Image Captioning and Visual Question Answering Based on Attributes and Their Related External Knowledge

Qi Wu, Chunhua Shen, Anton van den Hengel, Peng Wang, Anthony Dick

Abstract—Much recent progress in Vision-to-Language 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 first 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 further show that the same mechanism can be used to incorporate external knowledge, which is critically important for answering high level visual questions. Specifically, we design a visual question answering model that combines an internal representation of the content of an image with information extracted from a general knowledge base to answer a broad range of image-based questions. It particularly allows questions to be asked about the contents of an image, even when the image itself does not contain a complete answer. Our final model achieves the best reported results on both image captioning and visual question answering on several benchmark datasets.

Index Terms—Image Captioning, Visual Question Answering, Concepts Learning, Recurrent Neural Networks, LSTM.

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 [1], [2], [3], 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 a thousand 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.

• *The authors are with the Australian Centre for Visual Technologies; and School of Computer Science, The University of Adelaide, Australia. E-mail: ({qi.wu01, chunhua.shen, anton.vandenhengel, p.wang, anthony.dick}@adelaide.edu.au).*

Fig. 1: An example from our *V2L* system. Attributes are predicted by our CNN-based attributes prediction model. Image captions are generated by our attribute-based captioning generation model. All the predicted attributes and generated captions combined with the mined external knowledge from a large-scale knowledge base are fed to an LSTM to produce the answer to the asked question. Underlined words indicate the information required to answer the question.

This approach is reflected in many recent successful works on image captioning, such as [4], [5], [6], [7], [8], [9], [10]. Current state-of-the-art captioning methods use a CNN as an image 'encoder' to produce a fixed-length vector representation [11], [12], [13], [14], which is then fed into the 'decoder' RNN to generate a caption.

Visual Question Answering (VQA) is a more recent challenge than image captioning. It is distinct from many problems in Computer Vision because the question to be answered is not determined until run time [15]. In more traditional problems such as segmentation or detection, the single question to be answered by an algorithm is predetermined, and only the image changes. In visual question answering, in contrast, the form that the question will take is unknown, as is the set of operations required to answer it. In this sense it more closely reflects the challenge of general image interpretation. 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 [15]. As in image captioning, current state of the art in VQA [16], [17], [18] relies on passing CNN features to an RNN language model. However, visual question answering is a significantly more complex problem than image captioning, not least because it requires accessing information not present in the image. This may be common sense, or specific knowledge about the image subject. For example, given an image, such as Figure 1, showing 'a group of people enjoying a sunny day at the beach with umbrellas', if one asks a question 'why do they have umbrellas?', to answer this question, the machine must not only detect the scene 'beach', but must know that 'umbrellas are often used as points of shade on a sunny beach'. Recently, Antol *et al.* [15] also have suggested that VQA is a more "AI-complete" task since it requires multimodal knowledge beyond a single sub-domain.

Contributions of this paper are two-fold. First, we propose a *fully trainable* attribute-based neural network upon the CNN+RNN architecture, that can be applied to multiple *V2L* problems. 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 form a high-level representation of image content. An RNN is then trained to generate captions, or question answers, on the basis of the likelihoods. Our attributes based model yields significantly better performance than current state-of-the-art approaches in the task of image captioning. For example, in the Microsoft COCO Captioning Challenge, we produce a BLEU-1 score of 0.73, which is the state of the art at the time of writing this paper.

Based on the proposed attribute-based *V2L* model, more impotantly, our second contribution is to introduce external commonsense and knowledge for the visual question answering. In this work, we fuse the automatically generated description of an image with information extracted from an external knowledge base (KB) to provide an answer to a general question about the image (See Figure 5). The image description takes the form of a set of captions, and the external knowledge is text-based information mined from a Knowledge Base. Specifically, for each of the top-k attributes detected in the image we generate a query which may be applied to a Resource Description Framework (RDF) KB, such as DBpedia. RDF is the standard format for large KBs, of which there are many. The queries are specified using Semantic Protocol And RDF Query Language (SPARQL). We encode the paragraphs extracted from the KB using Doc2Vec [19], which maps paragraphs into a fixed-length feature representation. The encoded attributes, captions, and KB information are then input to an LSTM which is trained so as to maximise the likelihood of the ground truth answers in a training set. We further propose a question-guided knowledge selection scheme to improve the quality of the extracted KB information. Those knowledge that is not related to the question is filtered out. The approach that we propose here combines the generality of information that using a KB allows with the generality of questions that the LSTM allows. In addition, it achieves an accuracy of 70.98% on the Toronto COCO-QA [18], while the latest state of the art is 61.60%. On the VQA [15] evaluation server (which does not publish ground truth answers for its test set), we also produce the state-of-the-art result, which is 59.44%. Preliminary results of this paper appeared in [20], [21].

2 Related work

2.1 Attribute-based Representation

Using attribute-based models as a high-level representation has shown potential in many computer vision tasks such as object recognition, image annotation and image retrieval. Farhadi *et al.* [22] are among the first to propose to use a set of visual semantic attributes to identify familiar objects, and to describe unfamiliar objects. Lampert *et al.* [23] showed that semantic attributes can be used to recognize object classes in the absence of training images, known as zero-short learning. Vectors of visual attributes which are predicted using corresponding attribute classifiers were also used to describe faces by Kumar *et al.* [24]. In addition to describing objects semantically, there are several works describing the whole image using semantic features. Vogel and Schiele [25] used visual attributes describing scenes to characterize image regions and combined these local semantics into a global image description. Su *et al.* [26] defined six groups of attributes to build intermediate level features for image classification. Li *et al.* [27], [28] introduced the concept of an 'object bank' which enables objects to be used as attributes for scene representation.

2.2 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.* [29] proposed to frame sentence-based image annotation as the task of ranking a given pool of captions. Similarly, [30], [31], [32] posed the task as a retrieval problem, but based on co-embedding of images and text in the same space. Recently, Socher *et al.* [33] used neural networks to co-embed image and sentences together and Karpathy *et al.* [6] 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.* [34], for instance, used detections to infer a triplet of scene elements which is converted to text using a template. Li *et al.* [35] composed image descriptions given

computer vision based inputs such as detected objects, modifiers and locations using web-scale n-grams. Zhu *et al.* [36] converted image parsing results into a semantic representation in the form of Web Ontology Language, which is converted to human readable text. A more sophisticated CRF-based method use of attribute detections beyond triplets was proposed by Kulkarni *et al* [37]. 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 [38], [39], [40], [41].

Fang *et al.* [42] 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, our work employ several well-designed experiments (Sec 5) prove the value of explicit high-level concept in multiple *V2L* applications.

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.* [7], 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.* [43] constructed a joint multimodal embedding space using a powerful deep CNN model and an LSTM that encodes text. Karpathy and Li [44] also proposed a multimodal RNN generative model, but in contrast to [7], their RNN is conditioned on the image information only at the first time step. Vinyals *et al.* [8] 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.* [4] learn a bi-directional mapping between images and their sentence-based descriptions, which allows to reconstruct visual features given an image description. Xu *et al.* [45] proposed a model based on visual attention. Jia *et al.* [46] applied additional retrieved sentences to guide the LSTM in generating 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 [34], [35], [37], [47]. 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.* [5] 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 [48]. 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.

2.3 Visual Question Answering

Malinowski *et al.* [49] may be the first to study the VQA problem. They proposed a method that combines semantic parsing and image segmentation with a Bayesian approach to sampling from nearest neighbors in the training set. This approach requires human defined predicates, which are inevitably dataset-specific. This approach is also very dependent on the accuracy of the image segmentation algorithm and on the estimated image depth information. Tu *et al.* [50] built a query answering system based on a joint parse graph from text and videos. Geman *et al.* [51] proposed an automatic 'query generator' that is trained on annotated images and produces a sequence of binary questions from any given test image. Each of these approaches places significant limitations on the form of question that can be answered.

Most recently, inspired by the significant progress achieved using deep neural network models in both computer vision and natural language processing, an architecture which combines a CNN and RNN to learn the mapping from images to sentences has become the dominant trend. Both Gao *et al.* [16] and Malinowski *et al.* [17] used RNNs to encode the question and output the answer. Whereas Gao *et al.* [16] used two networks, a separate encoder and decoder, Malinowski *et al.* [17] used a single network for both encoding and decoding. Ren *et al.* [18] focused on questions with a single-word answer and formulated the task as a classification problem using an LSTM. A single-word answer dataset COCO-QA was published with [18]. Ma *et al.* [52] used CNNs to both extract image features and sentence features, and fuse the features together with another multimodal CNN. Antol *et al.* [15] proposed a large-scale open-ended VQA dataset based on COCO, which is called VQA. They also provided a baseline for this dataset using a CNN+BOW method, which encodes the image with CNN features and questions with BOW representation. Inspired by Xu *et al.* [45] who encode visual attention in the Image Captioning, [53], [54], [55], [56], [57], [58] propose to use the spatial attention to help answering visual questions. [54], [58], [59] formulate the VQA as a classification problem and restrict the answer only can be drawn from a fixed answer space. In other words, they can not generate open-ended answers. Zhu *et al.* [60] investigate the video question answering problem using the question form of 'fill-in-the-blank'.

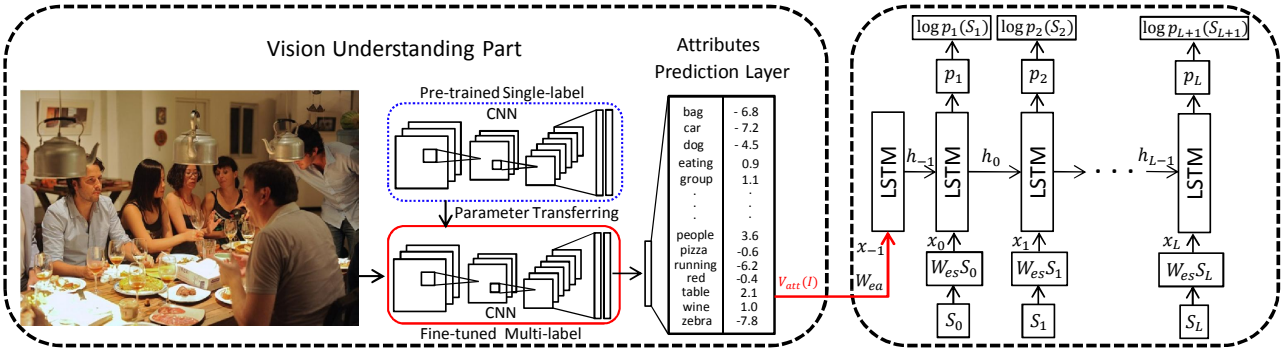


Fig. 2: Our attribute-based image captioning framework. The image analysis module learns a mapping between an image and the semantic attributes through a CNN. The language module learns a mapping from the attributes vector to a sequence of words using an LSTM.

Our framework also exploits both CNN and RNNs, but in contrast to preceding approaches which use only image features extracted from a CNN in answering a question, we employ multiple sources, including image content, generated image captions and mined external knowledge, to feed to an RNN to answer questions. Large-scale Knowledge Bases (KBs), such as Freebase [61] and DBpedia [62], have been used successfully in several natural language Question Answering (QA) systems [63], [64]. However, VQA systems exploiting KBs are still relatively rare.

The quality of the information in the KB is one of the primary issues in this approach to VQA. The problem is that KBs constructed by analysing Wikipedia and similar are patchy and inconsistent at best, and hand-curated KBs are inevitably very topic specific. Using visually-sourced information is a promising approach to solve this problem [65], [66], but has a way to go before it might be usefully applied within our approach. Thus, although our SPARQL and RDF driven approach can incorporate any information that might be extracted from a KB, the limitations of the existing available KBs mean that the text descriptions of the detected attributes is all that can be usefully extracted. Zhu *et al.* [67], in contrast used a hand-crafted KB primarily containing image-related information such as category labels, attribute labels and affordance labels, but also some quantities relating to their specific question format such as GPS coordinates and similar. The questions in that system are phrased in the DBMS query language, and are thus tightly coupled to the nature of the hand-crafted KB. This represents a significant restriction on the form of question that might be asked, but has the significant advantage that the DBMS is able to respond decisively as to whether it has the information required to answer the question.Instead of building a problem-specific KB, we use a pre-built large-scale KB (DBpedia [62]) from which we extract information using a standard RDF query language. DBpedia has been created by extracting structured information from Wikipedia, and is thus significantly larger and more general than a hand-crafted KB. Rather than having a user pose their question in a formal query language, our VQA system is able to encode questions written in natural language automatically. This is achieved without manually specified formalization, but rather depends on processing a suitable training set. The result is a model which is very general in the forms of question that it will accept.

3 Image Captioning using Attributes

Our image captioning model is summarized in Figure 2. The model includes an image analysis part and a captioning 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 [1]. 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 captioning generation part, we apply an LSTM-based sentence generator. In the baseline model, as in [8], [16], [18] 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.

3.1 Attribute-based Image Representation

Our first task is to describe the image content in terms of a set of attributes. An attributes vocabulary is first constructed. Unlike [37], [47], 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 ($c = 256$) most common words in the training captions to determine the attribute vocabulary $\mathcal{V}_{att}$. Similar to [42], the top 15 most frequent closed-class words such as `'a'`, `'on'`, `'of'` are removed since they are in nearly every caption. In contrast to [42], our vocabulary is not tense or plurality sensitive, 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. 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, allowing for greater flexibility in the generated text.

1. Please note that we use image captions to build our attributes vocabulary regardless of the final (*i.e.*captioning, VQA) tasks.

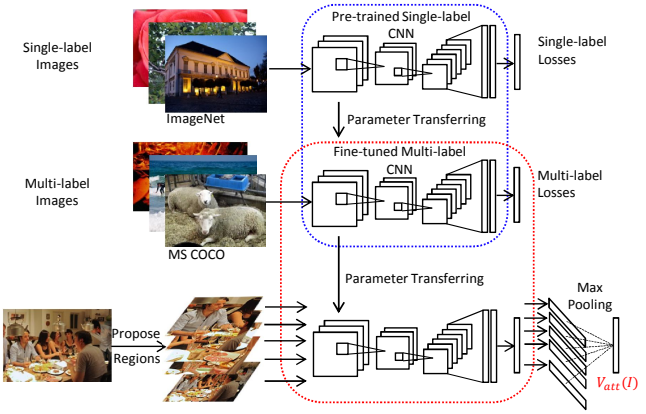


Fig. 3: Attribute prediction CNN: the model is initialized from VggNet [13] pre-trained on ImageNet. The model is then fine-tuned on the target multi-label dataset. Given a test image, a set of proposal regions are selected and passed to the shared CNN, and finally the CNN outputs from different proposals are aggregated with max pooling to produce the final multi-label prediction, which gives us the high-level image representation, $V_{att}(I)$

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.* [42] solved a similar problem using a Multiple Instance Learning framework [68] 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. To address the concern that some attributes may only apply to image sub-regions, we follow Wei *et al.* [69] in designing a region-based multi-label classification framework that takes an arbitrary number of sub-region proposals as input, then a shared CNN is associated with each proposal, and the CNN output results from different proposals are aggregated with max pooling to produce the final prediction.

Figure 3 summarizes the attribute prediction network. In contrast to [69], which uses AlexNet [11] as the initialization of the shared CNN, we use the more powerful VggNet [13] pre-trained on ImageNet [70]. This model has been widely used in image captioning tasks [7], [42], [44], [71]. 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 over the c class labels. The c here represents the attributes vocabulary size. In contrast to [69] 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

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

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 [72]. 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 [69], 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 [69], 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) [73] 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 Caption Generation Model

Similar to [7], [8], [44], we propose to train a caption 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 value $V_{att}(I)$ 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:

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

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 [74], a particular form of RNN, to model this.

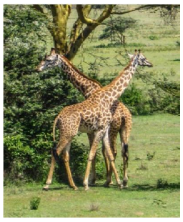
The LSTM is a memory cell encoding knowledge at every time step for what inputs have been observed up to this step. We follow the model used in [75]. Letting σ be the sigmoid nonlinearity, the LSTM updates for time step t given inputs x_t, h_{t-1}, c_{t-1} are:

$$\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) && (3)\\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) && (4)\\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) && (5)\\
g_t &= \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) && (6)\\
c_t &= f_t \odot c_{t-1} + i_t \odot g_t && (7)\\
h_t &= o_t \odot \tanh(c_t) && (8)\\
p_{t+1} &= \text{softmax}(h_t) && (9)
\end{aligned}$$

Here, i_t, f_t, c_t, o_t are the input, forget, memory, output state of the LSTM. The various W matrices are trained parameters and $\odot$ represents the product with a gate value. h_t is the hidden state at time step t and is fed to a Softmax,

Top 5 Attributes:
players, catch, bat, baseball, swing

Generated Captions:
A baseball player swing a bat at a ball.
A baseball player holding a bat on a field.
A baseball player swinging a bat on a field.
A baseball player is swinging a bat at a ball.
A batter catcher and umpire during a baseball game.



Top 5 Attributes:
field, two, tree, grass, giraffe

Generated Captions :
Two giraffes are standing in a grassy field.
A couple of giraffe standing next to each other.
Two giraffes standing next to each other in a field.
A couple of giraffe standing next to each other on a lush green field.

Top 5 Attributes:
pizza, bottle, sitting, table, beer

Generated Captions :
A large pizza sitting on top of a table.
A pizza sitting on top of a white plate.
A pizza sitting on top of a table next to a beer.
A pizza sitting on top of a table next to a bottle of beer.

Fig. 4: Examples of predicted attributes and generated captions.

which will produce a probability distribution p_{t+1} over all words and indicate the word at time step $t+1$.

Training details: 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 2538, 7414, and 8791 words on Flickr8k, Flickr30k and MS COCO datasets separately. Note it is different from the semantic attributes vocabulary $\mathcal{V}_{att}$. The training procedure is as following: 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. 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 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:

$$\mathcal{C} = -\frac{1}{N}\sum_{i=1}^{N}\log p(S^{(i)}|V_{att}(I^{(i)})) + \lambda_{\boldsymbol{\theta}}\cdot||\boldsymbol{\theta}||_2^2 \quad (10)$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{L^{(i)}+1}\log p_t(S_t^{(i)}) + \lambda_{\boldsymbol{\theta}}\cdot||\boldsymbol{\theta}||_2^2 \quad (11)$$

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.

To infer the sentence given an input image, we use the Beam Search, *i.e.*, we iteratively consider the set of b best sentences up to time t as candidates to generate sentences at time $t+1$, and only keep the best b results. We set the b as 5. Figure 4 shows some examples of the predicted attributes and generated captions. More results can be found in the supplementary material.

4 A VQA Model with External Knowledge

The key differentiator of our VQA model is that it is able to usefully combine image information with that extracted from a Knowledge Base, within the LSTM framework. The novelty lies in the fact that this is achieved by representing both of these disparate forms of information as text before combining them. Figure 5 summarises how this is achieved: given an image, an attribute-based representation $V_{att}(I)$ (in Section 3.1) is first generated and it will used as one of input sources of our VQA-LSTM model. The second input source are those captions generated in section 3.2. Rather than inputing the generated words directly, the hidden state vector of the caption-LSTM after it has generated the last word in each caption is used to represent its content. Average-pooling is applied over the 5 hidden-state vectors, to obtain a vector representation $V_{cap}(I)$ for the image I. The third input source is the textual knowledge which is mined from a large-scale knowledge base, the DBpedia. More details are shown in the following section.

4.1 Relating to the Knowledge Base

The external data source that we use here is DBpeida [62] as a source of general background information, although any such KB could equally be applied. DBpeida is a structured database of information extracted from Wikipedia. The whole DBpedia dataset describes 4.58 million entities, of which 4.22 million are classified in a consistent ontology. The data can be accessed using an SQL-like query language for RDF called SPARQL. Given an image and its predicted attributes, we use the top-five[2] most strongly predicted attributes to generate DBpedia queries. There are a range of problems with DBpedia and similar, however, including the sparsity of the information, and the inconsistency of its representation. Inspecting the database shows that the 'comment' field is the most generally informative about an attribute, as it contains a general text description of it. We therefore retrieve the comment text for each query term. The KB+SPARQL combination is very general, however, and could be applied problem specific KBs, or a database of common sense information, and can even perform basic

2. We only use top-5 attributes to query the KB because, based on observation of training data, an image typically contains 5-8 attributes. We also tested with top-10, but no improvements were observed.

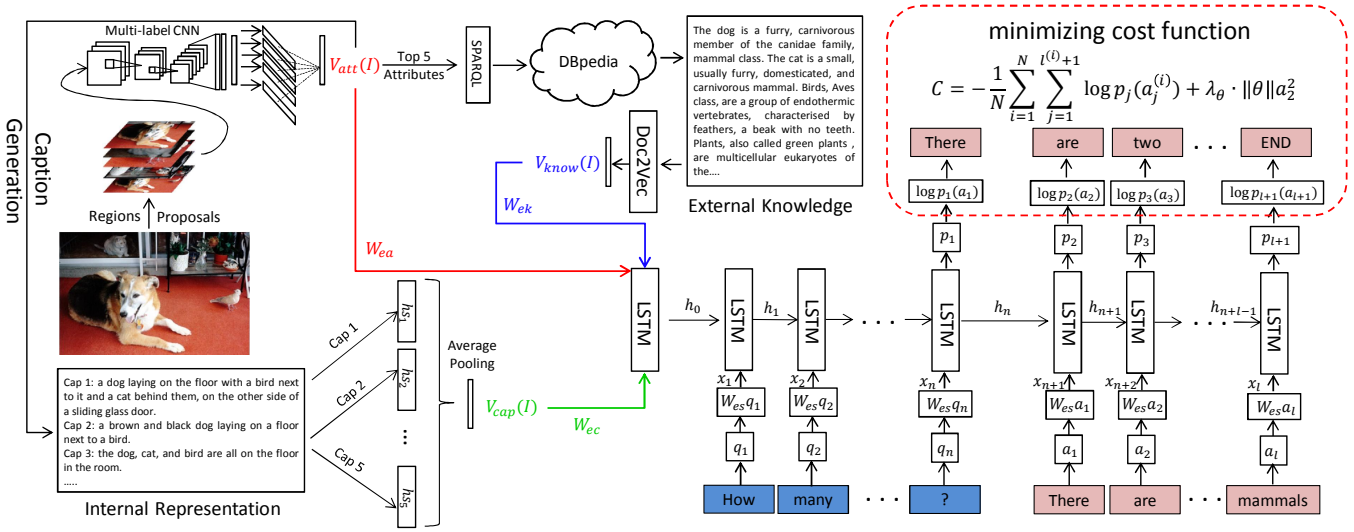


Fig. 5: Our proposed framework: given an image, a CNN is first applied to produce the attribute-based representation $V_{att}(I)$. The internal textual representation is made up of image captions generated based on the image-attributes. The hidden state of the caption-LSTM after it has generated the last word in each caption is used as its vector representation. These vectors are then aggregated as $V_{cap}(I)$ with average-pooling. The external knowledge is mined from the KB (in this case DBpedia) and the responses encoded by Doc2Vec, which produces a vector $V_{know}(I)$. The 3 vectors $\mathbf{V}$ are combined into a single representation of scene content, which is input to the VQA LSTM model which interprets the question and generates an answer.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
sparql SELECT DISTINCT ?comment WHERE {
  ?entry rdfs: label "Dog"@en.
  ?entry rdfs: comment ?comment.
}
```

The domestic dog is a furry, carnivorous member of the canidae family, mammal class. Domestic dogs are commonly known as "man's best friend". The dog was the first domesticated animal and has been widely kept as a working, hunting, and pet companion. It is estimated there are between 700 million and one billion domestic dogs, making them the most abundant member of order Carnivora.

Fig. 6: An example of SPARQL query language for the attribute `'dog'`. The mined text-based knowledge are shown below.

inference over RDF. Figure 6 shows an example of the query language and returned text.

Since the text returned by the SPARQL query is typically much longer than the captions generated in the section 3.2, we turn to Doc2Vec [19] to extract the semantic meanings[3]. Doc2Vec, also known as Paragraph Vector, is an unsupervised algorithm that learns fixed-length feature representations from variable-length pieces of texts, such as sentences, paragraphs, and documents. Le *et al.* [19] proved that it can capture the semantics of paragraphs. A Doc2Vec model is trained to predict words in the document given the context words. We collect 100,000 documents from DBpedia to train a model with vector size 500. To obtain the knowledge vector $V_{know}(I)$ for image I, we combine the 5 returned paragraphs in to a single large paragraph, before semantic features using our pre-trained Doc2Vec model.

4.2 Question-guided Knowledge Selection

We incrementally implemented a question-guided knowledge selection scheme to rule out the noise information, since we observed that some mined knowledge are not necessary for answering the given question. For example, if the question is asking about the 'dog' in the image, it does not make sense to input a piece of 'bird' knowledge into the model, although the image does have a 'bird' inside.

Given a question Q and mined n knowledge paragraphs using above KB+SPARQL combination, we first use our pre-trained Doc2Vec model to extract the semantic feature $V(Q)$ of the question and the feature $V(K_i)$ for each single knowledge paragraph, where $i \in n$. Then, we find the k most closest knowledge paragraph to the question based on the cosine similarity between the $V(Q)$ and $V(K_i)$. Finally, we combine the k selected knowledge paragraph in to a single one and use the Doc2Vec model to extract its semantic feature. In our experiments, we set $n = 10, k = 5$.

4.3 An Answer Generation Model with Multiple Inputs

We propose to train a VQA model by maximizing the probability of the correct answer given the image and question. We want our VQA model to be able to generate multiple word answers, so we formulate the answering process as a word sequence generation procedure. Let $Q = \{q_1,...,q_n\}$ represent the sequence of words in a question, and $A = \{a_1,...,a_l\}$ the answer sequence, where n and l are the length of question and answer, respectively. The log-likelihood of the generated answer can be written as:

$$\log p(A|I,Q) = \sum_{t=1}^{l} \log p(a_t|a_{1:t-1},I,Q) \qquad (12)$$

where $p(a_t|a_{1:t-1},I,Q)$ is the probability of generating a_t given image information I, question Q and previous words $a_{1:t-1}$. We employ an encoder LSTM [74] to take the semantic information from image I and the question Q, while using a decoder LSTM to generate the answer. Weights are shared between the encoder and decoder LSTM.

In the training phase, the question Q and answer A are concatenated as $\{q_1,...,q_n,a_1,...,a_l,a_{l+1}\}$, where a_{l+1} is a special END token. Each word is represented as a one-hot vector of dimension equal to the size of the word dictionary.

3. We investigated to use an LSTM to encode the mined paragraphs, but we observed little performance improvement, despite the additional training overhead.

The training procedure is as follows: at time step $t = 0$, we set the LSTM input:

$$x_{initial} = [W_{ea}V_{att}(I), W_{ec}V_{cap}(I), W_{ek}V_{know}(I)] \quad (13)$$

where W_{ea}, W_{ec}, W_{ek} are learnable embedding weights for the vector representation of attributes, captions and external knowledge, respectively. Given the randomly initialized hidden state, the encoder LSTM feeds forward to produce hidden state h_0 which encodes all of the input information. From $t = 1$ to $t = n$, we set $x_t = W_{es}q_t$ and the hidden state h_{t-1} is given by the previous step, where W_{es} is the learnable word embedding weights. The decoder LSTM runs from time step $n + 1$ to $l + 1$. Specifically, at time step $t = n + 1$, the LSTM layer takes the input $x_{n+1} = W_{es}a_1$ and the hidden state h_n corresponding to the last word of the question, where a_1 is the start word of the answer. The hidden state h_n thus encodes all available information about the image and the question. The probability distribution p_{t+1} over all answer words in the vocabulary is then computed by the LSTM feed-forward process. Finally, for the final step, when a_{l+1} represents the last word of the answer, the target label is set to the END token.

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

$$\mathcal{C} = -\frac{1}{N}\sum_{i=1}^{N} \log p(A^{(i)}|I,Q) + \lambda_{\boldsymbol{\theta}} \cdot ||\boldsymbol{\theta}||_2^2 \quad (14)$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{l^{(i)}+1} \log p_j(a_j^{(i)}) + \lambda_{\boldsymbol{\theta}} \cdot ||\boldsymbol{\theta}||_2^2 \quad (15)$$

where N is the number of training examples, and $n^{(i)}$ and $l^{(i)}$ are the length of question and answer respectively for the i-th training example. Let $p_t(a_t^{(i)})$ correspond to the activation of the Softmax layer in the LSTM model for the i-th input and $\boldsymbol{\theta}$ represent the model parameters. Note that $\lambda_{\boldsymbol{\theta}} \cdot ||\boldsymbol{\theta}||_2^2$ is a regularization term. We use Stochastic gradient Descent (SGD) with mini-batches of 100 image-QA pairs. The attributes, internal textual representation, external knowledge embedding size, word embedding size and hidden state size are all 256 in all experiments. The learning rate is set to 0.001 and clip gradients is 5. The dropout rate is set to 0.5.

5 EXPERIMENTS

We evaluate our image captioning model and visual question answering model separately in the following sections.

5.1 Evaluation on Image Captioning

5.1.1 Dataset

There are several datasets which consist of images and sentences in English describing these images. We report results on the popular Flickr8k [29], Flickr30k [76] and Microsoft COCO dataset [77]. These datasets contain 8,000, 31,000 and 123,287 images respectively, and each image is annotated with 5 sentences. In our reported results, we use pre-defined splits for Flickr8k, 1000 for validation, 1000 for testing and the rest for training. Because most of previous works in image captioning [5], [7], [8], [42], [44], [45] are not evaluated on the official split for Flickr30k and MS COCO, for fair comparison, we report results with the widely used publicly available splits in the work of [44], which use 1000 images for validation, 1000 for testing for Flickr30k, and 5000 images for both validation and testing in MS COCO. We further tested on the actually MS COCO test set (official split) consisting of 40775 images (human captions for this split are not available publicly), and evaluated them on the COCO evaluation server.

Flickr8k

State-of-art-Flickr8k	B-1	B-2	B-3	B-4	$\mathcal{PPL}$
Karpathy & Li (NeuralTalk) [44]	0.58	0.38	0.25	0.16	-
Chen & Zintick (Mind's Eye) [4]	-	-	-	0.14	15.10
Google(NIC) [8]	0.66	0.42	0.27	0.18	-
Mao et al. (m-Rnn-AlexNet) [7]	0.57	0.39	0.26	0.17	24.39
Xu et al. (Hard-Attention) [45]	0.67	0.46	0.31	0.21	-
Baseline - *CNN(I)*					
VggNet+LSTM	0.56	0.37	0.24	0.16	15.71
VggNet-PCA+LSTM	0.56	0.38	0.25	0.16	16.07
GoogLeNet+LSTM	0.56	0.38	0.24	0.16	15.71
VggNet+ft+LSTM	0.64	0.43	0.30	0.20	14.69
Ours - $V_{att}(I)$					
Attributes-GT+LSTM‡	0.76	0.57	0.41	0.29	12.52
Attributes-SVM+LSTM	0.73	0.53	**0.38**	0.26	12.63
Attributes-CNN+LSTM	**0.74**	**0.54**	**0.38**	**0.27**	**12.60**

Flickr30k

State-of-art-Flickr30k	B-1	B-2	B-3	B-4	$\mathcal{PPL}$
Karpathy & Li (NeuralTalk) [44]	0.57	0.37	0.24	0.16	-
Chen & Zintick (Mind's Eye) [4]	-	-	-	0.13	19.10
Google(NIC) [8]	0.66	-	-	-	-
Donahue et al. (LRCN) [5]	0.59	0.39	0.25	0.17	-
Mao et al. (m-Rnn-AlexNet) [7]	0.54	0.36	0.23	0.15	35.11
Mao et al. (m-Rnn-VggNet) [7]	0.60	0.41	0.28	0.19	20.72
Xu et al. (Hard-Attention) [45]	0.67	0.44	0.30	0.20	-
Baseline - *CNN(I)*					
VggNet+LSTM	0.57	0.38	0.25	0.17	18.83
VggNet-PCA+LSTM	0.59	0.40	0.26	0.17	18.92
GoogLeNet+LSTM	0.58	0.39	0.26	0.17	18.77
VggNet+ft+LSTM	0.67	0.47	0.31	0.21	16.62
Ours - $V_{att}(I)$					
Attributes-GT+LSTM‡	0.78	0.57	0.42	0.30	14.88
Attributes-SVM+LSTM	0.68	0.49	0.33	0.23	16.01
Attributes-CNN+LSTM	**0.73**	**0.55**	**0.40**	**0.28**	**15.96**

TABLE 1: BLEU-1,2,3,4 and $\mathcal{PPL}$ metrics compared to other state-of-the-art methods and our baseline on Flickr8k and Flickr30K dataset. ‡ indicates ground truth attributes labels are used, which (in gray) will not participate in rankings. Our $\mathcal{PPL}s$ are based on Flickr8k and Flickr30k word dictionaries of size 2538 and 7414, respectively.

5.1.2 Evaluation

Metrics: We report results with the frequently used BLEU metric and sentence perplexity ($\mathcal{PPL}$). BLEU [78] 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 ($\mathcal{PPL}$) 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. For MS COCO dataset, we additionally evaluate our model based on the metrics of METEOR [79] and CIDEr [80]. All scores (except $\mathcal{PPL}$) are computed with the `coco-caption` code [81].

Baselines: To verify the effectiveness of our high-level attributes representation, we provide a baseline method.

The baseline framework is same as the one proposed in section 3.2, except that the attributes vector $V_{att}(I)$ is replaced by the last hidden layer of CNN directly. Various CNN architectures are applied in the baseline method to extract image features, such as VggNet [13] and GoogLeNet [14]. For the **VggNet+LSTM**, we use the second fully connected layer (`fc7`) as the image features, which has 4096 dimensions. In **VggNet-PCA+LSTM**, PCA is applied to decrease the feature dimension from 4096 to 1000. For the **GoogLeNet+LSTM**, we use the model provided in the Caffe Model Zoo [82] and the last average pooling layer is employed, which is a 1024-d vector. **VggNet+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 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.

Results: Table 1 and 2 report image captioning results on Flickr8k, Flickr30k and Microsoft COCO dataset. It is not surprising that **Att-GT+LSTM** model performs best, since ground truth attributes labels are used. We report these results here 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 on all the three datasets 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. **VggNet+ft+LSTM** models perform 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. **Attributes-SVM+LSTM** under-perform **Attributes-CNN+LSTM** means our region-based attributes prediction network performs better than the whole image classification. Our final model also outperforms current state-of-the-arts listed in tables. We also evaluate an approach (not shown in table) that combines CNN features and attributes vector together as the input of the LSTM, but we find this approach is not as good as using attributes vector only in the same setting. In any case, above experiments show that an intermediate image-to-words stage (i.e. attributes prediction layer) bring us significant improvements.

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 3. We achieve 0.73 on B-1, and surpass human performances on 13 of the 14 metrics reported. Other state-of-the-art methods are also shown for comparison.

State-of-art	B-1	B-2	B-3	B-4	M	C	$\mathcal{P}$
NeuralTalk [44]	0.63	0.45	0.32	0.23	0.20	0.66	-
Mind's Eye [4]	-	-	-	0.19	0.20	-	11.60
NIC [8]	-	-	-	0.28	0.24	0.86	-
LRCN [5]	0.67	0.49	0.35	0.25	-	-	-
Mao et al. [7]	0.67	0.49	0.34	0.24	-	-	13.60
Jia et al. [46]	0.67	0.49	0.36	0.26	0.23	0.81	-
MSR [42]	-	-	-	0.26	0.24	-	18.10
Xu et al. [45]	0.72	0.50	0.36	0.25	0.23	-	-
Jin et al. [83]	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 2: BLEU-1,2,3,4, METEOR, CIDEr and $\mathcal{PPL}$ metrics compared to 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. Our $\mathcal{PPL}s$ are based on MS COCO word dictionaries of size 8791.

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 [42]	0.70	0.53	0.39	0.29	**0.25**	0.52	0.91
m-RNN [7]	0.68	0.51	0.37	0.27	0.23	0.50	0.79
LRCN [5]	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 [42]	0.88	0.79	0.68	0.57	0.33	0.66	**0.93**
m-RNN [7]	0.87	0.76	0.64	0.53	0.30	0.64	0.79
LRCN [5]	0.87	0.77	0.65	0.53	0.32	0.66	0.89

TABLE 3: 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.

Table 4 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.

	Ours	NIC [8]	LRCN [5]	m-RNN [7]	NeuralTalk [44]
VIS Input Dim	256	1000	1000	4096	4096
RNN Dim	256	512	1000×4	256	300-600

TABLE 4: 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.

5.2 Evaluation on Visual Question Answering

We evaluate our model on four recent publicly available visual question answering datasets, two toy size and two large size. DAQURA-ALL is proposed in [84]. There are 7,795 training questions and 5,673 test questions. These questions are generated on 795 and 654 images respectively. The questions are categorized into three types including *Object*, *Color* and *Number*. Most of the answers are single word. DAQURA-REDUCED is a reduced version of DAQURA-ALL. There are 3,876 training questions and only 297 test questions. This dataset is constrained to 37 object categories and uses only 25 test images. Two large-scale VQA data are

	DAQURA All	DAQURA Reduced	Toronto COCO-QA	VQA
# Images	1,449	1,423	117,684	204,721
# Questions	12,468	4,173	117,684	614,163
# Question Types	3	3	4	more than 20
# Ans per Que	1	1	1	10
# Words per Ans	1+	1+	1	1+

TABLE 5: Some statistics about the DAQURA, Toronto COCO-QA Dataset [18] and MS COCO-VQA dataset [15].

constructed both based on MS COCO images. The Toronto COCO-QA Dataset [18] contains 78,736 training and 38,948 testing examples, which are generated from 117,684 images. There are four types of questions, relating to the object, number, color and location, all constructed so as to have a single-word answer. All of the question-answer pairs in this dataset are automatically converted from human-sourced image descriptions. Another benchmarked dataset is VQA [15], which is a much larger dataset and contains 614,163 questions and 6,141,630 answers based on 204,721 MS COCO images. This dataset provides a surprising variety of question types, including "What is...', "How Many" and even "Why...". The ground truth answers were generated by 10 human subjects and can be single word or sentences. The data train/val split follows the COCO official split, which contains 82,783 training images and 40,504 validation images, each has 3 questions and 10 answers. We randomly choose 5000 images from the validation set as our val set, with the remainder testing. The human ground truth answers for the actual VQA test split are not available publicly and only can be evaluated via the VQA evaluation server. Hence, we also apply our final model on a test split and report the overall accuracy. Table 5 displays some dataset statistics.

5.2.1 Results on DAQURA

Metrics: Following [18], [52], the accuracy value (the proportion of correctly answered test questions), and the Wu-Palmer similarity (WUPS) [85] are used to measure performance. 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 considered to be right. We report on thresholds 0.9 and 0.0, following [18], [52].

Evaluations: To illustrate the effectiveness of our model, we provide two baseline models and several state-of-the-art results in table 6 and 7. The **Baseline** method is implemented simply by connecting a CNN to an LSTM. The CNN is a pre-trained (on ImageNet) VggNet model from which we extract the coefficients of the last fully connected layer. We also implement a baseline model **VggNet+ft-LSTM**, which applies a vggNet that has been fine-tuned on the COCO dataset, based on the task of image-attributes classification. We also present results from a series of cut down versions of our approach for comparison. **Att-LSTM** uses only the semantic level attribute representation V_{att} as the LSTM input. To evaluate the contribution of the internal textual representation and external knowledge for the question answering, we feed the image caption representation V_{cap} and knowledge representation V_{know} with the V_{att} separately, producing two models, **Att+Cap-LSTM** and **Att+Know-LSTM**. We also tested the **Cap+Know-LSTM**, for the experiment completeness. **Att+Cap+Know-LSTM** combines all the available information. Our final model is the **A+C+Selected-K-LSTM**, which uses the selected knowledge information (see section 4.2) as the input. **GUESS** [18] simply selects the modal answer from the training set for each of 4 question types. **VIS+BOW** [18] performs multinomial logistic regression based on image features and a BOW vector obtained by summing all the word vectors of the question. **VIS+LSTM** [18] has one LSTM to encode the image and question, while **2-VIS+BLSTM** [18] has two image feature inputs, at the start and the end. Malinowski*et al.* [17] propose a neural-based approach and Ma *et al.* [52] encodes both images and questions with a CNN. Yang *et al.* [58] use a stacked attention networks to infer the answer progressively.

DAQURA-All	Acc(%)	WUPS@0.9	WUPS@0.0
Askneuron [17]	19.43	25.28	62.00
Ma *et al.* [52]	23.40	29.59	62.95
Yang *et al.* [58]	**29.30**	35.10	68.60
Noh *et al.* [59]	28.98	34.80	67.81
Baseline			
VggNet-LSTM	23.13	30.01	63.61
VggNet+ft-LSTM	23.75	30.22	63.66
Our-Proposal			
Att-LSTM	24.27	30.41	62.29
Att+Cap-LSTM	27.04	33.40	67.65
Att+Know-LSTM	24.89	31.27	66.11
Cap+Know-LSTM	23.91	30.64	65.01
Att+Cap+Know-LSTM	29.16	35.30	68.66
A+C+Selected-K-LSTM	29.23	**35.37**	**68.72**

TABLE 6: Accuracy, WUPS metrics compared to other state-of-the-art methods and our baseline on DAQURA-All.

DAQURA-Reduced	Acc(%)	WUPS@0.9	WUPS@0.0
GUESS [18]	18.24	29.65	77.59
VIS+BOW [18]	34.17	44.99	81.48
VIS+LSTM [18]	34.41	46.05	82.23
2-VIS+BLSTM [18]	35.78	46.83	82.15
Askneuron [17]	34.68	40.76	79.54
Ma *et al.* [52]	39.66	44.86	83.06
Xu *et al.* [54]	40.07	-	-
Yang *et al.* [58]	45.50	50.20	83.60
Noh *et al.* [59]	44.48	49.56	**83.95**
Baseline			
VggNet-LSTM	38.72	43.97	83.01
VggNet+ft-LSTM	39.13	44.03	83.33
Our-Proposal			
Att-LSTM	40.07	45.43	82.67
Att+Cap-LSTM	44.78	50.07	83.85
Att+Know-LSTM	41.08	46.04	82.39
Cap+Know-LSTM	40.81	45.04	82.01
Att+Cap+Know-LSTM	45.79	51.53	83.91
A+C+Selected-K-LSTM	**46.13**	**51.83**	**83.95**

TABLE 7: Accuracy, WUPS metrics compared to other state-of-the-art methods and our baseline on DAQURA-Reduced.

All of our proposed models outperform the **Baseline** method. And our final model **A+C+Selected-K-LSTM** achieves the best state-of-the-art on the DAQURA-Reduced set. **Att+Cap+Know-LSTM** performs not as good as **A+C+Selected-K-LSTM**, which shows the effectiveness of our question-based knowledge selection scheme.

5.2.2 Results on Toronto COCO-QA

Evaluations: Table 8 reports the results on Toronto COCO-QA. All of our proposed models outperform the **Baseline** and all of the comparator state-of-the-art methods. Our final

Toronto COCO-QA	Acc(%)	WUPS@0.9	WUPS@0.0
GUESS [18]	6.65	17.42	73.44
VIS+BOW [18]	55.92	66.78	88.99
VIS+LSTM [18]	53.31	63.91	88.25
2-VIS+BLSTM [18]	55.09	65.34	88.64
Ma *et al.* [52]	54.95	65.36	88.58
Chen *et al.* [55]	58.10	68.44	89.85
Yang *et al.* [58]	61.60	71.60	90.90
Noh *et al.* [59]	61.19	70.84	90.61
Baseline			
VggNet-LSTM	50.73	60.37	87.48
VggNet+ft-LSTM	58.34	67.32	89.13
Our-Proposal			
Att-LSTM	61.38	71.15	91.58
Att+Cap-LSTM	69.02	76.20	92.38
Att+Know-LSTM	63.07	72.22	90.84
Cap+Know-LSTM	64.31	73.31	90.01
Att+Cap+Know-LSTM	69.73	77.14	92.50
A+C+Selected-K-LSTM	**70.98**	**78.35**	**92.87**

TABLE 8: Accuracy, WUPS metrics compared to other state-of-the-art methods and our baseline on Toronto COCO-QA dataset.

Toronto COCO-QA	Object	Number	Color	Location
GUESS [18]	2.11	35.84	13.87	8.93
VIS+BOW [18]	58.66	44.10	51.96	49.39
VIS+LSTM [18]	56.53	46.10	45.87	45.52
2-VIS+BLSTM [18]	58.17	44.79	49.53	47.34
Chen *et al.* [55]	62.46	45.70	46.81	53.67
Yang *et al.* [58]	64.50	48.60	57.90	54.00
Baseline				
VggNet-LSTM	53.71	45.37	36.23	46.37
VggNet+ft-LSTM	61.67	50.04	52.16	54.40
Our-Proposal				
Att-LSTM	63.92	51.83	57.29	54.84
Att+Cap-LSTM	71.30	69.98	61.50	60.98
Att+Know-LSTM	64.57	54.37	62.79	56.98
Cap+Know-LSTM	65.61	55.13	62.02	57.28
Att+Cap+Know-LSTM	71.45	**75.33**	**64.09**	60.98
A+C+Selected-K-LSTM	**73.66**	72.20	62.97	**61.18**

TABLE 9: Toronto COCO-QA accuracy (%) per category.

model **A+C+Selected-K-LSTM** achieves the best results. It surpasses the baseline by nearly 20% and outperforms the previous state-of-the-art methods around 10%. **Att+Cap-LSTM** clearly improves the results over the **Att-LSTM** model. This proves that internal textual representation plays a significant role in the VQA task. The **Att+Know-LSTM** model does not perform as well as **Att+Cap-LSTM** , which suggests that the information extracted from captions is more valuable than that extracted from the KB. **Cap+Know-LSTM** also performs better than **Att+Know-LSTM**. This is not surprising because the Toronto COCO-QA questions were generated automatically from the MS COCO captions, and thus the fact that they can be answered by training on the captions is to be expected. This generation process also leads to questions which require little external information to answer. The comparison on the Toronto COCO-QA thus provides an important benchmark against related methods, but does not really test the ability of our method to incorporate extra information. It is thus interesting that the additional external information provides any benefit at all.

Table 9 shows the per-category accuracy for different models. Surprisingly, the counting ability (see question type 'Number') increases when both captions and external knowledge are included. This may be because some 'counting' questions are not framed in terms of the labels used in the MS COCO captions. Ren *et al.*also observed similar cases. In [18] they mentioned that "there was some observable counting ability in very clean images with a single object type but the ability was fairly weak when different object types are present". We also find there is a slight increase for the 'color' questions when the KB is used. Indeed, some questions like 'What is the color of the stop sign?' can be answered directly from the KB, without the visual cue.

5.2.3 Results on VQA

Antol *et al.* [15] provide the VQA dataset which is intended to support "free-form and open-ended Visual Question Answering". They also provide a metric for measuring performance: $\min\{\frac{\#\text{ humans that said answer}}{3}, 1\}$ thus 100% means that at least 3 of the 10 humans who answered the question gave the same answer. We have used the provided evaluation code[4] to produce the results.

Evaluation: There are several splits for VQA dataset, such as the validation set, test-develop and test-standard set. We first tested several aspects of our models on the validation set (we randomly choose 5000 images from the validation set as our val set, with the remainder testing).

Inspecting Table 10, results the on VQA validation set, we see that the attribute-based **Att-LSTM** is a significant improvement over our **VggNet+LSTM** baseline. We also evaluate another baseline, the **VggNet+ft+LSTM**, which uses the penultimate layer of the attributes prediction CNN (in Section 3.1) as the input to the LSTM. Its overall accuracy on the VQA is 50.01, which is still lower than our proposed models (detailed results of different question types are not shown in Table 10 due to the limited space.) Adding either image captions or external knowledge further improves the result. Our final model **A+C+S-K-LSTM** produces the best results, outperforming the baseline **VggNet-LSTM** by 11% overall. Some other stat-of-the-art methods such as [54], [55], [56] produce the overall accuracy 54.69%, 48.38% and 50.48%, respectively, on the validation set (on the different splits). The performance comparison across categories is of particular interest here because answering different classes of questions requires different amounts of external knowledge. The 'Where' questions, for instance, require knowledge of potential locations, and 'Why' questions typically require general knowledge about people's motivation. 'Number' and 'Color' questions, in contrast, can be answered directly. The results show that for 'Why' questions, adding the KB improves performance by more than 50% (Att-LSTM achieves 7.77% while Att+Know-LSTM achieves 11.88%), and that the combined A+C+K-LSTM achieves 13.53%. We further improve it to 13.76% by using the question-guided knowledge selected model A+C+S-K-LSTM.

We have also tested on the VQA test-dev and test-standard[5] consisting of 60,864 and 244,302 questions (for which ground truth answers are not published) using our final A+C+S-K-LSTM model, and evaluated them on the VQA evaluation server[6]. Table 11 and 12 shows the server reported results.

Antol *et al.* [15] provide several results for this dataset. In each case they encode the image with the final hidden layer

4. https://github.com/VT-vision-lab/VQA
5. http://www.visualqa.org/challenge.html
6. https://www.codalab.org/competitions/6961

Question Type	Our-Baseline	Our Proposal				
	VggNet + LSTM	Att + LSTM	Att+Cap + LSTM	Att+Know + LSTM	A+C+K + LSTM	A+C+S-K + LSTM
what is	21.41	34.63	42.21	37.11	**42.52**	42.51
what colour	29.96	39.07	48.65	39.68	48.86	**48.89**
what kind	24.15	41.22	47.93	46.16	**48.05**	48.02
what are	23.05	38.87	47.13	41.13	47.21	**47.27**
what type	26.36	41.71	47.98	44.91	48.11	**48.14**
is the	71.49	73.22	74.63	74.40	**74.70**	**74.70**
is this	73.00	75.26	76.08	**76.56**	76.14	76.17
how many	34.42	39.14	46.61	39.78	**47.38**	**47.38**
are	73.51	75.14	76.01	75.75	76.14	**76.15**
does	76.51	76.71	78.07	76.55	**78.11**	**78.11**
where	10.54	21.42	25.92	24.13	**26.00**	25.96
is there	86.66	87.10	86.82	85.87	87.01	**87.33**
why	3.04	7.77	9.63	11.88	13.53	**13.76**
which	31.28	36.60	**39.55**	37.71	38.70	38.83
do	76.44	75.76	78.18	75.25	78.42	**78.44**
what does	15.45	19.33	21.80	19.50	22.16	**22.71**
what time	13.11	15.34	15.44	**15.47**	15.34	15.17
who	17.07	22.56	25.71	21.23	25.74	**25.97**
what sport	65.65	91.02	93.96	90.86	**94.20**	94.18
what animal	27.77	61.39	70.65	63.91	71.70	**72.33**
what brand	26.73	32.25	33.78	32.44	34.60	**35.68**
others	44.37	50.23	53.29	52.11	53.45	**53.53**
Overall	44.93	51.60	55.04	53.79	55.96	**56.17**

TABLE 10: Results on the open-answer task for various question types on VQA validation set. All results are in terms of the evaluation metric from the VQA evaluation tools. The overall accuracy for the model of **VggNet+ft+LSTM** and **Cap+Know+LSTM** is 50.01 and 52.31 respectively. Detailed results of different question types for these two models are not shown in the table due to the limited space.

VQA Test-dev	Answer Type			Overall
	Yes/No	Other	Number	
Question [15]	75.66	27.14	36.70	40.09
Image [15]	64.01	3.77	0.42	28.13
Q+I [15]	75.55	37.37	33.67	52.64
LSTM Q [15]	78.20	26.59	35.68	48.76
LSTM Q+I [15]	78.94	36.42	35.24	53.74
Jiang *et al.* [56]	78.33	34.46	35.93	52.62
Andreas *et al.* [57]	77.70	39.30	37.20	54.80
Yang *et al.* [58]	79.30	**46.10**	36.60	58.70
Noh *et al.* [59]	80.71	41.69	37.24	57.22
Ours	**81.02**	45.30	**38.47**	**59.22**

TABLE 11: VQA Open-Ended evaluation server results. Accuracies for different answer types and overall performances on the test-dev.

VQA Test-standard	Answer Type			Overall
	Yes/No	Other	Number	
LSTM Q [15]	78.12	26.99	34.94	48.89
LSTM Q+I [15]	79.01	36.80	35.55	54.06
Andreas *et al.* [57]	-	-	-	55.10
Yang *et al.* [58]	-	-	-	58.90
Noh *et al.* [59]	80.28	42.24	36.92	57.36
Ours	**81.10**	**45.90**	**37.18**	**59.50**

TABLE 12: VQA Open-Ended evaluation server results. Accuracies for different answer types and overall performances on the test-standard.

from VggNet, and questions and captions are encoded using a BOW representation. A softmax neural network classifier with 2 hidden layers and 1000 hidden units (dropout 0.5) in each layer with tanh non-linearity is then trained, the output space of which is the 1000 most frequent answers in the training set. They also provide an LSTM model followed by a softmax layer to generate the answer. Two version of this approach are used, one which is given only the question and the image, and one which is given only the question (see [15] for details). Our final model outperforms all the listed approaches according to the overall accuracy. Table 13 provides some indicative results. More results can be found in the supplementary material.

6 Conclusions

In this paper, we first examined the importance of introducing an intermediate attribute prediction layer into the predominant CNN-LSTM framework, which was neglected by almost all previous work. We implemented an attribute-based model which can be applied to the task of image captioning. We have shown that an explicit representation of image content improves *V2L* performance, in all cases. Indeed, at the time of submitting this paper, our image captioning model outperforms the state-of-the-art on several captioning datasets.

Secondly, in this paper we have shown that it is possible to extend the state-of-the-art RNN-based VQA approach so as to incorporate the large volumes of information required to answer general, open-ended, questions about images. The knowledge bases which are currently available do not contain much of the information which would be beneficial to this process, but nonetheless can still be used to significantly improve performance on questions requiring external knowledge (such as 'Why' questions). The approach that we propose is very general, however, and will be applicable to more informative knowledge bases should they become available. We further implement a knowledge selection scheme which reflects both of the content of the question and the image, in order to extract more specifically related information. Currently our system is the state-of-the-art on three VQA datasets and produces the best results on the VQA evaluation server.

Further work includes generating knowledge-base queries which reflect the content of the question and the image, in order to extract more specifically related information. The Knowledge Base itself also can be improved. For instance, Open-IE provides more general common-sense knowledge such as 'cats eat fish'. Such knowledge will help answer high-level questions.

Acknowledgements

This research was in part supported by the Data to Decisions Cooperative Research Centre. Correspondence should be addressed to C. Shen.

References

[1] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[2] 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 on Natural Language Processing*, 2014.

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

[4] 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.

[5] 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.

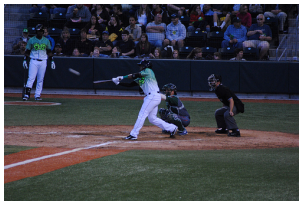
[6] A. Karpathy, A. Joulin, and F. F. Li, "Deep fragment embeddings for bidirectional image sentence mapping," in *Proc. Advances in Neural Inf. Process. Syst.*, 2014.

[7] 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.




	What color is the tablecloth?	How many people in the photo?	What is the red fruit?	What are these people doing?
Ours:	white	2	apple	eating
Vgg+LSTM:	red	1	banana	playing
Ground Truth:	white	2	apple	eating



	Why are his hands outstretched?	Why are the zebras in water?	Is the dog standing or laying down?	Which sport is this?
Ours:	balance	drinking	laying down	baseball
Vgg+LSTM:	play	water	sitting	tennis
Ground Truth:	balance	drinking	laying down	baseball

TABLE 13: Some example cases where our final model gives the correct answer while the base line model **VggNet-LSTM** generates the wrong answer. All results are from MS COCO-VQA. More results can be found in the supplementary material.

[8] 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.

[9] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville, "Describing videos by exploiting temporal structure," *arXiv preprint arXiv:1502.08029*, 2015.

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

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

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

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

[14] 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.

[15] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, "Vqa: Visual question answering - version 2," *arXiv preprint arXiv:1505.00468v2*, 2015.

[16] 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.

[17] 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.

[18] 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.

[19] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," *arXiv preprint arXiv:1405.4053*, 2014.

[20] Q. Wu, C. Shen, A. van den Hengel, L. Liu, and A. Dick, "What value high level concepts in vision to language problems?" in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2016.

[21] Q. Wu, P. Wang, C. Shen, A. van den Hengel, and A. Dick, "Ask me anything: free-form visual question answering based on knowledge from external sources," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2016.

[22] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2009.

[23] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2009.

[24] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in *Proc. IEEE Int. Conf. Comp. Vis.* IEEE, 2009, pp. 365–372.

[25] J. Vogel and B. Schiele, "Semantic modeling of natural scenes for content-based image retrieval," *Int. J. Comput. Vision*, vol. 72, no. 2, pp. 133–157, 2007.

[26] Y. Su and F. Jurie, "Improving image classification using semantic attributes," *IJCV*, vol. 100, no. 1, pp. 59–77, 2012.

[27] L.-J. Li, H. Su, L. Fei-Fei, and E. P. Xing, "Object bank: A high-level image representation for scene classification & semantic feature sparsification," in *Proc. Advances in Neural Inf. Process. Syst.*, 2010, pp. 1378–1386.

[28] L.-J. Li, H. Su, Y. Lim, and L. Fei-Fei, "Objects as attributes for scene classification," in *Trends and Topics in Computer Vision*. Springer, 2012, pp. 57–69.

[29] M. Hodosh, P. Young, and J. Hockenmaier, "Framing image description as a ranking task: Data, models and evaluation metrics," *J. Aritificial Intelligence Research*, pp. 853–899, 2013.

[30] 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.

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

[32] 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.

[33] 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. Annual meeting of the Association for Computational Linguistics*, 2014.

[34] 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.

[35] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi, "Composing simple image descriptions using web-scale n-grams," in *CoNLL: Conference on Natural Language Learning*, 2011.

[36] B. Z. Yao, X. Yang, L. Lin, M. W. Lee, and S.-C. Zhu, "I2t: Image parsing to text description," *Proceedings of the IEEE*, vol. 98, no. 8, pp. 1485–1508, 2010.

[37] 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.*, vol. 35, no. 12, pp. 2891–2903, 2013.

[38] A. Aker and R. Gaizauskas, "Generating image descriptions using dependency relational patterns," in *Proc. Annual meeting of the Association for Computational Linguistics*, 2010.

[39] P. Kuznetsova, V. Ordonez, A. C. Berg, T. L. Berg, and Y. Choi, "Collective generation of natural image descriptions," in *Proc. Annual meeting of the Association for Computational Linguistics*, 2012.

[40] P. Kuznetsova, V. Ordonez, T. L. Berg, and Y. Choi, "Treetalk: Composition and compression of trees for image descriptions," *Proc. Annual meeting of the Association for Computational Linguistics*, 2014.

[41] 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 *Proc. Conf. European Chapter of the Association for Computational Linguistics*, 2012.

[42] 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.

[43] R. Kiros, R. Salakhutdinov, and R. S. Zemel, "Unifying visual-semantic embeddings with multimodal neural language models," in *Trans. Association for Computational Linguistics*, 2015.

[44] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015.

[45] 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 *ICML*, 2015.

[46] 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.

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

[48] 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.

[49] 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.*, 2014, pp. 1682–1690.

[50] K. Tu, M. Meng, M. W. Lee, T. E. Choe, and S.-C. Zhu, "Joint video and text parsing for understanding events and answering queries," *MultiMedia*, vol. 21, no. 2, pp. 42–70, 2014.

[51] D. Geman, S. Geman, N. Hallonquist, and L. Younes, "Visual Turing test for computer vision systems," *Proceedings of the National Academy of Sciences*, vol. 112, no. 12, pp. 3618–3623, 2015.

[52] L. Ma, Z. Lu, and H. Li, "Learning to Answer Questions From Image using Convolutional Neural Network," *arXiv preprint arXiv:1506.00333*, 2015.

[53] Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei, "Visual7W: Grounded Question Answering in Images," *arXiv preprint arXiv:1511.03416*, 2015.

[54] H. Xu and K. Saenko, "Ask, Attend and Answer: Exploring Question-Guided Spatial Attention for Visual Question Answering," *arXiv preprint arXiv:1511.05234*, 2015.

[55] K. Chen, J. Wang, L.-C. Chen, H. Gao, W. Xu, and R. Nevatia, "Abc-cnn: An attention based convolutional neural network for visual question answering," *arXiv preprint arXiv:1511.05960*, 2015.

[56] A. Jiang, F. Wang, F. Porikli, and Y. Li, "Compositional memory for visual question answering," *arXiv preprint arXiv:1511.05676*, 2015.

[57] J. Andreas, M. Rohrbach, T. Darrell, and D. Klein, "Deep compositional question answering with neural module networks," *arXiv preprint arXiv:1511.02799*, 2015.

[58] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, "Stacked Attention Networks for Image Question Answering," *arXiv preprint arXiv:1511.02274*, 2015.

[59] H. Noh, P. H. Seo, and B. Han, "Image question answering using convolutional neural network with dynamic parameter prediction," *arXiv preprint arXiv:1511.05756*, 2015.

[60] L. Zhu, Z. Xu, Y. Yang, and A. G. Hauptmann, "Uncovering Temporal Context for Video Question and Answering," *arXiv preprint arXiv:1511.04670*, 2015.

[61] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: a collaboratively created graph database for structuring human knowledge," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. ACM, 2008, pp. 1247–1250.

[62] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, *Dbpedia: A nucleus for a web of open data*. Springer, 2007.

[63] J. Berant, A. Chou, R. Frostig, and P. Liang, "Semantic parsing on freebase from question-answer pairs," in *Proc. Conf. Empirical Methods on Natural Language Processing*, 2013, pp. 1533–1544.

[64] D. Ferrucci, E. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A. A. Kalyanpur, A. Lally, J. W. Murdock, E. Nyberg, J. Prager *et al.*, "Building Watson: An overview of the DeepQA project," *AI magazine*, vol. 31, no. 3, pp. 59–79, 2010.

[65] X. Lin and D. Parikh, "Don't just listen, use your imagination: Leveraging visual common sense for non-visual tasks," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, June 2015.

[66] F. Sadeghi, S. K. Kumar Divvala, and A. Farhadi, "VisKE: Visual knowledge extraction and question answering by visual verification of relation phrases," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, June 2015.

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

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

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

[70] 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.

[71] X. Chen and C. L. Zitnick, "Learning a Recurrent Visual Representation for Image Caption Generation," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015.

[72] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. Int. Conf. Artificial Intelligence and Statistics*, 2010, pp. 249–256.

[73] J. Pont-Tuset, P. Arbeláez, J. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping for image segmentation and object proposal generation," in *arXiv preprint arXiv:1503.00848*, March 2015.

[74] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[75] W. Zaremba and I. Sutskever, "Learning to execute," *arXiv:1410.4615*, 2014.

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

[77] 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.

[78] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proc. Annual meeting of the Association for Computational Linguistics*, 2002.

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

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

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

[82] 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*, 2014.

[83] 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 preprint arXiv:1506.06272*, 2015.

[84] M. Malinowski and M. Fritz, "Towards a Visual Turing Challenge," *arXiv preprint arXiv:1410.8027*, 2014.

[85] Z. Wu and M. Palmer, "Verbs semantics and lexical selection," in *Proc. Annual meeting of the Association for Computational Linguistics*, 1994.

Qi Wu is a postdoctoral researcher in Australia Centre for Visual Technologies, University of Adelaide. His research interests include cross-depiction object detection and classification, attributes learning, neural networks and image captioning and so on. He received the Bachelor of mathematical science degree from China Jiliang University, Master's degree in computer science, and the PhD degree in computer vision from University of Bath, United Kingdom in 2012 and 2015, respectively.

Chunhua Shen is a Professor of Computer Science at the University of Adelaide. He was with the computer vision program at NICTA (National ICT Australia) Canberra for about six years before he moved back to Adelaide. He studied at Nanjing University, at Australian National University, and received his PhD degree from the University of Adelaide. In 2012, he was awarded the Australian Research Council Future Fellowship.

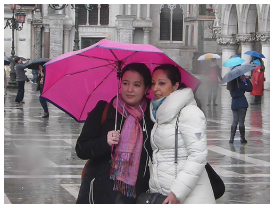
Anton van den Hengel is a Professor and the Founding Director of the Australian Centre for Visual Technologies, at the University of Adelaide, focusing on innovation in the production and analysis of visual digital media. He received the Bachelor of mathematical science degree, Bachelor of laws degree, Master's degree in computer science, and the PhD degree in computer vision from The University of Adelaide in 1991, 1993, 1994, and 2000, respectively.

Peng Wang received the B.S. degree in electrical engineering and automation, and the PhD degree in control science and engineering from Beihang University, China, in 2004 and 2011, respectively. He is now a post-doctoral researcher at the University of Adelaide.

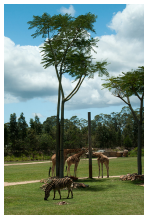
Anthony Dick received the PhD degree in 2002 from the University of Cambridge, UK, where he worked on problems in 3D reconstruction of architecture from images. He is currently an Associate Professor at the University of Adelaide, Australia. His research interests include image based modeling, automated video surveillance, and image search.

7 Additional Results

				
	Why is she wearing a crown?	Why is he smiling?	Why is the zebra on the ground?	Why do they have umbrellas?
Ours:	birthday	happy	resting	shade
Vgg+LSTM:	to eat	unknown	eat	raining
Ground Truth:	birthday	happy	resting	shade

				
	Why is a man sitting under an umbrella?	Why are there animals pinned to the wall?	Why do they have umbrellas?	Why is he swinging backhand?
Ours:	shade	decoration	raining	to hit ball
Vgg+LSTM:	safety	teddy	yes	tennis ball
Ground Truth:	shade	decoration	raining	to hit ball

				
	Why are there so many pillows on the couch?	Why is there water on the ground?	Why are the people wearing wetsuits?	Why are the men wearing helmets?
Ours:	decoration	rain	surfing	safety
Vgg+LSTM:	to rest	drinking	safety	yes
Ground Truth:	decoration	rain	surfing	safety

				
	Why do these sheep have paint on them?	Why is his arm outflung?	Why are the animals laying here?	Why are all the giraffes gathered together?
Ours:	identification	balance	resting	eating
Vgg+LSTM:	to eat	to play	no	to play
Ground Truth:	identification	balance	resting	eating

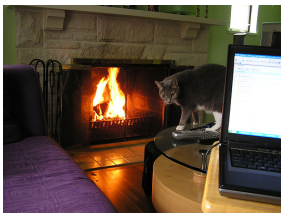
				
	Why are they wearing such bright colors?	Why are the men wearing orange?	Why is the man jumping?	Why is this room warm?
Ours:	safety	team	skateboarding	fireplace
Vgg+LSTM:	yes	to	unknown	to sleep
Ground Truth:	safety	team	skateboarding	fireplace

TABLE 14: Some examples that our final model gives the right answer while the base line model **VggNet-LSTM** generates the wrong answer. All questions are start with 'why' and some of them only can be answered with common sense knowledge.

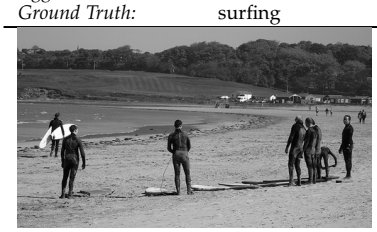
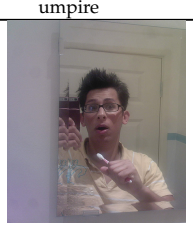
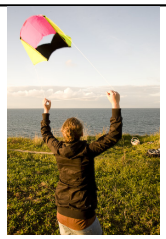
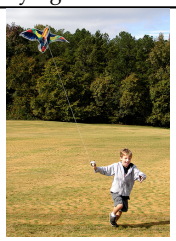
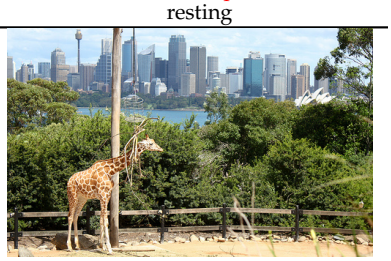
				
Why is this person wet?	Why is the baby wearing a snowsuit?	Why does the boy have his arms in that position?	Why are two of the giraffes so much shorter than the other three?	
Ours:	surfing	cold	balance	they are babies
Vgg+LSTM:	beach	safety	to catch	yes
Ground Truth:	surfing	cold	balance	babies
				
Why is he at the beach in long pants?	Why is this ground white?	Why is there sand around the orange object?	Why is the man wearing black there?	
Ours:	surfing	snow	safety	umpire
Vgg+LSTM:	to water	cold	to balance	safety
Ground Truth:	surfing	snow	safety	umpire
				
Why is she wearing a potholder on her arm?	Why is the road closed?	Why are there no leaves on the trees?	Why does he have glasses on?	
Ours:	cooking	train	winter	to see
Vgg+LSTM:	drinking	stop	unknown	to be
Ground Truth:	cooking	train	winter	to see
				
Why is the ground wet?	Why is he squatting?	Why is the man running?	Why is the cat sitting on the bench?	
Ours:	rain	flying kite	playing frisbee	resting
Vgg+LSTM:	cold	resting	running	to sleep
Ground Truth:	rain	flying kite	playing frisbee	resting
				
Why are her hands in the air?	Why is the man standing?	Why is the child running?	Why is there a giraffe in this setting?	
Ours:	flying kite	playing tennis	flying kite	zoo
Vgg+LSTM:	surfing	tennis ball	playing frisbee	to eat
Ground Truth:	flying kite	playing tennis	flying kite	zoo

TABLE 15: Some examples that our final model gives the right answer while the base line model **VggNet-LSTM** generates the wrong answer. All questions are start with 'why' and some of them only can be answered with common sense knowledge.

Top -5 Attributes
- top, cake, fruits, table, plates

Question Answering
Q1: What are the orange sticks?
A1: carrots (carrots)
Q2: How many carrots are in the bowls?
A2: 2 (over 10)
Q3: Is this set up for a party?
A3: yes (yes)

Generated Caption
- a table topped with plates of food.

Top -5 Attributes
- jumping, watching, skate, board, people

Question Answering
Q1: What is the guy doing?
A1: skateboarding (skateboarding)
Q2: Are both of these skateboarders upside down?
A2: yes (no)
Q3: Why are there signs on the wall?
A3: to keep clean (advertising)

Generated Caption
- a man doing a trick on a skate board.

Top -5 Attributes
- skate, road, board, people, hill

Question Answering
Q1: What is the man standing on?
A1: skateboard (skateboard)
Q2: Is the man facing the camera?
A2: no (no)
Q3: Is this man worshiping the local mountains?
A3: no (no)

Generated Caption
- a young man riding a skateboard down the side of a road.

Top -5 Attributes
- hotdog, people, eating, young, red

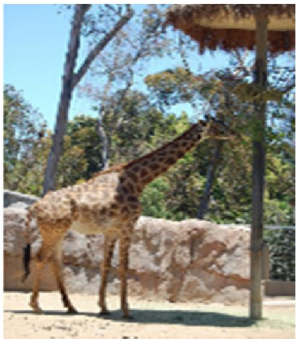
Question Answering
Q1: What color is the woman's jacket?
A1: red (red)
Q2: Is the food good?
A2: yes (yes)
Q3: What condiments did this woman put on the hot dog?
A3: ketchup (ketchup and mustard)

Generated Caption
- a man is holding a hot dog in his hand.

Top -5 Attributes
- shelf, small, book, room, television

Question Answering
Q1: What pattern is on the curtain?
A1: floral (leaves)
Q2: What sport is being displayed on the television?
A2: football (football)
Q3: Is there a bookcase nearby?
A3: yes (yes)

Generated Caption
- a living room with a couch and a television.



Top -5 Attributes
- giraffe, standing, tree, tall, zoo

Question Answering
Q1: Is it warm or cold in this picture?
A1: warm (warm)
Q2: What type of animal is this?
A2: giraffe (giraffe)
Q3: Do you find a stone wall?
A3: yes (yes)

Generated Caption
- a giraffe standing next to a tree in a zoo enclosure.

Top -5 Attributes
- table, vegetables, broccoli, carrots, onions

Question Answering
Q1: Do all the vegetables have roots?
A1: no (no)
Q2: Wouldn't you like to participate in a CSA with veggies like these?
A2: yes (yes)

Generated Captions
- a bunch of green vegetables on a table along with some literature.

Top -5 Attributes
- room, couch, pillows, table, coffee

Question Answering
Q1: What room is this?
A1: living room (living room)
Q2: What shape is the table without any lamps on it?
A2: round (round)

Generated Captions
- a living room with a couch and a coffee table.

Fig. 7: Some qualitative results of our attributes prediction, image captions and question answering. Ground truth answers are in parentheses. Blue indicates we give the right answer, red means we are wrong.

Top -5 Attributes
- zebra, standing, ground, two, zoo

Question Answering
Q1: Where is this picture taken?
A1: zoo (zoo)
Q2: How many zebras?
A2: 2 (2)
Q3: Is the zebra eating cake?
A3: yes (no)

Generated Caption
- two zebras standing next to each other in a zoo enclosure.

Top -5 Attributes
- wine, table, meat, white, vegetables

Question Answering
Q1: What is this drink?
A1: wine (wine)
Q2: How many slices of meat is here?
A2: 2 (6)
Q3: What brand of wine is that?
A3: Daisies (Bock)

Generated Captions
- a plate of food sitting on a table with a glass of wine.

Top -5 Attributes
- baseball, bat, swinging, red, people

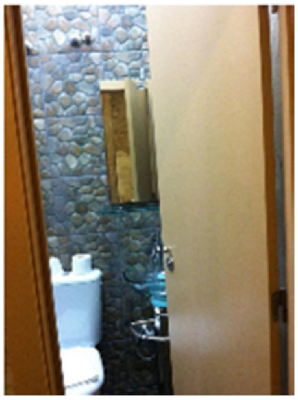
Question Answering
Q1: What brand of cleats is the athlete wearing?
A1: Nike (Nike)
Q2: What type of hat is the better wearing?
A2: baseball (helmet)

Generated Caption
- a baseball player is swinging a bat at a ball.

Top -5 Attributes
- people, holds, cellphone, air, racket

Question Answering
Q1: Is he holding a camera?
A1: yes (yes)
Q2: What is on the man's back?
A2: backpack (backpack)
Q3: Is he bald?
A3: yes (yes)

Generated Caption
- a man is holding a cell phone in his hand.



Top -5 attributes
- bathroom, door, small, wall, sink

Question Answering
Q1: What type of room is this?
A1: bathroom (bathroom)
Q2: Is there medicine in the medicine cabinet?
A2: no (no)
Q3: What is the wall made of?
A3: brick (stone)

Generated Caption
- a small bathroom with a toilet and a sink.

Top -5 Attributes
- bananas, tree, large, green, ground

Question Answering
Q1: Is this a fruit or vegetable?
A1: fruit (fruit)
Q2: Are these bananas ripe?
A2: no (no)
Q3: Does this tree have large leaves?
A3: no (yes)

Generated Caption
- a bunch of bananas hanging from a tree.

Top -5 Attributes
- sheep, field, grass, standing, green

Question Answering
Q1: Which animals are these?
A1: sheep (sheep)
Q2: Will the sheep taste good?
A2: yes (yes)
Q3: What type of ecosystem was this picture taken in?
A3: sheep (farm)

Generated Caption
- a herd of sheep standing on top of a lush green field.

Top -5 Attributes
- bench, park, people, sitting, white

Question Answering
Q1: What color are the slats on the bench?
A1: green (green)
Q2: What's the statue holding?
A2: umbrella (newspaper)
Q3: What color is the statue?
A3: white (white)

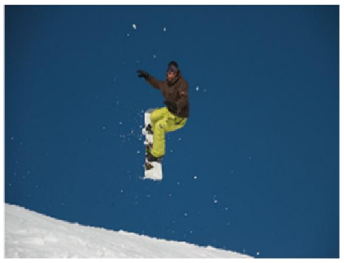
Generated Caption
- a man sitting on a bench in front of a building.

Fig. 8: Some qualitative results of our attributes prediction, image captions and question answering. Ground truth answers are in parentheses. Blue indicates we give the right answer, red means we are wrong.

Generated Caption
- a bathroom with a toilet sink and a mirror.

Top -5 Attributes
- bathroom, sink, wall, two, yellow

Question Answering
Q1: What kind of room is this?
A1: bathroom (bathroom)
Q2: How many sinks are there?
A2: 2 (2)
Q3: Is the door across from the sinks?
A3: yes (yes)



Generated Caption
- a person riding a snow board in the air.

Top -5 Attributes
- snow, people, snowboard, air, riding

Question Answering
Q1: What sport is the man engaging in?
A1: snowboarding (snowboarding)
Q2: Is the man touching the ground?
A2: no (no)
Q3: What is on the man's hands?
A3: gloves (gloves)

Generated Caption
- a brown horse standing on top of a grass covered field.

Top -5 Attributes
- horse, field, brown, grass, standing

Question Answering
Q1: What is the animal eating?
A1: grass (grass)
Q2: Is there a fence?
A2: yes (yes)
Q3: Is there a house in the background?
A3: yes (yes)

Generated Caption
- a busy city street filled with lots of cars.

Top -5 Attributes
- car, traffic, road, tree, people

Question Answering
Q1: Is the street crowded?
A1: yes (yes)
Q2: Can you see the body of ocean in the back?
A2: yes (yes)
Q3: How many red trucks are there?
A3: 2 (8)

Generated Caption
- a police officer riding a motorcycle on a city street.

Top -5 Attributes
- motorcycle, riding, people, office, helmet

Question Answering
Q1: Is this a police officer?
A1: yes (yes)
Q2: Is the police officer happy?
A2:yes (yes)
Q3: What type of vehicle is the policeman driving?
A3: motorcycle (motorcycle)

Generated Caption
- a man swinging a tennis racket at a tennis ball.

Top -5 Attributes
- people, tennis, racket, ball, hitting

Question Answering
Q1: What is the sport the man is playing?
A1: tennis (tennis)
Q2: Did the man hit the ball?
A2: yes (yes)
Q3: What car advertisement is in the background?
A3: Mercedes-benz (Mercedes-benz)

Generated Caption
- a man riding a skateboard down a sidewalk.

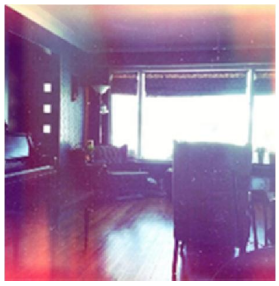
Top -5 Attributes
- skate, board, people, road, riding

Question Answering
Q1: Is the skateboarder casting a shadow?
A1: yes (yes)
Q2: Is the boy airborne?
A2: yes (yes)

Generated Caption
- a group of young men playing a game of frisbee on a beach.

Top -5 Attributes
- boat, young, playing, water, children

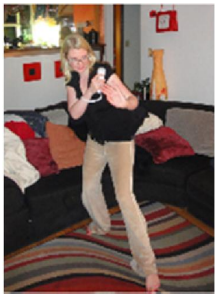
Question Answering
Q1: Is it a chilly day?
A1: yes (yes)
Q2: Are the children fishing?
A2: no (no)
Q3: Is this a recent photo?
A3: no (no)



Generated Caption
- a living room filled with furniture and a large window.

Top -5 Attributes
- room, furniture, large, windows, couch

Question Answering
Q1: Is it a sunny day?
A1: yes (yes)
Q2: Is there a big window?
A2: yes (yes)
Q3: What room is this?
A3: living room (living room)

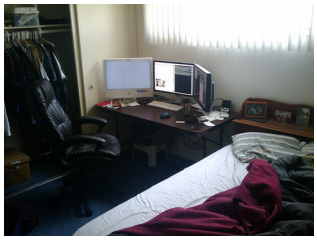


Generated Caption
- a woman is playing a video game in a living room.

Top -5 Attributes
- people, playing, wii, room, glass

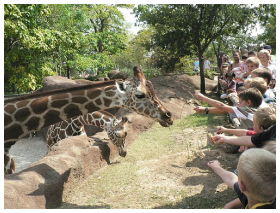
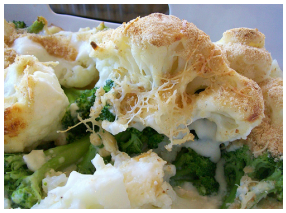
Question Answering
Q1: What gaming system is the woman playing?
A1: Wii (Wii)
Q2: Do you see pillows on the couch?
A2: yes (yes)
Q3: What color is the game controller?
A3: white (white)

Fig. 9: Some qualitative results of our attributes prediction, image captions and question answering. Ground truth answers are in parentheses. Blue indicates we give the right answer, red means we are wrong.

				
	What kind of weather it is?	What is in the cup?	What kind of room is this?	Where did the water come from?
Ours:	sunny	coffee	bedroom	ocean
Vgg+LSTM:	cloudy	wine	living	beach
Ground Truth:	sunny	coffee	bedroom	ocean

				
	What game is being played on the beach?	How many busses are there?	Is the person wearing a shirt?	What is the colorful object in the middle of the image?
Ours:	volleyball	1	no	kite
Vgg+LSTM:	soccer	2	yes	frisbee
Ground Truth:	volleyball	1	no	kite

				
	What are the children holding?	Where is this picture?	What kind of cheese is this?	What room is this?
Ours:	teddy bears	market	mozzarella	bathroom
Vgg+LSTM:	wii	on left	chicken	kitchen
Ground Truth:	teddy bears	market	mozzarella	bathroom

				
	What style of cooking is this?	Is this a men's room or a women's room?	What is on the top of the animals' heads?	Is this a vegetable?
Ours:	chinese	men's	horns	yes
Vgg+LSTM:	pizza	hotel	rocks	no
Ground Truth:	chinese	men's	horns	yes

				
	What are the cats sleeping on?	Is it safe for the pedestrians to cross the street?	What other word is written for this sign?	How many airplanes?
Ours:	car	no	stop	4
Vgg+LSTM:	table	yes	new	1
Ground Truth:	car	no	stop	4

TABLE 16: Some examples that our final model gives the right answer while the base line model **VggNet-LSTM** generates the wrong answer. Various question types are shown.

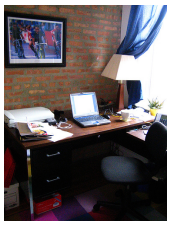
				
	What is he looking at?	What kind of meat is on this?	Which game is being played?	What brand is the bat bag?
Ours:	toothbrush	bacon	soccer	nike
Vgg+LSTM:	camera	chicken	tennis	wilson
Ground Truth:	toothbrush	bacon	soccer	nike
				
	Is this inside?	What season does it look like?	Is this a healthy breakfast?	Is this meal healthy?
Ours:	no	fall	no	yes
Vgg+LSTM:	yes	winter	yes	no
Ground Truth:	no	fall	no	yes
				
	The green item on the pizza, what is it called?	Does this animal have fur?	Is this a home office?	What letter is inside the blue circle?
Ours:	broccoli	no	yes	p
Vgg+LSTM:	carrots	yes	no	b
Ground Truth:	broccoli	no	yes	p
				
	What type of food is this person eating?	In what type of establishment is this taken?	Is that meat on the plate?	What is being celebrated?
Ours:	donut	zoo	yes	birthday
Vgg+LSTM:	pizza	zebra	no	pizza
Ground Truth:	donut	zoo	yes	birthday
				
	What kind of building is this?	Is the weather cold or warm?	Is that normal a banana on a record?	What color is the snow?
Ours:	barn	cold	no	white
Vgg+LSTM:	church	sunny	yes	blue
Ground Truth:	barn	cold	no	white

TABLE 17: Some examples that our final model gives the right answer while the base line model **VggNet-LSTM** generates the wrong answer. Various question types are shown.

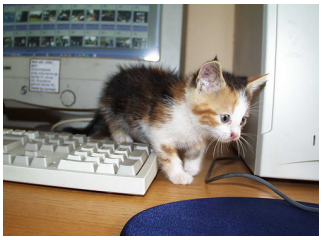
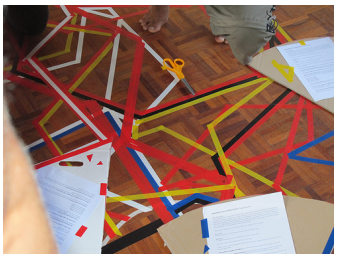
				
What season is it?	What shoe company is advertised?	Is this guy going to jump high?	What is on the keyboard?	
Ours:	fall	vans	no	mouse
Ground Truth:	winter	nike	yes	cat
				
What color is the front of the tow truck?	What kind of wood is the table made of?	Where is the telephone?	How deep is water?	
Ours:	red	oak	on desk	shallow
Ground Truth:	white	cherry	on nightstand	10 feet
				
Who ate some of the cake?	What city is this?	What utensils are shown?	What is the building facade made from?	
Ours:	man	new york	fork and knife	brick
Ground Truth:	person	las vegas	fork	stone
				
What is she holding in her hand?	What creature is this?	What are the colors of the court?	What is on their hand?	
Ours:	ski poles	horse	blue	hot dog
Ground Truth:	ski pole	pegasus	blue and green	glove
				
What's on the floor?	Who took this photo?	What food is this, really?	What kind of weather is this?	
Ours:	scissors	photographer	chicken	rainy
Ground Truth:	tape	christopher brown	cake	cloudy

TABLE 18: Some fail cases produced by our final model