Hierarchical Question-Image Co-Attention for Visual Question Answering

Jiasen Lu, Jianwei Yang, Dhruv Batra, Devi Parikh
Virginia Tech
{jiasenlu, jw2yang, dbatra, parikh}@vt.edu

Abstract

A number of recent works have proposed attention models for Visual Question Answering (VQA) that generate spatial maps highlighting image regions relevant to answering the question. In this paper, we argue that in addition to modeling "where to look" or visual attention, it is equally important to model "what words to listen to" or *question attention*. We present a novel *co-attention* model for VQA that jointly reasons about image and question attention. In addition, our model reasons about the question (and consequently the image via the co-attention mechanism) in a hierarchical fashion via a novel 1-dimensional convolution neural networks (CNN) model. Our final model outperforms all reported methods, improving the state-of-the-art on the VQA dataset from 60.4% to 62.1%, and from 61.6% to 65.4% on the COCO-QA dataset[1].

1 Introduction

Visual Question Answering (VQA) [2, 5, 13, 14] has emerged as a prominent multi-discipline research problem in both academia and industry. To correctly answer visual questions about an image, the machine needs to understand both the image and question. Recently, visual attention based models [16, 21–23] have been explored for VQA, where the attention mechanism typically produces a spatial map highlighting image regions relevant to answering the question.

So far, all attention models for VQA in literature have focused on the problem of identifying "where to look" or visual attention. In this paper, we argue that the problem of identifying "which words to listen to" or *question attention* is equally important. Consider the questions "how many horses are in this image?" and "how many horses can you see in this image?". They have the same meaning, essentially captured by the first three words. A machine that attends to the first three words would arguably be more robust to linguistic variations irrelevant to the meaning and answer of the question. Motivated by this observation, in addition to reasoning about visual attention, we also address the problem of question attention. Specifically, we present a novel multi-modal attention model for VQA with the following two unique features:

Co-Attention: We propose a novel mechanism that jointly reasons about visual attention and question attention, which we refer to as *co-attention*. Unlike previous works, which only focus on visual attention, our model has a natural symmetry between the image and question, in the sense that the image representation is used to guide the question attention and the question representation(s) are used to guide image attention.

Question Hierarchy: We build a hierarchical architecture that co-attends to the image and question at three levels: (a) word level, (b) phrase level and (c) question level. At the word level, we embed the words to a vector space through an embedding matrix. At the phrase level, 1-dimensional convolution neural networks (CNN) are used to capture the information contained in unigrams, bigrams and

---

[1]The source code can be downloaded from https://github.com/jiasenlu/HieCoAttenVQA
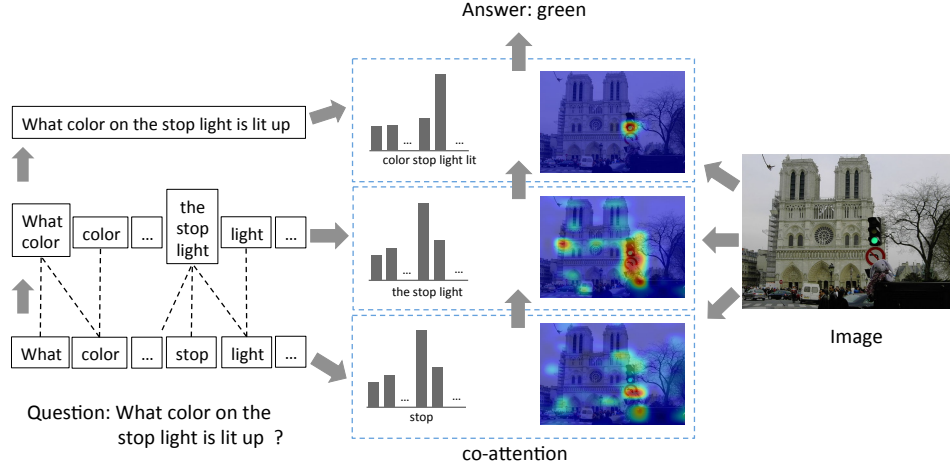
Figure 1: Flowchart of our proposed hierarchical co-attention model. Given a question, we extract its word level, phrase level and question level embeddings. At each level, we apply co-attention on both the image and question. The final answer prediction is based on all the co-attended image and question features.

trigrams. Specifically, we convolve word representations with temporal filters of varying support, and then combine the various n-gram responses by pooling them into a single phrase level representation. At the question level, we use recurrent neural networks (RNN) to encode the entire question. For each level of the question representation in this hierarchy, we construct joint question and image co-attention maps, which are then combined recursively to ultimately predict a distribution over the answers.

Overall, the main contributions of our work are:

- We propose a novel co-attention mechanism for VQA that jointly performs question-guided visual attention and image-guided question attention. We explore this mechanism with two strategies, parallel and alternating co-attention, which are described in Sec. 3.3.1 and Sec. 3.3.2;

- We propose a hierarchical architecture to represent the question, and consequently construct image-question co-attention maps at 3 different levels: word level, phrase level and question level. These co-attended features are then recursively combined from word level to question level for the final answer prediction;

- At the phrase level, we propose a novel convolution-pooling strategy to adaptively select the phrase sizes whose representations are passed to the question level representation;

- Finally, we evaluate our proposed model on two large datasets, VQA [2] and COCO-QA [14]. Our model sets a new state of art on both VQA and COCO-QA outperforming the previous best method on this challenging problem by 2% and 4% respectively. We also perform ablation studies to quantify the roles of different components in our proposed model.

2   Related Work

A number of recent works [2, 5, 10, 13, 14] have proposed models for VQA. We compare and relate our proposed co-attention mechanism to other vision and language attention mechanisms in literature.

Image attention. Instead of directly using the holistic entire-image embedding from the fully connected layer of a deep CNN (as in [2, 12–14]), a number of recent works have explored image attention models for VQA. Zhu *et al*. [25] add spatial attention to the standard LSTM model for pointing and grounded QA. Andreas *et al*. [1] propose a compositional scheme that consists of a language parser and a number of neural modules networks. The language parser predicts which neural module network should be instantiated to answer the question. One of these neural modules networks involves attending to specific regions in an image. Some other works perform image

attention multiple times in a stacked manner. In [23], the authors propose a stacked attention network, which runs multiple iterations or hops to infer the answer progressively. To capture fine-grained information from the question, Xu *et al.* [22] propose a multi-hop image attention scheme. It aligns words to image patches in the first hop, and then refers to the entire question for obtaining image attention maps in the second hop. In [16], the authors generate image regions with object proposals and then select the regions relevant to the question and answer choice. Xiong *et al.* [21] augments dynamic memory network by introducing a new input fusion module. This module captures the spatial information from the neighboring image patches, and retrieves an answer from an attention GRU. Note that all of these approaches model visual attention alone, and do not model question attention.

Language Attention. Though no prior work has explored question attention in VQA, there are some related works in natural language processing (NLP) in general that have modeled language attention. In order to overcome difficulty in translation of long sentences, Hermann *et al.* [3] propose RNNSearch to learn an alignment over the input sentences. In [7], the authors propose an attention model to circumvent the bottleneck caused by fixed width hidden vector in text reading and comprehension. The model first encodes the document and the query using separate bidirectional single layer LSTMs, and then use the outputs as cues for attention. A more fine-grained attention mechanism is proposed in [15]. The authors employ a word-by-word neural attention mechanism to reason about the entailment in two sentences. Also focused on modeling sentence pairs, the authors in [24] propose an attention-based bigram CNN for jointly performing attention between two CNN hierarchies. In their work, three attention schemes are proposed and evaluated.

3 Method

We begin by introducing the notation used in this paper. To ease understanding, our full model is described in parts. First, our hierarchical question representation is described in Sec. 3.2 and the proposed co-attention mechanism is then described in Sec. 3.3. Finally, Sec. 3.4 shows how to recursively combine the attended question and image features to output answers.

3.1 Notation

Given a question with T words, its representation is denoted by $\boldsymbol{Q} = \{\boldsymbol{q}_1, \ldots \boldsymbol{q}_T\}$, where $\boldsymbol{q}_t$ is the feature vector for the t-th word. We denote $\boldsymbol{q}_t^w$, $\boldsymbol{q}_t^p$ and $\boldsymbol{q}_t^s$ as word embedding, phrase embedding and question embedding at position t, respectively. The image feature is denoted by $\boldsymbol{V} = \{\boldsymbol{v}_1, ..., \boldsymbol{v}_N\}$, where $\boldsymbol{v}_n$ is the feature vector at the spatial location n. The co-attention features of image and question at each level in the hierarchy are denoted as $\hat{\boldsymbol{v}}^r$ and $\hat{\boldsymbol{q}}^r$ where $r \in \{w, p, s\}$. The weights in different modules/layers are denoted with $\boldsymbol{W}$, with appropriate sub/super-scripts as necessary. In the exposition that follows, we omit the bias term $\boldsymbol{b}$ to avoid notational clutter, but they are included in the model.
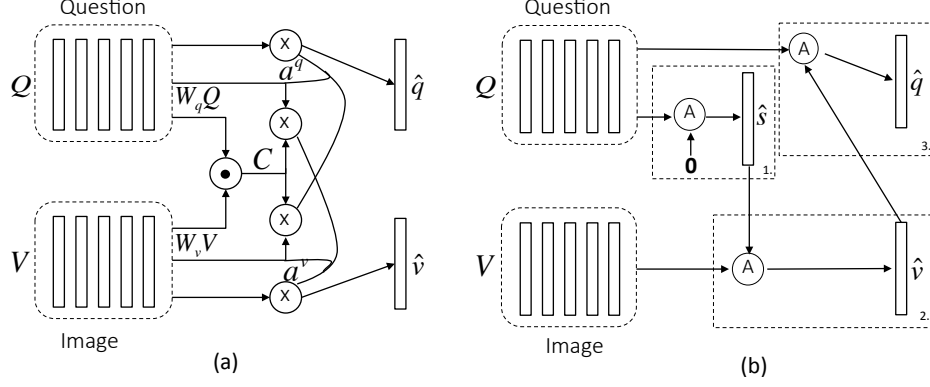
3.2 Question Hierarchy

Given the 1-hot encoding of the question words $\boldsymbol{Q} = \{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_T\}$, we first embed the words to a vector space (learnt end-to-end) to get $\boldsymbol{Q}^w = \{\boldsymbol{q}_1^w, \ldots, \boldsymbol{q}_T^w\}$. To compute the phrase features, we apply 1-D convolution on the word embedding vectors. Concretely, at each word location, we compute the inner product of the word vectors with filters of three window sizes: unigram, bigram and trigram. For the t-th word, the convolution output with window size s is given by

$$\hat{\boldsymbol{q}}_{s,t}^p = \tanh(\boldsymbol{W}_c^s \boldsymbol{q}_{t:t+s-1}^w), \quad s \in \{1, 2, 3\} \tag{1}$$

where $\boldsymbol{W}_c^s$ is the weight parameters. The word-level features $\boldsymbol{Q}^w$ are appropriately 0-padded before feeding into bigram and trigram convolutions to maintain the length of the sequence after convolution. Given the convolution result, we then apply max-pooling across different n-grams at each word location to obtain phrase-level features

$$\boldsymbol{q}_t^p = \max(\hat{\boldsymbol{q}}_{1,t}^p, \hat{\boldsymbol{q}}_{2,t}^p, \hat{\boldsymbol{q}}_{3,t}^p), \quad t \in \{1, 2, \ldots, T\} \tag{2}$$

Our pooling method differs from those used in previous works [8] in that it adaptively selects different gram features at each time step, while preserving the original sequence length and order. We use a

Figure 2: (a) Parallel co-attention mechanism (Sec. 3.3.1); (b) Alternating co-attention mechanism (Sec. 3.3.2).

LSTM to encode the sequence q_t^p after max-pooling. The corresponding question-level feature q_t^s is the LSTM hidden vector at time t.

Our hierarchical representation of the question is depicted in Fig. 3(a).

3.3 Co-Attention

We propose two co-attention mechanisms that differ in the order in which image and question attention maps are generated. The first mechanism, which we call parallel co-attention, generates image and question attention simultaneously. The second mechanism, which we call alternating co-attention, sequentially alternates between generating image and question attentions. See Fig. 2. These co-attention mechanisms are executed at all three levels of the question hierarchy.

3.3.1 Parallel Co-Attention

Parallel co-attention attends to the image and question simultaneously. Similar to [22], we connect the image and question by calculating the similarity between image and question features at all pairs of image-locations and question-locations. Specifically, given an image feature map $V \in \mathcal{R}^{d \times N}$, and the question representation $Q \in \mathcal{R}^{d \times T}$, the affinity matrix $C \in \mathcal{R}^{T \times N}$ is calculated by

$$C = \tanh(Q^T W_b V) \tag{3}$$

where $W_b \in \mathcal{R}^{d \times d}$ contains the weights. After computing this affinity matrix, one possible way of computing the image (or question) attention is to simply maximize out the affinity over the locations of other modality, *i.e.* $a^v[n] = \max_i(C_{i,n})$ and $a^q[t] = \max_j(C_{t,j})$. Instead of choosing the max activation, we find that performance is improved if we consider this affinity matrix as a feature and learn to predict image and question attention maps via the following

$$H^v = \tanh(W_v V + C(W_q Q)), \quad H^q = \tanh(W_q Q + C^T(W_v V))$$
$$a^v = \text{softmax}(w_{hv}^T H^v), \quad a^q = \text{softmax}(w_{hq}^T H^q) \tag{4}$$

where $W_v, W_q \in \mathcal{R}^{k \times d}$, $w_{hv}, w_{hq} \in \mathcal{R}^k$ are the weight parameters. $a^v \in \mathcal{R}^N$ and $a^q \in \mathcal{R}^T$ are the attention probabilities of each image region v_n and word q_t respectively. Based on the above attention weights, the image and question attention vectors are calculated as the weighted sum of the image features and question features, i.e.,

$$\hat{v} = \sum_{n=1}^N a_n^v v_n, \quad \hat{q} = \sum_{t=1}^T a_t^q q_t \tag{5}$$

The parallel co-attention is done at each level in the hierarchy, leading to $\hat{v}^r$ and $\hat{q}^r$ where $r \in \{w, p, s\}$.

3.3.2 Alternating Co-Attention

In this attention mechanism, we sequentially alternate between generating image and question attention. Briefly, this consists of three steps (marked in Fig. 2b): 1) summarize the question into a
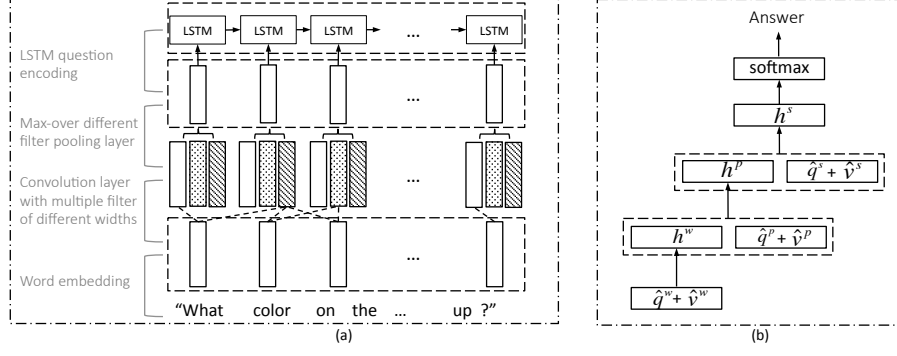
Figure 3: (a) Hierarchical question encoding (Sec. 3.2); (b) Encoding for predicting answers (Sec. 3.4).

single vector $\boldsymbol{q}$; 2) attend to the image based on the question summary $\boldsymbol{q}$; 3) attend to the question based on the attended image feature.

Concretely, we define an attention operation $\hat{\boldsymbol{x}} = \mathcal{A}(\boldsymbol{X}; \boldsymbol{g})$, which takes the image (or question) features $\boldsymbol{X}$ and attention guidance $\boldsymbol{g}$ derived from question (or image) as inputs, and outputs the attended image (or question) vector. The operation can be expressed in the following steps

$$
\begin{aligned}
\boldsymbol{H} &= \tanh(\boldsymbol{W}_x \boldsymbol{X} + (\boldsymbol{W}_g \boldsymbol{g}) \mathbb{1}^T) \\
\boldsymbol{a}^x &= \mathrm{softmax}(\boldsymbol{w}_{hx}^T \boldsymbol{H}) \\
\hat{\boldsymbol{x}} &= \sum_i a_i^x \boldsymbol{x}_i
\end{aligned}
\tag{6}
$$

where $\mathbb{1} \in \mathcal{R}^k$ is a vector with all elements 1. $\boldsymbol{W}_x, \boldsymbol{W}_g \in \mathcal{R}^{k \times d}$ and $\boldsymbol{w}_{hx} \in \mathcal{R}^k$ are parameters. $\boldsymbol{a}^x$ is the attention weight of feature $\boldsymbol{X}$.

The alternating co-attention process is illustrated in Fig. 2 (b). At the first step of alternating co-attention, $\boldsymbol{X} = \boldsymbol{Q}$, and $\boldsymbol{g}$ is $\boldsymbol{0}$; At the second step, $\boldsymbol{X} = \boldsymbol{V}$ where $\boldsymbol{V}$ is the image features, and the guidance $\boldsymbol{g}$ is intermediate attended question feature $\hat{s}$ from the first step; Finally, we use the attended image feature $\hat{v}$ as the guidance to attend the question again, i.e., $\boldsymbol{X} = \boldsymbol{Q}$ and $\boldsymbol{g} = \hat{v}$. Similar to the parallel co-attention, the alternating co-attention is also done at each level of the hierarchy.

3.4 Encoding for Predicting Answers

Following [2], we take the top-1000 frequent answers and treat VQA as 1000-way classification problem. We predict the answer based on the co-attended image and question features from all three levels. We use a multi-layer perceptron (MLP) to recursively encode the attention features as shown in Fig. 3(b).

$$
\begin{aligned}
\boldsymbol{h}^w &= \tanh(\boldsymbol{W}_w(\hat{\boldsymbol{q}}^{\boldsymbol{w}} + \hat{\boldsymbol{v}}^{\boldsymbol{w}})) \\
\boldsymbol{h}^p &= \tanh(\boldsymbol{W}_p[(\hat{\boldsymbol{q}}^{\boldsymbol{p}} + \hat{\boldsymbol{v}}^{\boldsymbol{p}}), \boldsymbol{h}^w]) \\
\boldsymbol{h}^s &= \tanh(\boldsymbol{W}_s[(\hat{\boldsymbol{q}}^{\boldsymbol{s}} + \hat{\boldsymbol{v}}^{\boldsymbol{s}}), \boldsymbol{h}^p]) \\
\boldsymbol{p} &= \mathrm{softmax}(\boldsymbol{W}_h \boldsymbol{h}^s)
\end{aligned}
\tag{7}
$$

where $\boldsymbol{W}_w, \boldsymbol{W}_p, \boldsymbol{W}_s$ and $\boldsymbol{W}_h$ are the weight parameters. $[\cdot]$ is the concatenation operation on two vectors. $\boldsymbol{p}$ is the probability of the final answer.

4 Experiment

4.1 Datasets

We evaluate the proposed model on two datasets, the VQA dataset [2] and the COCO-QA dataset [14].

VQA dataset is the largest dataset for this problem, containing human annotated questions and answers on Microsoft COCO dataset [11]. The dataset contains 248,349 training questions, 121,512

validation questions and 244,302 testing questions. There are three sub-categories according to answer-types including yes/no, number, and other. Each question has 10 free-response answers. We use the top 1000 most frequent answers as the possible outputs similar to [2]. This set of answers covers 86.54% of the train+val answers. For testing, we train our model on VQA train+val and report the test-dev and test-standard results from the VQA evaluation server. We use the evaluation protocol of [2] in the experiment.

COCO-QA dataset is automatically generated from captions in the Microsoft COCO dataset [11]. There are 78,736 train questions and 38,948 test questions in the dataset. These questions are based on 8,000 and 4,000 images respectively. There are four types of questions including object, number, color, and location. Each type takes 70%, 7%, 17%, and 6% of the whole dataset, respectively. All answers in this data set are single word. We report the results both on classification accuracy and Wu-Palmer similarity (WUPS) measure [20] in Table 2.

4.2 Setup

We use torch [4] to develop our model. We use the Rmsprop optimizer [19] with a base learning rate of 4e-4, momentum 0.99 and weight-decay 1e-8. We set batch size to be 300 and train for up to 256 epochs with early stopping if the validation accuracy has not improved in the last 5 epochs. For COCO-QA, the size of hidden layer W_s is set to 512 and 1024 for VQA since it is a much larger dataset. All the other word embedding and hidden layers were vectors of size $d = 512$. We apply dropout with probability 0.5 on each layer. Following [23], we rescale the image to 448×448, and then take the activation from the last pooling layer of VGGNet [17] or Deep Residual network [6] as its feature. Our model is end-to-end trainable, but we do not finetune the CNN. The code is available at https://github.com/jiasenlu/HieCoAttenVQA.

4.3 Results and analysis

There are two test scenarios on VQA: open-ended and multiple-choice. The best performing method **deeper LSTM Q + norm I** from [2] is used as our baseline. For open-ended test scenario, we compare our method with the recent proposed **SMem** [22], **SAN** [23], **FDN** [9] and **DMN+** [21]. For multiple choice, we compare with **Region Sel.** [16] and **FDN** [9]. We compare with **2-VIS+BLSTM** [14], **IMG-CNN** [12] and **SAN** [23] on COCO-QA. We use Ours^p to refer to our parallel co-attention and Ours^a for alternating co-attention.

Table 1 shows results on the VQA test sets for both open-ended and multiple-choice settings. We can see that our approach outperforms all previous results, improving the state of art from 60.4% (DMN+ [21]) to 62.1% (Ours^a+Residual) on open-ended and from 64.2% (FDN [9]) to 66.1% (Ours^a+Residual) on multiple-choice. Notably, for the question type *Other* and *Num*, we achieve 3.4% and 1.4% improvement on open-ended questions, and 4.0% and 1.1% on multiple-choice questions. As we can see, deep residual features outperform or match VGG features in all cases. Note that our improvements are not solely due to the use of a better CNN. Specifically, FDN [9] also uses ResidualNet [6], but Ours^a+Residual outperforms it by 1.8% on test-dev. SMem [22] uses GoogLeNet [18] and the rest all use VGGNet [17], and Ours+VGG outperforms them by 0.2% on test-dev (DMN+ [21]).

Table 2 shows results on the COCO-QA test set. Similar to the result on VQA, our model improves the state-of-the-art from 61.6% (SAN(2,CNN) [23]) to 65.4% (Ours^a+Residual). We observe that parallel co-attention performs better than alternating co-attention in this setup. Due to significant computational requirements, we could not train $Ours^p$+Residual on VQA and COCO-QA. This is left as future work.

4.4 Ablation study

In this section, we perform ablation studies to quantify the role of each component in our model. Specifically, we re-train our approach by ablating certain components:

- *Image Attention alone*, where in a manner similar to previous works [23], we do not use any question attention. The goal of this comparison is to verify that our improvements are not the result of orthogonal contributions (say better optimization or better CNN features).

Table 1: Results on the VQA dataset. "-" indicates the results is not available.

| | Open-Ended | | | | | Multiple-Choice | | | | |
| | test-dev | | | | test-std | test-dev | | | | test-std |
Method	Y/N	Num	Other	All	All	Y/N	Num	Other	All	All
LSTM Q+I [2]	80.5	36.8	43.0	57.8	58.2	80.5	38.2	53.0	62.7	63.1
Region Sel. [16]	-	-	-	-	-	77.6	34.3	55.8	62.4	-
SMem [22]	80.9	37.3	43.1	58.0	58.2	-	-	-	-	-
SAN [23]	79.3	36.6	46.1	58.7	58.9	-	-	-	-	-
FDN [9]	**81.1**	36.2	45.8	59.2	59.5	**81.5**	39.0	54.7	64.0	64.2
DMN+ [21]	80.5	36.8	48.3	60.3	60.4	-	-	-	-	-
Ours^p+VGG	79.5	**38.7**	48.3	60.1	-	79.5	39.8	57.4	64.6	-
Ours^a+VGG	79.6	38.4	49.1	60.5	-	79.7	**40.1**	57.9	64.9	-
Ours^a+Residual	79.7	**38.7**	**51.7**	**61.8**	**62.1**	79.7	40.0	**59.8**	**65.8**	**66.1**

Table 2: Results on the COCO-QA dataset. "-" indicates the results is not available.

Method	Object	Number	Color	Location	Accuracy	WUPS0.9	WUPS0.0
2-VIS+BLSTM [14]	58.2	44.8	49.5	47.3	55.1	65.3	88.6
IMG-CNN [12]	-	-	-	-	58.4	68.5	89.7
SAN(2, CNN) [23]	64.5	48.6	57.9	54.0	61.6	71.6	90.9
Ours^p+VGG	65.6	49.6	61.5	56.8	63.3	73.0	91.3
Ours^a+VGG	65.6	48.9	59.8	56.7	62.9	72.8	91.3
Ours^a+Residual	**68.0**	**51.0**	**62.9**	**58.8**	**65.4**	**75.1**	**92.0**

- *W/O Conv*, where no convolution and pooling is performed to represent phrases. Instead, we stack another word embedding layer on the top of word level outputs. The goal of this comparison is to verify whether our 1D convolution+pooling strategy over words indeed better captures local information from question fragments.

- *W/O W-Atten*, where no word level co-attention is performed. We replace the word level attention with a uniform distribution. Phrase and question level co-attentions are still modeled.

- *W/O P-Atten*, where no phrase level co-attention is performed, and the phrase level attention is set to be uniform. Word and question level co-attentions are still modeled.

- *W/O Q-Atten*, where no question level co-attention is performed. We replace the question level attention with a uniform distribution. Word and phrase level co-attentions are still modeled.

Table 3 shows the comparison of our full approach w.r.t these ablations on the VQA validation set (test sets are not recommended to be used for such experiments). The **deeper LSTM Q + norm I** baseline in [2] is also reported for comparison. We can see that image-attention-alone does improve performance over the holistic image feature (**deeper LSTM Q + norm I**), which is consistent with findings of previous attention models for VQA [23, 21].

Comparing the full model w.r.t. ablated versions without word, phrase, question level attentions reveals a clear interesting trend – the attention mechanisms closest to the 'top' of the hierarchy (*i.e.* question) matter most, with a drop of 1.7% in accuracy if not modeled; followed by the intermediate level (*i.e.* phrase), with a drop of 0.3%; finally followed by the 'bottom' of the hierarchy (*i.e.* word), with a drop of 0.2% in accuracy. We hypothesize that this is because the question level is the 'closest' to the answer prediction layers in our model. Note that *all* levels are important, and our final model significantly outperforms not using any linguistic attention (1.1% difference between Full Model and Image Atten).

To further understand our proposed co-attention mechanism, we break down the performance of our full model as a function of question lengths. Our hypothesis is that our proposed question attention (and corresponding co-attention) mechanism should play a more important role for longer questions, because shorter pithier questions may contain all equally important words. Fig. 4 shows these results

Table 3: Ablation study on the VQA dataset using VGGNet [17].

Method	validation			
	Y/N	Num	Other	All
LSTM Q+I	**79.8**	32.9	40.7	54.3
Image Atten	79.8	33.9	43.6	55.9
W/O Conv	79.0	34.8	45.5	56.6
W/O Q-Atten	79.6	32.1	42.9	55.3
W/O P-Atten	79.5	34.1	45.4	56.7
W/O W-Atten	79.6	34.4	45.6	56.8
Full Model	79.6	**35.0**	**45.7**	**57.0**
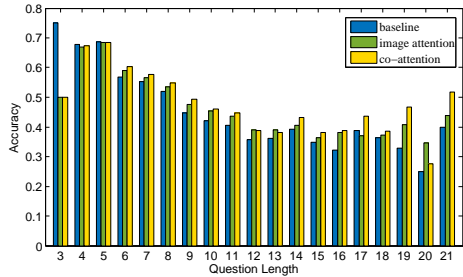
Figure 4: Performance at various question lengths.

using alternating co-attention (Ours[a]+VGG). We can see that while our model does provide larger improvements for longer questions (thus confirming our hypothesis), it works consistently well for all question lengths except for the boundary case of length 3, which has only two meaningful words (third token is the question mark) and occurs only 4 times in the VQA validation set.

4.5 Qualitative results

In this section, we visualize some co-attention maps of our generated by our method in Fig. 5. From top to bottom, the rows are original images and questions, word level co-attention maps, phrase level co-attention maps and question level co-attention maps. At the word level, our model attends mostly to the object regions in an image, e.g., heads, bird. On the question side, the attentions depend on the specific question being asked. In the examples, we see that some have relatively uniform attention while some have peaky attention. At the phrase level, the image attention has different patterns across images. For the first two images and the forth image, the attention transfers from objects to background regions. For the other two images, the attention becomes more focused on the objects. We suspect that this is caused by the different question types. For the former three images, the questions ask more about global attributes of the images, while the later two images have more object-specific questions. Comparing the question attention at the phrase level and word level, we find that the emphasis changes. Because we convolve and pool multiple n-grams from words to phrases, our model is capable of localizing the key important phrases in the question, thus essentially discovering the question types in the dataset. For example, our model pays attention to the phrases "what color", "how many snowboarders" and "what sport playing". At the question level, our model summarizes the whole question and then performs the co-attention. It successfully attends to the regions in images and phrases in the questions appropriate for answering the question, e.g., "color of the bird" and bird region, "what color ball" and the ball region. Because our model performs co-attention at three levels, it often captures complementary information from each level, and then combines them to predict the answer. More success and failure cases can be seen in Fig. 6 and 7 respectively.

5 Conclusion

In this paper, we proposed a hierarchical co-attention model for visual question answering. Co-attention allows our model to attend to different regions of the image as well as different fragments of the question. We model the question hierarchically at three levels to capture information from different granularities. The experimental results showed that our model outperforms the state-of-the-art on both the VQA and COCO-QA datasets. The ablation studies further demonstrate the roles of co-attention and question hierarchy in our final performance. Through visualizations, we can see that our model co-attends to interpretable regions of images and questions for predicting the answer. Though our model was evaluated on visual question answering, it can be potentially applied to other tasks involving vision and language.
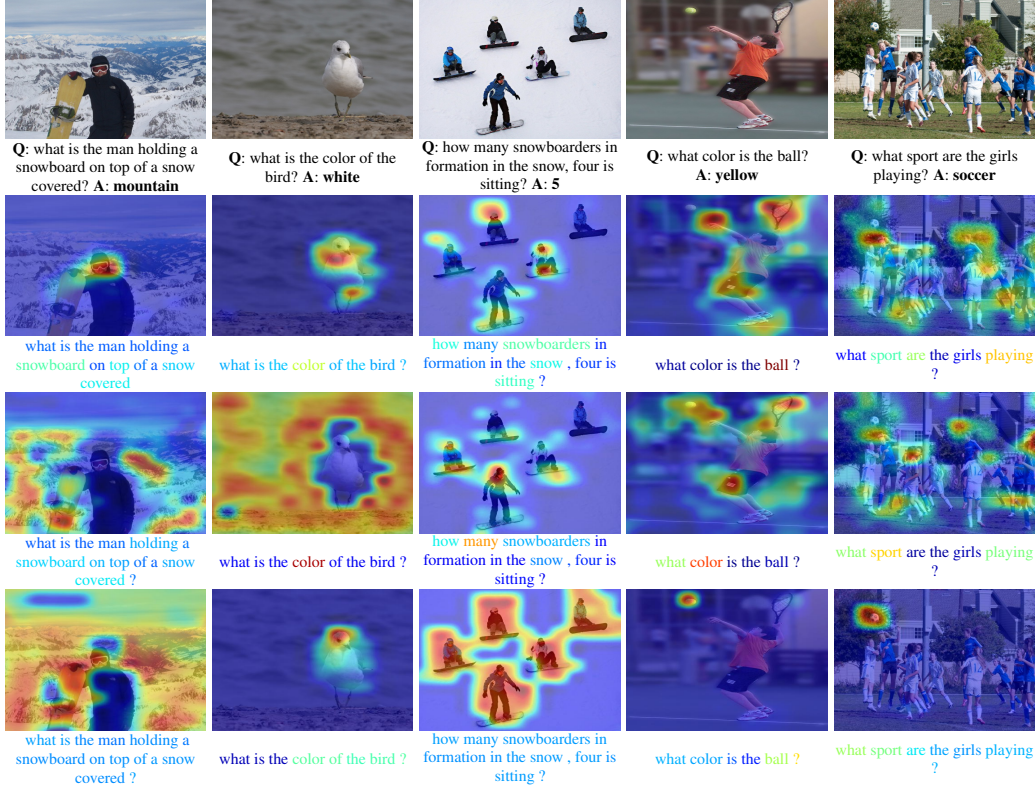
Figure 5: Visualization of image and question co-attention maps on the COCO-QA (first three columns using Ours^p+VGG) and VQA (last two columns using Ours^a+VGG) dataset. From top to bottom: original image and question pairs, word level co-attention maps, phrase level co-attention maps and question level co-attention maps. For visualization, both image and question attentions are scaled (from red:high to blue:low). Best viewed in color.

6   Acknowledgements

References

[1] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Deep compositional question answering with neural module networks. In *CVPR*, 2016.

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, 2015.

[3] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015.

[4] R. Collobert, K. Kavukcuoglu, and C. Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*, 2011.

[5] Haoyuan Gao, Junhua Mao, Jie Zhou, Zhiheng Huang, Lei Wang, and Wei Xu. Are you talking to a machine? dataset and methods for multilingual image question answering. In *NIPS*, 2015.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[7] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *NIPS*, 2015.

[8] Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. Convolutional neural network architectures for matching natural language sentences. In *NIPS*, 2014.

[9] Jiashi Feng Ilija Ilievski, Shuicheng Yan. A focused dynamic attention model for visual question answering. *arXiv:1604.01485*, 2016.

[10] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *arXiv preprint arXiv:1602.07332*, 2016.

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[12] Lin Ma, Zhengdong Lu, and Hang Li. Learning to answer questions from image using convolutional neural network. In *AAAI*, 2016.

[13] Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. Ask your neurons: A neural-based approach to answering questions about images. In *ICCV*, 2015.

[14] Mengye Ren, Ryan Kiros, and Richard Zemel. Exploring models and data for image question answering. In *NIPS*, 2015.

[15] Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomáš Kočiskỳ, and Phil Blunsom. Reasoning about entailment with neural attention. In *ICLR*, 2016.

[16] Kevin J Shih, Saurabh Singh, and Derek Hoiem. Where to look: Focus regions for visual question answering. In *CVPR*, 2016.

[17] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[18] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[19] T. Tieleman and G. Hinton. Lecture 6.5 - rmsprop, coursera: Neural networks for machine learning. In *Technical report*, 2012.

[20] Z. Wu and M. Palmer. Verb semantics and lexical selection. In *ACL*, 1994.

[21] Caiming Xiong, Stephen Merity, and Richard Socher. Dynamic memory networks for visual and textual question answering. In *ICML*, 2016.

[22] Huijuan Xu and Kate Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. *arXiv preprint arXiv:1511.05234*, 2015.

[23] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question answering. In *CVPR*, 2016.

[24] Wenpeng Yin, Hinrich Schütze, Bing Xiang, and Bowen Zhou. Abcnn: Attention-based convolutional neural network for modeling sentence pairs. In *ACL*, 2016.

[25] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. Visual7w: Grounded question answering in images. In *CVPR*, 2016.
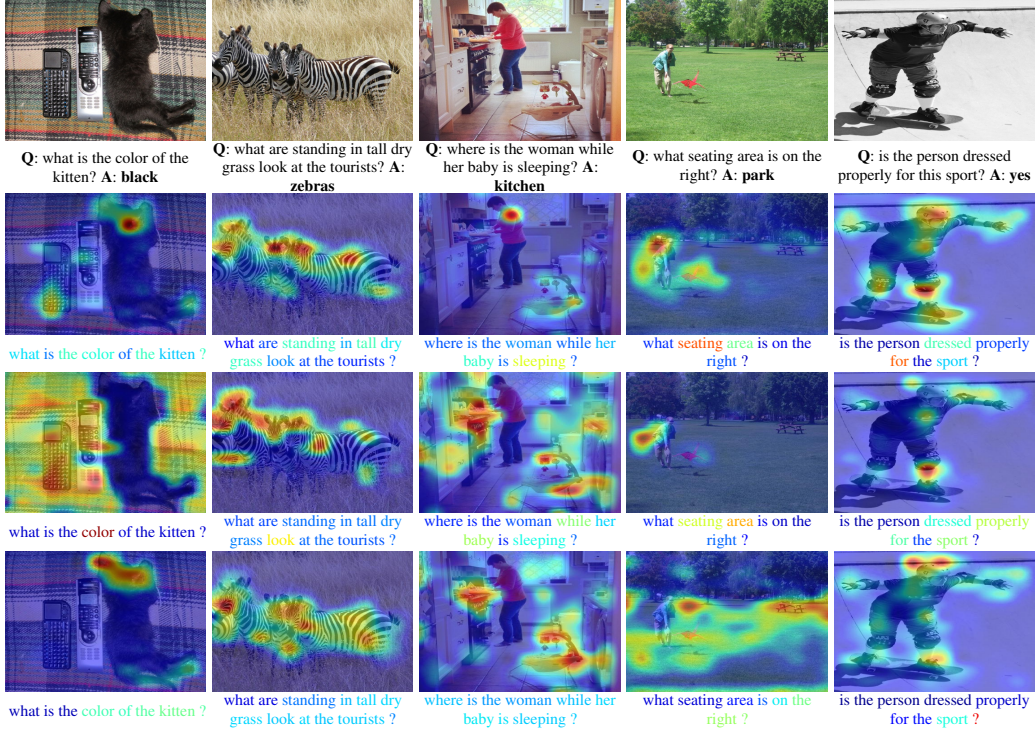
Figure 6: Visualization of co-attention maps on success cases in the COCO-QA (first three columns using Ours^p+VGG) and VQA (last two columns Ours^a+VGG) dataset. The layout is the same as Fig. 5.

Figure 7: Visualization of co-attention maps on failure cases in the COCO-QA (first three columns using Ours^p+VGG) and VQA (last two columns Ours^a+VGG) dataset. Predicted answer is in red and ground truth answer is in green. The layout is the same as Fig. 5.