-wise logistic loss function performs better. Suppose that there are N training examples and $\boldsymbol{y_i} = [y_{i1}, y_{i2}, ..., y_{ic}]$ is the label vector of the i^{th} image, where $y_{ij} = 1$ if the image is annotated with attribute j, and $y_{ij} = 0$ otherwise. If the predictive probability vector is $\boldsymbol{p_i} = [p_{i1}, p_{i2}, ..., p_{ic}]$, then the cost function to be minimized is

$$J = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{c}\log(1+\exp(-y_{ij}p_{ij})) \qquad (1)$$

During the fine-tuning process, the parameters of the last fully connected layer (i.e. the attribute prediction layer) are initialized with a Xavier initialization [14]. The learning rates of '`fc6`' and '`fc7`' of the VggNet are initialized as 0.001 and the last fully connected layer is initialized as 0.01. All the other layers are fixed during training. We executed 40 epochs in total and decreased the learning rate to one tenth of the current rate for each layer after 10 epochs. The momentum is set to 0.9. The dropout rate is set to 0.5.

To predict attributes based on regions, we first extract hundreds of proposal windows from an image. However, considering the computational inefficiency of deep CNNs, the number of proposals processed needs to be small. Similar to [51], we first apply the normalized cut algorithm to group the proposal bounding boxes into m clusters based on the IoU scores matrix. The top k hypotheses in terms of the predictive scores reported by the proposal generation algorithm are kept and fed into the shared CNN. In contrast to [51], we also include the whole image in the hypothesis group. As a result, there are $mk + 1$ hypotheses for each image. We set $m = 10, k = 5$ in all experiments. We use Multiscale Combinatorial Grouping (MCG) [42] for the proposal generation. Finally, a cross hypothesis max-pooling is applied to integrate the outputs into a single prediction vector $V_{att}(I)$.

3.2. Language Generator

Similar to [22, 36, 50], we propose to train a language generation model by maximizing the probability of the correct description given the image. However, rather than using image features directly as in typically the case, we use the semantic attribute prediction probability $V_{att}(I)$ from the previous section as the input. Suppose that $\{S_1, ..., S_L\}$ is a sequence of words. The log-likelihood of the words given their context words and the corresponding image can be written as:

$$\log p(S|V_{att}(I)) = \sum_{t=1}^{L}\log p(S_t|S_{1:t-1}, V_{att}(I)) \qquad (2)$$

where $p(S_t|S_{1:t-1}, V_{att}(I))$ is the probability of generating the word S_t given attribute vector $V_{att}(I)$ and previous words $S_{1:t-1}$. We employ the LSTM [16], a particular form of RNN, to model this. See Figure 3 for different language generators designed for multiple *V2L* tasks.

Image Captioning Model The LSTM model for image captioning is trained in an unrolled form. More formally, the LSTM takes the attributes vector $V_{att}(I)$ and a sequence of words $S = (S_0, ..., S_L, S_{L+1})$, where S_0 is a special start word and S_{L+1} is a special END token. Each word has been represented as a one-hot vector S_t of dimension equal to the size of words dictionary. The words dictionaries are built based on words that occur at least 5 times in the training set, which lead to 8791 words on MS COCO datasets. Note it is different from the semantic attributes vocabulary $\mathcal{V}_{att}$. The training procedure is as following (see Figure 3 (a)) : At time step $t = -1$, we set $x_{-1} = W_{ea}V_{att}(I)$ and $h_{initial} = \vec{0}$, where W_{ea} is the learnable attributes embedding weights. The LSTM memory state is initialized to the range $(-0.1, 0.1)$ with a uniform distribution. This gives us an initial LSTM hidden state h_{-1} which can be used in the next time step. From $t = 0$ to $t = L$, we set $x_t = W_{es}S_t$ and the hidden state h_{t-1} is given by the previous step, where W_{es} is the learnable word embedding

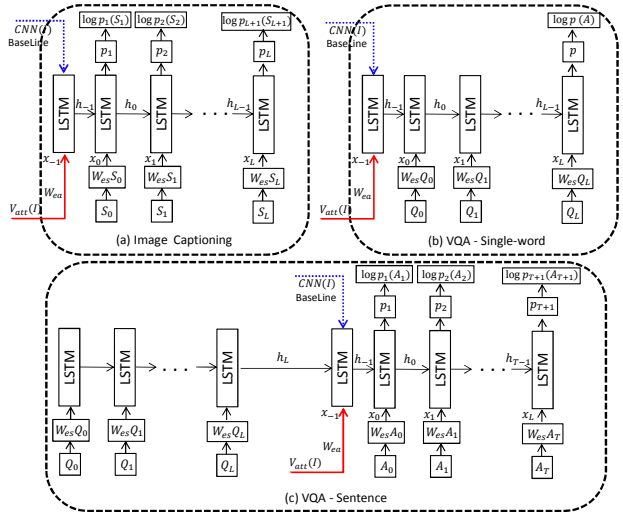

Figure 3. Language generators for different types of tasks: (a) Image Captioning, (b) VQA-single word, (c) VQA-sentence. red arrow indicates our attributes input $V_{att}(I)$ while blue dash arrow shows the baseline method input CNN(I).

weights. The probability distribution p_{t+1} over all words is then computed by the LSTM feed-forward process. Finally, on the last step when S_{L+1} represents the last word, the target label is set to the END token.

Our training objective is to learn parameters W_{ea}, W_{es} and all parameters in LSTM by minimizing the following cost function:

$$\mathcal{C} = -\frac{1}{N}\sum_{i=1}^{N}\log p(S^{(i)}|V_{att}(I^{(i)})) + \lambda_{\boldsymbol{\theta}}\cdot||\boldsymbol{\theta}||_2^2 \quad (3)$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{L^{(i)}+1}\log p_t(S_t^{(i)}) + \lambda_{\boldsymbol{\theta}}\cdot||\boldsymbol{\theta}||_2^2 \quad (4)$$

where N is the number of training examples and $L^{(i)}$ is the length of the sentence for the i-th training example. $p_t(S_t^{(i)})$ corresponds to the activation of the Softmax layer in the LSTM model for the i-th input and $\boldsymbol{\theta}$ represents model parameters, $\lambda_{\boldsymbol{\theta}}\cdot||\boldsymbol{\theta}||_2^2$ is a regularization term. We use SGD with mini-batches of 100 image-sentence pairs. The attributes embedding size, word embedding size and hidden state size are all set to 256 in all the experiments. The learning rate is set to 0.001 and clip gradients is 5. The dropout rate is set to 0.5.

Question Answering Model For question answering, a triplet $\{V_{att}(I), \{Q_1, ..., Q_L\}, \{A_1, ..., A_T\}\}$ is given, whereas L and T is the length of the question and answer, separately. We define it to be a single-word answering problem when $T = 1$ and a sentence-based problem if $T > 1$.

For the single-word answering problem, the LSTM takes the attributes score vector $V_{att}(I)$ and a sequence of input words of the question $Q = (Q_1, ..., Q_L)$. The feed-forward process is the same as image captioning, except that an END token is not required anymore. Instead, we use the word generated by the last word of the question as the predicted answer (see Figure 3 (b)). Hence, the cost function is $\mathcal{C} = -\frac{1}{N}\sum_{i=1}^{N}\log p(A^{(i)}) + \lambda_{\boldsymbol{\theta}}\cdot||\boldsymbol{\theta}||_2^2$, where N is the number of training examples. $\log p(A^{(i)})$ is the log-probability distribution over all candidate answers that is computed by the last LSTM cell, given the previous hidden state and the last word of question Q_L.

For the sentence-based question answering, we have a question encoding LSTM and an answer decoding LSTM. However, different from Gao *et al*. [13] using two separates LSTMs for question and answer, weights between our encoding and decoding LSTMs are shared. The information stored in the LSTM memory cells of the last word in the question is treated as the representation of the sentence. And its hidden state will be used as the initial state of the answering LSTM part. Moreover, different from [13, 35, 43] who use CNN features directly, we use our attributes representations $V_{att}(I)$ as the input for decoding LSTM (see Figure 3 (c)). The cost function of sentence-based question answering is $\mathcal{C} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T^{(i)}+1}\log p_t(A_t^{(i)}) + \lambda_{\boldsymbol{\theta}}\cdot||\boldsymbol{\theta}||_2^2$, where $T^{(i)}+1$ is the length of the answer plus one END token for the i-th training example. According to training configuration, the learning rate is set to 0.0005 and other parameters are same as image captioning configuration.

4. Image Captioning

4.1. Dataset

There are several datasets which consist of images and sentences describing them in English. We mainly report results on the popular Microsoft COCO [31] dataset. Results on Flickr8k [17] and Flickr30k [57] can be found in the supplementary material. MS COCO contains 123,287 images, and each image is annotated with 5 sentences. Because most previous work in image captioning [10, 11, 22, 36, 50, 53] is not evaluated on the official test split of MS COCO, for fair comparison, we report results with the widely used publicly available splits in the work of [22], which use 5000 images for validation, and 5000 for testing. We further tested on the actual MS COCO test set consisting of 40775 images (human captions for this split are not available publicly), and evaluated them on the COCO evaluation server.

4.2. Evaluation

Metrics We report results with the frequently used BLEU metric and sentence perplexity ($\mathcal{PPL}$). BLEU [41] scores are originally designed for automatic machine translation where they measure the fraction of n-grams (up to 4-gram) that are in common between a hypothesis and a reference or set of references. Here we compare against 5 references. Perplexity is a standard measure for evaluating language

models which measures how many bits on average would be needed to encode each word given the language model, so a low $\mathcal{PPL}$ means a better language model. Additionally, we evaluate our model based on the metrics METEOR [4], and CIDEr [49]. All scores (except $\mathcal{PPL}$) are computed with the `coco-evaluation` code [5].

Baselines To verify the effectiveness of our attribute representation, we provide a baseline method. The baseline framework is the same as that proposed in section 3.2, except that the attributes vector $V_{att}(I)$ is replaced by the last hidden layer of CNN directly (see the blue arrow in Figure 3). Various CNN architectures are applied in the baseline method to extract image features, such as VggNet[45] and GoogLeNet[48]. For the **VNet+LSTM**, we use the second fully connected layer (`fc7`), which has 4096 dimensions. In **VNet-PCA+LSTM**, PCA is applied to decrease the feature dimension from 4096 to 1000. For the **GNet+LSTM**, we use the GoogleNet model provided in the Caffe Model Zoo [20] and the last average pooling layer is employed, which is a 1024-d vector. **VNet+ft+LSTM** applies a VggNet that has been fine-tuned on the target dataset, based on the task of image-attributes classification.

Our Approaches We evaluate several variants of our approach: **Att-GT+LSTM** models use ground-truth attributes as the input while **Att-CNN+LSTM** uses the attributes vector $V_{att}(I)$ predicted by the attributes prediction network in section 3.1. We also evaluate an approach **Att-SVM+LSTM** with linear SVM ($C = 1$) predicted attributes vector. SVM classifiers are trained to divide positive attributes from those negatives given an image-attributes correspondence. We use the second fully connected layer of the fine-tuned VggNet to feed the SVM. To infer the sentence given an input image, we use Beam Search, which iteratively considers the set of b best sentences up to time t as candidates to generate sentences at time $t + 1$, and only keeps the best b results. We set the b as 5.

Results Table 1 reports image captioning results on the COCO. It is not surprising that **Att-GT+LSTM** model performs best, since ground truth attributes labels are used. We report the results just to show the advances of adding an intermediate image-to-word mapping stage. Ideally, if we are able to train a strong attributes predictor which gives us a good enough estimation of attributes, we could obtain an outstanding improvement comparing with both baselines and state-of-the-arts. Indeed, apart from using ground truth attributes, our **Attributes-CNN+LSTM** models generate the best results over all evaluation metrics. Especially comparing with baselines, which do not contain an attributes prediction layer, our final models bring significant improvements, nearly 15% for B-1 and 30% for CIDEr on average. **VNet+ft+LSTM** model performs better than other baselines because of the fine-tuning on the target dataset. However, they do not perform as good as our attributes-based models. **Att-SVM+LSTM** under-performs **Att-CNN+LSTM** means our region-based attributes prediction network performs better than the SVM classifier. Our final model also outperforms current state of the arts listed in tables. We also evaluate an approach that combines CNN features and attributes vector together as the input of the LSTM, but we find this approach (B-1=0.71) is not as good as using attributes vector alone in the same setting. In any case, above experiments show that an intermediate image-to-words stage (i.e. attributes prediction layer) brings us significant improvements. Results on Flickr8k and Flickr30k can be found in the supplementary material, as well as some qualitative results.

We further generated captions for the images in the COCO test set containing 40,775 images and evaluated them on the COCO evaluation server. These results are shown in Table 2. We achieve 0.73 on B-1, and surpass human performances on 13 of the 14 metrics reported. We are the best results on 3 evaluations metrics (B-1,2,3) on the

State-of-art	B-1	B-2	B-3	B-4	M	C	$\mathcal{P}$
NeuralTalk [22]	0.63	0.45	0.32	0.23	0.20	0.66	-
Mind's Eye [6]	-	-	-	0.19	0.20	-	11.60
NIC [50]	-	-	-	0.28	0.24	0.86	-
LRCN [10]	0.67	0.49	0.35	0.25	-	-	-
Mao et al.[36]	0.67	0.49	0.34	0.24	-	-	13.60
Jia et al.[18]	0.67	0.49	0.36	0.26	0.23	0.81	-
MSR [11]	-	-	-	0.26	0.24	-	18.10
Xu et al.[53]	0.72	0.50	0.36	0.25	0.23	-	-
Jin et al.[21]	0.70	0.52	0.38	0.28	0.24	0.84	-
Baseline-*CNN(I)*							
VNet+LSTM	0.61	0.42	0.28	0.19	0.19	0.56	13.58
VNet-PCA+LSTM	0.62	0.43	0.29	0.19	0.20	0.60	13.02
GNet+LSTM	0.60	0.40	0.26	0.17	0.19	0.55	14.01
VNet+ft+LSTM	0.68	0.50	0.37	0.25	0.22	0.73	13.29
Ours-$V_{att}(I)$							
Att-GT+LSTM‡	0.80	0.64	0.50	0.40	0.28	1.07	9.60
Att-SVM+LSTM	0.69	0.52	0.38	0.28	0.23	0.82	12.62
Att-CNN+LSTM	**0.74**	**0.56**	**0.42**	**0.31**	**0.26**	**0.94**	**10.49**

Table 1. BLEU-1,2,3,4, METEOR, CIDEr and $\mathcal{PPL}$ metrics compared with other state-of-the-art methods and our baseline on MS COCO dataset. ‡ indicates ground truth attributes labels are used, which (in gray) will not participate in rankings.

COCO-TEST	B-1	B-2	B-3	B-4	M	R	CIDEr
5-Refs							
Ours	**0.73**	**0.56**	**0.41**	**0.31**	**0.25**	**0.53**	**0.92**
Human	0.66	0.47	0.32	0.22	**0.25**	0.48	0.85
MSR [11]	0.70	0.53	0.39	0.29	**0.25**	0.52	0.91
m-RNN [36]	0.68	0.51	0.37	0.27	0.23	0.50	0.79
LRCN [10]	0.70	0.53	0.38	0.28	0.24	0.52	0.87
40-Refs							
Ours	**0.89**	**0.80**	**0.69**	**0.58**	0.33	**0.67**	**0.93**
Human	0.88	0.74	0.63	0.47	**0.34**	0.63	0.91
MSR [11]	0.88	0.79	0.68	0.57	0.33	0.66	**0.93**
m-RNN [36]	0.87	0.76	0.64	0.53	0.30	0.64	0.79
LRCN [10]	0.87	0.77	0.65	0.53	0.32	0.66	0.89

Table 2. COCO evaluation server results. M and R stands for METEOR and ROUGE-L. Results using 5 references and 40 references captions are both shown. We only list the comparison results that have been officially published in the corresponding references.

	Ours	NIC[50]	LRCN[10]	m-RNN[36]	NeuralTalk[22]
VIS Input Dim	256	1000	1000	4096	4096
RNN Dim	256	512	1000×4	256	300-600

Table 3. Visual feature input dimension and properties of RNN. Our visual features has been encoded as a 256-d attributes score vector while other models need higher dimensional features to feed to RNN. According to the unit size of RNN, we achieve state-of-the-art using a relatively small dimensional recurrent layer.

server leaderboard at the time of writing this paper. We also achieve the top-5 ranking on the other evaluation metrics.

Table 3 summarizes some properties of recurrent layers employed in some recent RNN-based methods. We achieve state-of-the-art using a relatively small dimensional visual input feature and recurrent layer. Lower dimension of visual input and RNN normally means less parameters in the RNN training stage, as well as lower computation cost.

5. Visual Question Answering

5.1. Dataset

We report VQA results on two recently publicly available visual question answering datasets, both are created based on MS COCO. Toronto COCO-QA dataset [43] contains four types of questions, specifically the object, number, color and location. The answers are all single-word. We use this dataset to examine our single-word question answering model. VQA [2] is a much larger dataset which contains 614,163 questions. These questions and answers are sentence-based and open-ended. The training and testing split follows COCO official split, which contains 82,783 training images, 40,504 validation images and 81,434 test images, each has 3 questions and 10 answers. We use the official test split for our testing.

5.2. Evaluation

Our experiments in question answering are designed to verify the effectiveness of introducing the intermediate attribute layer. Hence, apart from listing several state of art methods, we focus on comparing with a baseline method, which only uses the second fully connected layer (fc7) of the VggNet (and a fine-tuned VggNet) as the input.

Table 4 reports results on the Toronto COCO-QA dataset, within which all answers are a single-word. Besides the accuracy value (the proportion of correct answered testing questions to the total testing questions), the Wu-Palmer similarity (WUPS) [52] is also used to measure the performance of different models. The WUPS calculates the similarity between two words based on the similarity between their common subsequence in the taxonomy tree. If the similarity between two words is greater than a threshold then the candidate answer is assumed to be right. We follow [32, 43] in setting the threshold as 0.9 and 0.0. **GUESS** is a simple baseline to predict the most common answer

Toronto COCO-QA	Acc	WUPS@0.9	WUPS@0.0
GUESS[43]	6.65	17.42	73.44
VIS+BOW[43]	55.92	66.78	88.99
VIS+LSTM[43]	53.31	63.91	88.25
2-VIS+BLSTM[43]	55.09	65.34	88.64
Ma et al.[32]	54.94	65.36	88.58
BaseLine			
VggNet-LSTM	50.73	60.37	87.48
VggNet+ft-LSTM	58.34	67.32	89.13
Our-Proposal			
Att-GT+LSTM‡	67.66	75.76	93.63
Att-CNN+LSTM	**61.38**	**71.15**	**91.58**

Table 4. Accuracy, WUPS@0.9 and WUPS@0.0 metrics compared with other state-of-the-art methods and our baseline on the Toronto COCO-QA dataset. Each image has one question and only a single word answer is given for each. ‡ indicates that ground truth attributes labels were used, and thus that the method does not participate in rankings.

from the training set based on the question type. The modes are 'cat', 'two', 'white', and 'room' for the four types of questions. **VIS+BOW** [43] performs multinomial logistic regression based on image features and a BOW vector obtained by summing all the word vectors of the question. **VIS+LSTM** [43] has one LSTM to encode the image and question, while **2-VIS+BLSTM** has two image feature input points, at the start and the end of the sentences. Ma *et al.* [32] encoded both images and questions by CNN. From the Table 4, we clearly see that our attribute-based model outperforms the baselines and all state-of-the-art methods by a significant degree, which proves the effectiveness of our attribute-based representation for *V2L* tasks.

Table 5 summarizes the results on the test split of VQA dataset. In contrast to the above single-word question answering task, here we follow [2], and measure performance by recording the percentage of answers in agreement with ground truth from human subjects. Antol *et al*. [2] provided a baseline for this dataset using a **Q+I** method, which encodes the image with CNN features and questions with LSTM representation. Then they train a softmax neural network classifier with a single hidden layer and the output space is the 1000 most frequent answers in the training set. Human performance is also given in [2] for reference. **VNet+ft+LSTM** is the model with fine-tuned VggNet features. It is slightly less accurate than our explicit attributes based model **Att-CNN+LSTM**, but the gap is small. **LSTM Q+I** [2] can be treated as our baseline

	Test-dev				Test-standard			
	All	Y/N	Num	Others	All	Y/N	Num	Others
Q+I [2]	52.64	75.55	33.67	37.37	-	-	-	-
LSTM Q [2]	48.76	78.20	35.68	26.59	48.89	78.12	34.94	26.99
LSTM Q+I [2]	53.74	78.94	35.24	36.42	54.06	79.01	35.55	36.80
Human [2]	-	-	-	-	83.30	95.77	83.39	72.67
VNet+ft+LSTM	55.03	78.19	35.47	39.68	55.34	78.10	35.30	40.27
Att-CNN+LSTM	55.57	78.90	36.11	40.07	55.84	78.73	**36.08**	40.60
Att-KB+LSTM	**57.46**	**79.77**	**36.79**	**43.10**	**57.62**	**79.72**	36.04	**43.44**

Table 5. Results on test-dev and test-standard split of VQA dataset compared with [2].

as it uses CNN features as the input to the LSTM, while **LSTM Q** only provides questions as the input. Our attributes based model outperforms **LSTM Q+I** nearly in all cases, especially when the answer types are 'others'. Our hypothesis is that this performance increase occurs because the separately-trained attribute layer discards irrelevant image information. This ensures that the LSTM does not interpret irrelevant variations in the expression of the text as relating to irrelevant image details, and try to learn a mapping between them.

However, there is still a big gap between our proposed models and the human performance. After looking into details, we notice that accuracies on some question types such as 'why' are very low. These kinds of questions are hard to answer because commonsense knowledge and reasoning is normally required. Zhu *et al.* [59] cast a MRF model into a Knowledge Base representation to answer commonsense-related visual questions. Our semantic attribute representation offers hope of a solution, however, as it can be used as a key by which to source other, external information. In the following experiment, we propose to expand our image-based attributes set to a knowledge-based attributes set through a large lexical ontology - the WordNet.

5.3. Attribute Expansion using WordNet

WordNet [38] records a variety of relationships between words, some of which we hope to use to address the many ways of expressing the same idea in natural language. The most frequently encoded relation is the hyponymy (such as `bed` and `bunkbed`). Meronymy represents the part-whole relation. Verb synsets are arranged into hierarchies (troponyms) (such as `buy-pay`). All these relationships are defined based on commonsense knowledge.

To expand our image-sourced attributes to knowledge-sourced information, we first select candidate words from WordNet. Candidate words must fulfill two selection criteria. The first is that the word must directly linked with an arbitrary word in our attribute vocabulary $\mathcal{V}_{att}$ through the WordNet. Secondly, the candidate word must appear in at least 5 training question examples. In our experiment, given $M = 256$ image-sourced attributes, we finally mined a knowledge-sourced vocabulary $\mathcal{V}_{kb}$ with $N = 9762$ words, and $\mathcal{V}_{kb}$ has covered all the words in $\mathcal{V}_{att}$. Then, a similarity matrix $S \in \mathbb{R}^{M \times N}$ is computed based on a pre-trained word2vec model [37], where S_{ij} gives both semantic and syntactic similarity between word i in $\mathcal{V}_{att}$ and word j in $\mathcal{V}_{kb}$. Given an image I and its image-sourced attribute vector $V_{att}(I) = (v_{att}^{(1)}, ..., v_{att}^{(i)}, ..., v_{att}^{(M)})$ predicted by the attribute prediction network, the j^{th} component of the knowledge-sourced attribute vector is obtained by a max-pooling operator $v_{kb}^{(j)} = \max(v_1^{(j)}, ..., v_i^{(j)}, ..., v_M^{(j)})$, where $v_i^{(j)} = v_{att}^{(i)} \times S_{ij}$. The final knowledge-sourced attributes vector $V_{kb}(I) = (v_{kb}^{(1)}, ..., v_{kb}^{(j)}, ..., v_{kb}^{(N)})$ will be fed into the LSTM to generate answers.

Question-Type	Vgg+LSTM	Att-CNN+LSTM	Att-KB+LSTM
why	3.04	7.77	9.88
what kind	24.15	41.22	45.23
which	31.28	36.60	37.28
is the	71.49	73.22	74.59
is this	73.00	75.26	76.63

Table 6. Results on the open-answer task for some commonsense reasoning question types on validation split of VQA.

Table 6 compares results using image-sourced attributes vs. knowledge-sourced on the validation split of VQA dataset. We gain a significant improvement in commonsense reasoning related questions. For example, on the 'why' questions, we achieve 9.88%. Our hypothesis is that this reflects the fact that indexing into WordNet in this manner provides some independence as to the exact manner of expression used in the text, but also adds extra information. In answering questions about beds and hammocks, for example, it is useful to know that both are related to sleep. The overall performance of this **Att-KB+LSTM** model on the test split of VQA can be found in the Table 5. Our overall result is 57.62% accuracy, which performs better than the model of **Att-CNN+LSTM** (the model before attributes expansion) and achieves the state-of-the-art result on the VQA dataset.

6. Conclusion

We have described an investigation into the value of high level concepts in *V2L* problems, motivated by the belief that without an explicit representation of the content of an image it is very difficult to answer reason about it. In the process we examined the effect of introducing an intermediate attribute prediction layer into the predominant CNN-LSTM framework. We implemented three attribute-based models for the tasks of image captioning, single-word question answering and sentence question answering.

We have shown that an explicit representation of image content improves *V2L* performance, in all cases. Indeed, at the time of writing this paper, our image captioning model outperforms the state of the art on several captioning datasets. Our question answering models perform best on the Toronto COCO-QA datasets, producing an accuracy of 61.38%. It also achieves the state of the art on the VQA, at 57.62%, which is a big improvement over the baseline. Moreover, attribute representation enables access to high-level commonsense knowledge, which is necessary for answering commonsense reasoning related questions.

Acknowledgements This research was in part supported by the Data to Decisions Cooperative Research Centre.

References

[1] A. Aker and R. Gaizauskas. Generating image descriptions using dependency relational patterns. In *Proc. Conf. Associ-*

ation for Computational Linguistics, 2010. 2

[2] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. VQA: Visual Question Answering. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2015. 2, 3, 7

[3] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *Proc. Int. Conf. Learn. Representations*, 2015. 1

[4] S. Banerjee and A. Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 2005. 6

[5] 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:1504.00325*, 2015. 6

[6] X. Chen and C. Lawrence Zitnick. Mind's Eye: A Recurrent Visual Representation for Image Caption Generation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, June 2015. 1, 2, 4, 6

[7] K. Cho, B. van Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Proc. Conf. Empirical Methods in Natural Language Processing*, 2014. 1

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2009. 4

[9] J. Devlin, H. Cheng, H. Fang, S. Gupta, L. Deng, X. He, G. Zweig, and M. Mitchell. Language models for image captioning: The quirks and what works. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2015. 2

[10] 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. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015. 1, 3, 5, 6, 7

[11] 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. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015. 2, 3, 4, 5, 6

[12] 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 *Proc. Eur. Conf. Comp. Vis.* 2010. 2, 3

[13] H. Gao, J. Mao, J. Zhou, Z. Huang, L. Wang, and W. Xu. Are You Talking to a Machine? Dataset and Methods for Multilingual Image Question Answering. In *Proc. Advances in Neural Inf. Process. Syst.*, 2015. 2, 3, 5

[14] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proc. Int. Conf. Artificial Intell. & Stat.*, pages 249–256, 2010. 4

[15] Y. Gong, L. Wang, M. Hodosh, J. Hockenmaier, and S. Lazebnik. Improving image-sentence embeddings using large weakly annotated photo collections. In *Proc. Eur. Conf. Comp. Vis.* 2014. 2

[16] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. 4

[17] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *JAIR*, pages 853–899, 2013. 2, 5

[18] X. Jia, E. Gavves, B. Fernando, and T. Tuytelaars. Guiding Long-Short Term Memory for Image Caption Generation. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2015. 2, 6

[19] Y. Jia, M. Salzmann, and T. Darrell. Learning cross-modality similarity for multinomial data. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2011. 2

[20] 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:1408.5093*, 2014. 6

[21] J. Jin, K. Fu, R. Cui, F. Sha, and C. Zhang. Aligning where to see and what to tell: image caption with region-based attention and scene factorization. *arXiv:1506.06272*, 2015. 6

[22] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015. 2, 4, 5, 6, 7

[23] A. Karpathy, A. Joulin, and F. F. Li. Deep fragment embeddings for bidirectional image sentence mapping. In *Proc. Advances in Neural Inf. Process. Syst.*, 2014. 1, 2

[24] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. In *Proc. Conf. Association for Computational Linguistics*, 2015. 2

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. Advances in Neural Inf. Process. Syst.*, 2012. 1, 4

[26] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Babytalk: Understanding and generating simple image descriptions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(12):2891–2903, 2013. 2, 3

[27] P. Kuznetsova, V. Ordonez, A. C. Berg, T. L. Berg, and Y. Choi. Collective generation of natural image descriptions. In *Proc. Conf. Association for Computational Linguistics*, 2012. 2

[28] P. Kuznetsova, V. Ordonez, T. L. Berg, and Y. Choi. Treetalk: Composition and compression of trees for image descriptions. *Proc. Conf. Association for Computational Linguistics*, 2014. 2

[29] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998. 1

[30] 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, 3

[31] 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. In *Proc. Eur. Conf. Comp. Vis.* 2014. 5

[32] L. Ma, Z. Lu, and H. Li. Learning to Answer Questions From Image using Convolutional Neural Network. In *AAAI*, 2016. 3, 7

[33] M. Malinowski and M. Fritz. A multi-world approach to question answering about real-world scenes based on uncertain input. In *Proc. Advances in Neural Inf. Process. Syst.*, pages 1682–1690, 2014. 3

[34] M. Malinowski and M. Fritz. Hard to Cheat: A Turing Test based on Answering Questions about Images. *arXiv:1501.03302*, 2015. 3

[35] M. Malinowski, M. Rohrbach, and M. Fritz. Ask Your Neurons: A Neural-based Approach to Answering Questions about Images. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2015. 2, 3, 5

[36] J. Mao, W. Xu, Y. Yang, J. Wang, and A. Yuille. Deep Captioning with Multimodal Recurrent Neural Networks (m-RNN). In *Proc. Int. Conf. Learn. Representations*, 2015. 1, 2, 4, 5, 6, 7

[37] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Proc. Advances in Neural Inf. Process. Syst.*, pages 3111–3119, 2013. 8

[38] G. A. Miller. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995. 8

[39] 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

[40] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In *Proc. Advances in Neural Inf. Process. Syst.*, 2011. 2

[41] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proc. Conf. Association for Computational Linguistics*, 2002. 5

[42] J. Pont-Tuset, P. Arbeláez, J. Barron, F. Marques, and J. Malik. Multiscale Combinatorial Grouping for Image Segmentation and Object Proposal Generation. In *arXiv:1503.00848*, March 2015. 4

[43] M. Ren, R. Kiros, and R. Zemel. Image Question Answering: A Visual Semantic Embedding Model and a New Dataset. In *Proc. Advances in Neural Inf. Process. Syst.*, 2015. 2, 3, 5, 7

[44] M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele. Translating video content to natural language descriptions. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2013. 3

[45] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. Int. Conf. Learn. Representations*, 2015. 1, 4, 6

[46] R. Socher, A. Karpathy, Q. V. Le, C. D. Manning, and A. Y. Ng. Grounded compositional semantics for finding and describing images with sentences. *Proc. Conf. Association for Computational Linguistics*, 2014. 2

[47] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Proc. Advances in Neural Inf. Process. Syst.*, 2014. 1

[48] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015. 1, 6

[49] R. Vedantam, C. L. Zitnick, and D. Parikh. CIDEr: Consensus-based Image Description Evaluation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015. 6

[50] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2014. 1, 2, 3, 4, 5, 6, 7

[51] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, and S. Yan. CNN: Single-label to multi-label. *arXiv:1406.5726*, 2014. 4

[52] Z. Wu and M. Palmer. Verbs semantics and lexical selection. In *Proc. Conf. Association for Computational Linguistics*, 1994. 7

[53] K. Xu, J. Ba, R. Kiros, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *Proc. Int. Conf. Mach. Learn.*, 2015. 2, 5, 6

[54] Y. Yang, C. L. Teo, H. Daumé III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *Proc. Conf. Empirical Methods in Natural Language Processing*, 2011. 3

[55] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. Describing videos by exploiting temporal structure. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2015. 1

[56] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo. Image captioning with semantic attention. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, June 2016. 2

[57] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Proc. Conf. Association for Computational Linguistics*, 2, 2014. 5

[58] C. Zhang, J. C. Platt, and P. A. Viola. Multiple instance boosting for object detection. In *Proc. Advances in Neural Inf. Process. Syst.*, 2005. 3

[59] Y. Zhu, C. Zhang, C. Ré, and L. Fei-Fei. Building a Large-scale Multimodal Knowledge Base for Visual Question Answering. *arXiv:1507.05670*, 2015. 3, 8