Tell-and-Answer: Towards Explainable Visual Question Answering using Attributes and Captions

Qing Li[1], Jianlong Fu[2], Dongfei Yu[1], Tao Mei[3], Jiebo Luo[4]

[1]University of Science and Technology of China
[2]Microsoft Research, Beijing, China
[3]JD AI Research, Beijing 100105, China
[4]University of Rochester, Rochester, NY

[1]{sealq, ydf2010}@mail.ustc.edu.cn, [2]jianf@microsoft.com, [3]tmei@jd.com, [4]jluo@cs.rochester.edu

Abstract

In Visual Question Answering, most existing approaches adopt the pipeline of representing an image via pre-trained CNNs, and then using the uninterpretable CNN features in conjunction with the question to predict the answer. Although such end-to-end models might report promising performance, they rarely provide any insight, apart from the answer, into the VQA process. In this work, we propose to break up the end-to-end VQA into two steps: **explaining** and **reasoning**, in an attempt towards a more explainable VQA by shedding light on the intermediate results between these two steps. To that end, we first extract attributes and generate descriptions as explanations for an image. Next, a reasoning module utilizes these explanations in place of the image to infer an answer. The advantages of such a breakdown include: (1) the attributes and captions can reflect what the system extracts from the image, thus can provide some insights for the predicted answer; (2) these intermediate results can help identify the inabilities of the image understanding or the answer inference part when the predicted answer is wrong. We conduct extensive experiments on a popular VQA dataset and our system achieves comparable performance with the baselines, yet with added benefits of explanability and the inherent ability to further improve with higher quality explanations.

1 Introduction

Answering textual questions from images, which is referred to as visual question answering, presents fundamental challenges to both computer vision and natural language processing communities. Significant progress has been made on VQA in recent years (Antol et al., 2015; Zhu et al., 2016; Wu et al., 2016a; Yang et al., 2016; Goyal et al., 2017; Yu et al., 2017; Teney et al., 2017;
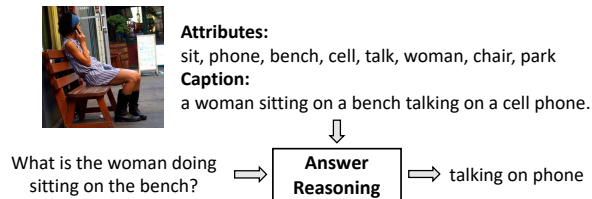

Figure 1: An example of explanation and reasoning in VQA. We first extract attributes in the image such as "sit", "phone" and "woman." A caption is also generated to encode the relationship between these attributes, e.g. "woman sitting on a bench." Then a reasoning module uses these explanations to predict an answer "talking on phone."
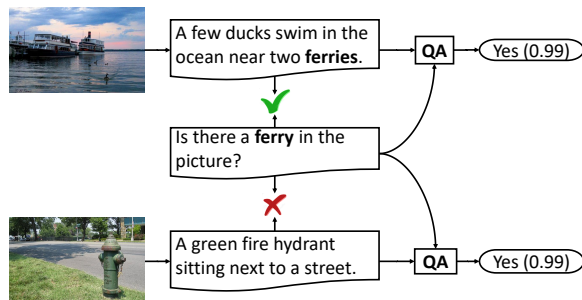

Figure 2: Two contrasting cases that show how the explanations can be used to determine if the system *guesses* the answer.

Wang et al., 2017; Gurari et al., 2018; Anderson et al., 2018; Lu et al., 2018; Li et al., 2018). A widely used pipeline is to first encode an image with Convolutional Neural Networks (CNNs) and represent associated questions with Recurrent Neural Networks (RNNs), and then formulate the vision-to-language task as a classification problem on a list of answer candidates. Although promising performance has been reported, this end-to-end paradigm fails to provide any insight to illuminate the VQA process. In most cases, giving answers without any explanation cannot satisfy hu-

man users, especially when the predicted answer is not correct. More frustratingly, the system gives no hint about which part of such systems is the culprit for a wrong answer.

To address the above issues, we propose to break up the popular end-to-end pipeline into two steps: **explaining** and **reasoning**. The philosophy behind such a break-up is to mimic the image question answering process of human beings: first understanding the content of the image and then performing inference about the answer according to the understanding. As is shown in Fig.1, we first generate two-level explanations for an image via pre-trained attribute detectors and image captioning model: 1). word-level: attributes, indicating individual objects and attributes the system learns from the image. 2). sentence-level: captions, representing the relationship between the objects and attributes. Then the generated explanations and question are infused to a reasoning module to predict an answer. The reasoning module is mainly composed of LSTMs.

Our method has three benefits. First, these explanations are interpretable. According to the attributes and captions, we can tell what objects, attributes and their relationship the machine learns from the image as well as what information is lost during the image understanding step. In contrast, the fully-connected layer features of CNNs are usually uninterpretable to humans. When the predicted answer is correct, these attributes and captions can be provided for users as the supplementary explanations to the answer. Second, the separation of explaining and reasoning enables us to localize which step of the VQA process the error comes from when the predicted answer is wrong. If the explanations don't include key information to answer the question, the error is caused by missing information during the explaining step. Otherwise, the reasoning module should be responsible for the wrong answer. Third, the explanations can also indicate whether the system really finds key information from the image to answer the question or merely *guesses* an answer. Fig.2 presents two contrasting cases to illustrate this. In the first case, both the generated caption and the question include the key concept "ferry", so the answer "Yes" with a high probability is reliable. However, although the answer "Yes" has the same high probability in the second case, the caption is irrelevant to the question. The system sticks to

a wrong answer even with the correct input from sentence generation. This is due to the training set bias that a large proportion of questions starting with "is there" in the training set have the answer "Yes".

To our knowledge, this is the first effort to break down the previous end-to-end pipeline to shed light on the VQA process. Our main contributions are summarized as follows:

- We propose to formulate VQA into two separate steps: **explaining** and **reasoning**. Our framework generate attributes and captions for images to shed light on why the system predicts any specific answer.
- We adopt several ways to measure the explanation quality and demonstrate strong correlation between explanation quality and VQA accuracy. The current system achieves comparable performance to the baselines and can naturally improve with explanation quality.
- Extensive experiments are conducted on the popular VQA dataset (Antol et al., 2015). We dissect all results according to the measurements of the quality of explanations to present a thorough analysis of the strength and weakness of our framework.

2 Related Work

There is a growing research interest in the task of visual question answering. In this section, we summarize recent advances from two directions.

Attention in VQA. The attention mechanism is firstly used in the machine translation task (Bahdanau et al., 2014) and then is brought into the vision-to-language tasks (Xu et al., 2015; You et al., 2016; Yang et al., 2016; Lu et al., 2016; Nam et al., 2017; Yu et al., 2017; Teney et al., 2017; Anderson et al., 2018; Teney et al., 2018; Liang et al., 2018). The visual attention in the vision-to-language tasks is used to address the problem of "where to look". In VQA, the question is used as a query to search for the relevant regions in the image. Yang *et al.* propose a stacked attention model which queries the image for multiple times to infer the answer progressively. Beyond the visual attention, Lu *et al.* exploit a hierarchical question-image co-attention strategy to attend to both related regions in the image and crucial words in the question. Attention mechanism can find the question-related regions in the image, which accounts for the answer to some extent. But the at-

tended regions still don't explicitly exhibit what the system learns from the image and it is also not explained why these regions should be attended to. **High-level Concepts.** In the scenario of vision-to-language, high-level concepts exhibit superior performance than the low-level or middle-level visual features of the image (Fang et al., 2015; Wu et al., 2016a,b). (Fang et al., 2015) first learn independent detectors for visual words based on a multi-instance learning framework and then generate descriptions for images based on the set of visually detected words via a maximum entropy language model. (Wu et al., 2016a,b) presents a thorough study on how much the high-level concepts can benefit the image captioning and visual question answering tasks. These work mainly uses high-level concepts to obtain a better performance. Different from these work, our paper is focused on fully exploiting the readability and understandability of attributes and captions to explain the process of visual question answering and use these explanations to analyze our system.

3 Methodology

In this section, we introduce the proposed framework for the breakdown of VQA. As illustrated in Figure 3, the framework consists of three modules: word prediction, sentence generation, and answer reasoning. Next, we describe the three modules in details.

3.1 Word Prediction

From the work (Wu et al., 2016a), we have learned that explicit high-level attributes can benefit vision-to-language tasks. In fact, besides performance gain, the readability and understandability of attributes also makes them an intuitive way to explain what the model learns from images.

We first build a word list based on MS COCO Captions (Chen et al., 2015). We extract the most N frequent words in all captions and filter them by lemmatization and removing stop words to determine a list of 256 words, which cover over 90% of the word occurrences in the dataset. Our words are not tense or plurality sensitive, for example, "horse" and "horses" are considered as the same word. This significantly decreases the size of our word list. Given the word list, every image is paired with multiple labels (words) according to its captions. Then we formulate word prediction as a multi-label classification task and fine-tune the

ResNet-152 (He et al., 2016) on our image-words dataset by minimizing the element-wise sigmoid cross entropy loss:

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

(1)

where N is batch size, V is the size of word list, $\mathbf{y}_i = [y_{i1}, y_{i2}, ..., y_{iV}], y_{ij} \in \{0, 1\}$ is the label vector of the ith image, $\mathbf{p}_i = [p_{i1}, p_{i2}, ..., p_{iV}]$ is the probability vector.

In the testing phase, instead of using region proposals like (Wu et al., 2016a), we directly feed the whole image into the word prediction CNN in order to keep simple and efficient. As a result, each image is encoded into a fixed-length vector, where each dimension represents the probability of the corresponding word occurring in the image.

Word Quality Evaluation. We adopt two metrics to evaluate the predicted words. The first measures the **accuracy** of the predicted words by computing cosine similarity between the label vector $\mathbf{y}$ and the probability vector $\mathbf{p}$:

$$a = \frac{\mathbf{y}^T \mathbf{p}}{||\mathbf{y}|| \cdot ||\mathbf{p}||}$$

(2)

However, this metric disregards the extent to which the predicted words are relevant to the question. Intuitively speaking, question-relevant explanations for images should be more likely to help predict right answers than irrelevant ones. Therefore, we propose another metric to measure the **relevance** between the words and the question. We first encode the question into a 0-1 vector $\mathbf{q}$ in terms of the word list. Then the relevance is computed as:

$$r = \frac{\mathbf{q}^T \mathbf{p}}{||\mathbf{q}|| \cdot ||\mathbf{p}||}$$

(3)

3.2 Sentence Generation

This section we talk about generating sentence-level explanations for images by using a pre-trained image captioning model. Similar to (Vinyals et al., 2015), we train an image captioning model by maximizing the probability of the correct caption given an image. Suppose we have an image $\mathcal{I}$ to be described by a caption $\mathcal{S} = \{s_1, s_2, ..., s_L\}, s_t \in \mathcal{V}$, where $\mathcal{V}$ is the vocabulary, L is the caption length. First the image $\mathcal{I}$ is represented by the activations of the first
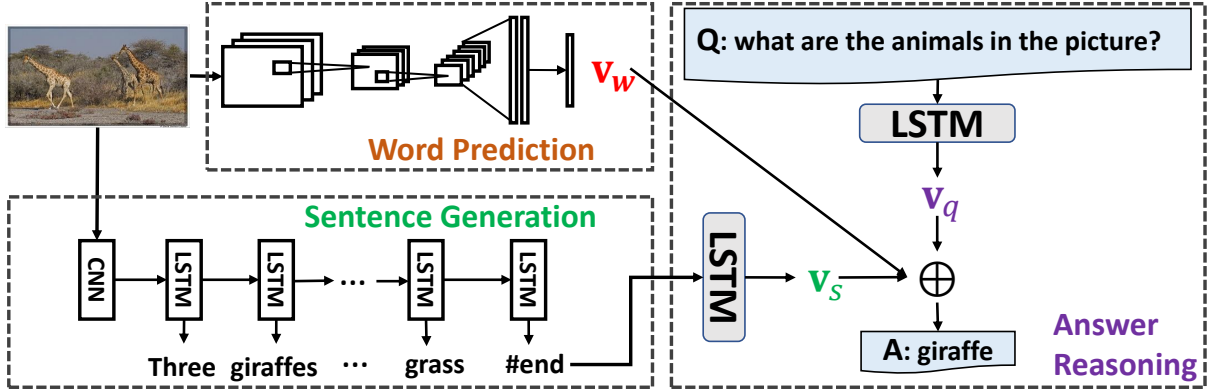
Figure 3: An overview of the proposed framework for VQA with three modules: word prediction (upper left), sentence generation (lower left), answer reasoning (right). **Explaining**: in word prediction, the image is fed into pre-trained visual detectors to extract word-level explanation, which is represented by probability vector $\mathbf{v}_w$; in sentence generation, we input the image to pre-trained captioning model to generate a sentence-level explanation. **Reasoning**: the caption and question are encoded by two different LSTMs into $\mathbf{v}_s$ and $\mathbf{v}_q$, respectively. Then $\mathbf{v}_q$, $\mathbf{v}_w$ and $\mathbf{v}_s$ are concatenated and fed to a fully connected layer with softmax to predict an answer.

fully connected layer of ResNet-152 pre-trained on ImageNet, denoted as $\mathbf{v}_i$. The caption $\mathcal{S}$ can be represented as a sequence of one-hot vector $\mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_L\}$. Then we formulate the caption generation problem as minimizing the cost function:

$$
\begin{aligned}
J(\mathbf{v}_i, \mathbf{S}) &= -\log P(\mathbf{S}|\mathbf{v}_i) \\
&= -\sum_{t=0}^{L} \log P(\mathbf{s}_t|\mathbf{v}_i, \mathbf{s}_1, ..., \mathbf{s}_{t-1})
\end{aligned}
\quad (4)
$$

where $P(\mathbf{s}_t|\mathbf{v}_i, \mathbf{s}_1, ..., \mathbf{s}_{t-1})$ is the probability of generating the word $\mathbf{s}_t$ given the image representation $\mathbf{v}_i$ and previous words $\{\mathbf{s}_1, ..., \mathbf{s}_{t-1}\}$. We employ a single-layer LSTM with 512-dimensional hidden states to model this probability. In the testing phase, the image is input to pre-trained image captioning model to generate sentence-level explanation.

Sentence Quality Evaluation. Similar to word quality evaluation, we evaluate the quality of the generated sentence from two perspectives: **accuracy** and **relevance**. The former one is an average fusion of four widely used metrics: BLEU@N, METEOR, ROUGE-L and CIDEr-D (Chen et al., 2015), which try to consider the accuracy of the generated sentence from different perspectives. Note that we normalize all the metrics into $[0, 1]$ before fusion. The latter metric is to measure the relevance between the generated sentence and the question. The binary TF weights are calculated over all words of the sentence to produce an integrated representation of the entire sentence, de-

noted by $\mathbf{s}$. Likewise, the question can be encoded to $\mathbf{q}$. The relevance is computed as:

$$
r = \frac{\mathbf{q}^T \mathbf{s}}{||\mathbf{q}|| \cdot ||\mathbf{s}||} \quad (5)
$$

3.3 Answer Reasoning

This section we discuss the reasoning module. Suppose we have an image $\mathcal{I}$ explained by the predicted words $\mathcal{W}$ and the generated sentence $\mathcal{S}$, the question $\mathcal{Q}$ and the answer $\mathcal{A}$. As shown in Fig.3, we denote the representations of the predicted words $\mathcal{W}$ as $\mathbf{v}_s$. The caption $\mathcal{S}$ and question $\mathcal{Q}$ are encoded by two different LSTMs into $\mathbf{v}_s$ and $\mathbf{v}_q$, respectively. What bears mentioning is that these two LSTMs share a common word-embedding matrix, but not other parameters, because the question and caption have different grammar structures and similar vocabularies. At last, the $\mathbf{v}_w$, $\mathbf{v}_s$, and $\mathbf{v}_q$ are concatenated and fed into a fully connected layer with softmax to predict the probability on a set of candidate answers:

$$
\mathbf{v} = [\mathbf{v}_w^T \ \mathbf{v}_s^T \ \mathbf{v}_q^T]^T \quad (6)
$$

$$
\mathbf{p} = softmax(\mathbf{W}\mathbf{v} + \mathbf{b}) \quad (7)
$$

where $\mathbf{W}, \mathbf{b}$ are the weight matrix and bias vector of the fully connected layer. The optimizing objective for the reasoning module is to minimize the cross entropy loss as:

$$
J(\mathcal{I}, \mathcal{Q}, \mathcal{A}) = J(\mathcal{W}, \mathcal{S}, \mathcal{Q}, \mathcal{A}) = -\log \mathbf{p}(\mathcal{A}) \quad (8)
$$

where $\mathbf{p}(\mathcal{A})$ denotes the probability of the ground truth $\mathcal{A}$.

4 Experiments and Analysis

4.1 Experiment Setting

Dataset. We evaluate our framework on VQA-real (Antol et al., 2015) dataset. For each image in VQA-real, 3 questions are annotated by different workers and each question has 10 answers from different annotators. We follow the official split and report our results on the open-ended task.

Metric. We use the accuracy: $\min(\frac{\#\text{humans giving that answer}}{3}, 1)$, i.e., an answer is deemed 100% accurate if at least three workers provided that exact answer.

Ablation Models. To analyze the contribution of word-level and sentence-level explanations, we ablate the full model and evaluate several variants as:

- **Word-based VQA**: use the feature concatenation of the predicted words and question in Eq.6.

- **Sentence-based VQA**: use the feature concatenation of the generated sentence and question in Eq.6.

- **Full VQA**: use the feature concatenation of words, sentence, and question in Eq.6.

4.2 Word-based VQA

Table 1: The relationship between word quality and accuracy (%).

(a) Word accuracy		(b) W-Q Relevance	
Word accuracy	VQA accuracy	W-Q Relevance	VQA accuracy
[0.0, 0.2)	46.30	[0.0, 0.2)	54.69
[0.2, 0.8)	55.84	[0.2, 0.8)	60.23
[0.8, 1.0)	58.52	[0.8, 1.0)	76.15

An important characteristics of our framework is that the quality of explanations can influence the final VQA performance. In this section, we analyze the impact of the quality of predicted words on the VQA accuracy. We measure the quality from two sides: word accuracy and word-question relevance. Table 1a shows the relationship between word accuracy and VQA performance. We can learn that the more accurate the predicted words, the better the VQA performance. Similar to word accuracy, the more relevant to the question the predicted words, the better the VQA performance. Particularly, when the word-question relevance exceeds 0.8, the predicted words are highly
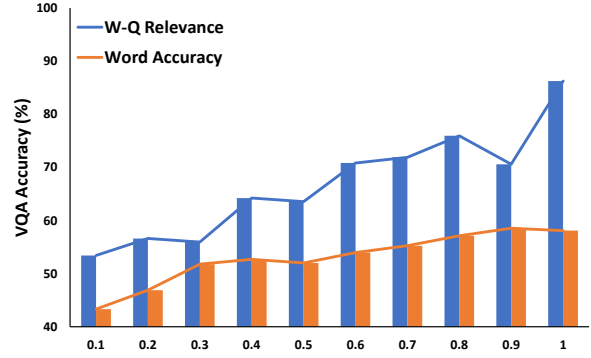

Figure 4: The comparison of the impact of word accuracy and word-question relevance on VQA performance.
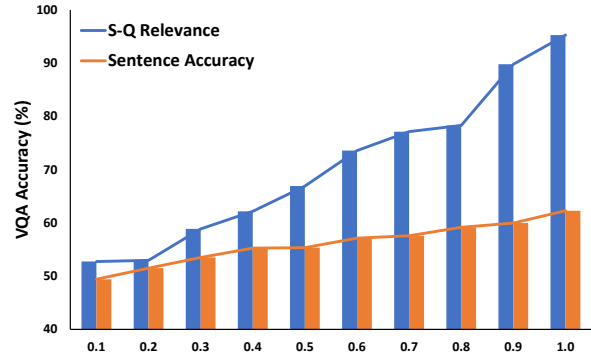

Figure 5: The comparison of the impact of sentence accuracy and sentence-question relevance on VQA performance.

pertinent to the question, boosting the VQA accuracy to 76.15%. This indicates high-quality word-level explanations can benefit the VQA performance a lot. As shown in Fig.4, word-question relevance has a bigger impact on the final VQA performance than word accuracy.

4.3 Sentence-based VQA

Table 2: The relationship between sentence quality and accuracy.

(a) Sentence accuracy		(b) S-Q Relevance	
Sentence accuracy	VQA accuracy	S-Q Relevance	VQA accuracy
[0.0, 0.2)	51.29	[0.0, 0.2)	52.79
[0.2, 0.8)	55.53	[0.2, 0.8)	62.34
[0.8, 1.0)	61.33	[0.8, 1.0)	89.81

In this section, we evaluate the sentence-based VQA model and analyze the relationship between the sentence quality and the VQA performance.

Table 3: Performance comparison on the validation split of VQA-real open-ended task when the sentence-based VQA model uses different sources of captions. (accuray in %)

| Caption source | validation | | | |
	All	Y/N	Num	Others
null	46.21	73.82	34.98	25.63
sentence generation	54.85	76.31	36.64	42.23
relevant groundtruth	56.05	77.42	41.04	44.34

Similar to the quality measurements of predicted words, we focus on the accuracy of the generated sentence itself and the relevance between sentence and question. As shown in Table 2a, the more accurate the generated sentence, the higher the VQA accuracy. The results suggest that the VQA performance can be further improved by a better image captioning model. From Table 2b, we can see that the more relevant to the question the generated sentence, the better the VQA performance. Once the relevance reaches 0.8, the accuracy can significantly increase to 89.81%. This proves that a question-related sentence is more likely to contain the key information for the VQA module to answer the question. As shown in Fig. 5, sentence-question relevance has greater influence on VQA performance than sentence accuracy does.

To further verify the causal relationship between sentence quality and VQA performance, we conduct the following control experiments. First, we evaluate sentence-based VQA model when feeding different sources of captions with ascending quality: *null* (only including an "#end" token), *sentence generation* and *relevant groundtruth* (selecting from the groundtruth captions the most relevant one to the question). As shown in Table 3, *sentence generation* performs much better than *null*. And using *relevant groundtruth* captions, the accuracy can improve by another 1.2 percent. Figure 6 presents an example to illustrate the effect of the sentence quality on the accuracy. From the above analysis, we can safely reach the conclusion that the VQA performance can be greatly improved by generating sentence-level explanations of high quality, especially of high relevance to the question.

4.4 Case Study

From the above evaluation of word-based and sentence-based VQA model, we conclude

Image and question	Generated Caption	Prediction (accuracy)
Q: what sport are they playing?	(**Good**) a group of people playing frisbee in a field.	frisbee (1.00)
	(**Wrong**) a group of people playing soccer in a field.	soccer (0.00)
	(**Empty**) *NULL*	tennis (0.00)

Figure 6: A control case for comparing the accuracy when inputting captions of different quality. When getting a caption of high quality (the first one), the system can answer the question correctly. If we manually change the "frisbee" to "soccer", a wrong answer is predicted. When using an empty sentence, the system predicts the most popular answer "tennis" for this question.

that the relevance between explanations (attributes/caption) and the question has a great impact on the final VQA performance. In this section we illustrate this conclusion by studying four possible types of cases: 1). *high relevance and correct answer*; 2). *low relevance and wrong answer*; 3). *high relevance but wrong answer*; 4). *low relevance but correct answer*.

High relevance and correct answer. From the first case in Fig. 7, we can see that the explanations for the image are highly relevant to the question: both the predicted attributes and the generated sentence contain the words "man" and "racket" occurring in the question. And the explanations also has key information that can predict the answer "tennis court." In this type of case, the system successfully extracts from the image the relevant information that covers the question, facilitating answer generation.

Low relevance and wrong answer. In the second case, although the attributes and caption can reflect part of the image content such as "man" and "food", they neglect the key information about the "glass" that is asked in the question. The absence of "glass" in the explanations produces a low explanation-question relevance score and leads the system to a wrong answer. In this type of case, two lessons can be derived from the low relevance: 1). as the explanations are irrelevant to the question, the system tends to predict the most frequent answer ("beer") for this question type ("what kind of drink ..."), which implies that the answer is actually *guessed* from the dataset bias; 2). the error comes from the image understanding part rather than the question answering module, because the system fails to extract from the image
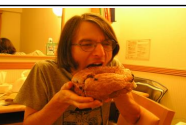
Image	Explanations, Question and Answer
① **High relevance and correct answer**	
	Attributes: tennis, ball, man, racket, hit, court, play, player, swing, hold (0.87, 0.72) **Caption**: a man holding a tennis racket on a tennis court. (0.88, 0.68) **Question**: where is the man swinging the racket? **Answer**: tennis court (tennis court)
② **Low relevance and wrong answer**	
	Attributes: bicycle, man, sit, eat, bike, look, outside, food, person, table (0.06, 0.48) **Caption**: a man sitting at a table with a plate of food. (0.00, 0.34) **Question**: what kind of drink is in the glass? **Answer**: beer (water)
③ **High relevance but wrong answer**	
	Attributes: street, bus, cow, city, walk, car, drive, stand, road, white (0.51, 0.77) **Caption:** a cow that is walking in the street (0.51, 0.55) **Question:** what is walking next to the bus? **Answer:** car (cow)
④ **Low relevance but correct answer**	
	Attributes: woman, bear, teddy, hold, sit, glass, animal, large, lady (0.00, 0.19) **Caption:** a woman holding a sandwich in her hands (0.00, 0.30) **Question:** does the man need a haircut? **Answer:** yes (yes)

Figure 7: Four types of cases in our results: 1). high relevance and correct answer; 2). low relevance and wrong answer; 3). high relevance but wrong answer; 4). low relevance but correct answer. "(*,*)" behind the explanations (attributes/caption) denotes the explanation-question relevance score and explanation accuracy, respectively. Gray denotes groundtruth answers.
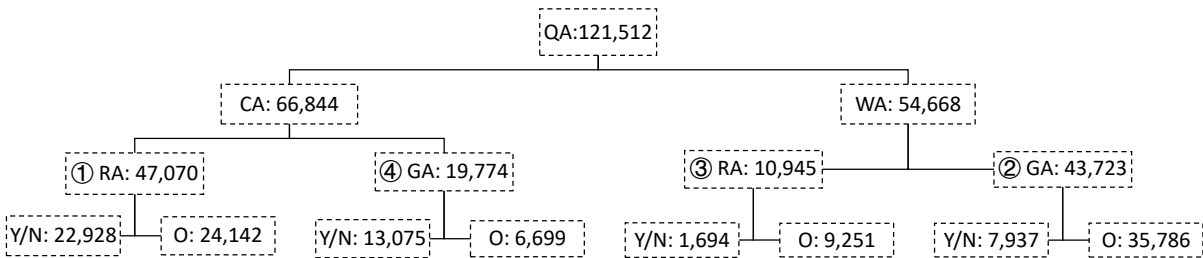

Figure 8: Dataset dissection according to the four types of cases. We define that the answer is guessed when the explanations are irrelevant to the question and otherwise reliable. The case numbers in the third row correspond to these in Fig.7. QA: all questions and answers. CA: questions with correct answers. WA: questions with wrong answers. GA: questions with guessed answers. RA: questions with reliable answers. Y/N: answer type "yes/no". O: answer types other than "yes/no".

enough information to answer the question in the first place. This error suggests that some improvements are needed in word prediction and sentence generation modules to generate more comprehensive explanations for the image.

High relevance but wrong answer. In the third case, we can see that although the system fails to predict the correct answer, the explanations for the image are indeed relevant to the question and the system also recognize the key information "cow." This indicates that the error is caused by the question answering module rather than the explanation

generation part. The system can recognize that "a cow is walking in the street" and "a bus is in the street", but it fails to conclude that "the cow is next to the bus." This error may lie in the weakness of LSTM which struggles on such complex spatial relationship inference. In the following analysis, we would show that such cases only occupy a relatively small proportion of the whole dataset.

Low relevance but correct answer. In the last example of Fig. 7, we know from the explanations that the system mistakes the "man" in the image for "woman" and neglects the information

about his "hair." The explanations, therefore, have a low relevance score, which indicates that the answer "yes" is guessed by the system. Although the guessed answer is correct, it cannot be credited to the correctness of the system. In fact, for this particular answer type "yes/no", the system has at least 50% chance to hit the right answer.

We dissect all the results in the dataset according to the above four types of cases, as shown in Fig. 8. Among the questions that the system answers correctly, nearly 30% are guessed. This discovery indicates that, buried in the seemingly promising performance, the system actually takes advantage of the dataset bias, rather than truly understands the image content. Over 65% of the answers that are correctly guessed belong to "yes/no", an answer type easier for the system to hit the right answer than other types. As for the questions to which the system predicts wrong answers, a large proportion (around 80%) has a low explanation-question relevance, which means that more efforts need to be put into improving the attributes detectors and image captioning model. Questions with other answer types account for more than 80% of the wrongly-guessed answers. This is not surprising because for these questions the system cannot rely on the dataset bias anymore, considering the great variety of the candidate answers.

4.5 Performance Comparison

Table 4: Performance comparison with the baselines. We show the performance on both test-dev and test-standard splits of VQA-real open-ended task. Human is the human performance for reference.

Method	test-dev				test-standard			
	All	Y/N	Num	Others	All	Y/N	Num	Others
LSTM Q+I (Antol et al., 2015)	53.74	78.94	35.24	36.42	54.06	79.01	35.55	36.80
Concepts (Wu et al., 2016a)	57.46	79.77	36.79	43.10	57.62	79.72	36.04	43.44
ACK (Wu et al., 2016b)	59.17	81.01	38.42	45.23	59.44	81.07	37.12	45.83
MCB w/n attention (Fukui et al., 2016)	60.80	81.20	35.10	49.30	-	-	-	-
Human (Antol et al., 2015)	-	-	-	-	83.30	95.77	83.39	72.67
Word-based VQA	56.76	77.57	35.21	43.85	-	-	-	-
Sentence-based VQA	57.91	78.03	36.73	45.52	-	-	-	-
Full VQA	59.93	79.32	38.41	48.25	60.07	79.09	38.25	48.57

In this section, we present the performance comparison between variants of our framework and the baselines. From Table 4, we can see that sentence-based VQA consistently outperforms word-based VQA, which indicates that sentence-level explanations are superior to word-level ones. The generated captions not only include the objects in the image, but also encode the relationship between these objects, which is important for predicting the correct answer. Furthermore, full VQA model obtains a better performance by combining attributes and captions.

Compared with the baselines, our framework achieves better performance than LSTM Q+I (Antol et al., 2015), Concepts (Wu et al., 2016a), and ACK (Wu et al., 2016b), which use CNN features, high-level concepts, and external knowledge, respectively. MCB without attention (Fukui et al., 2016) achieves better performance than ours and other methods, but it suffers from a high-dimensional feature (16,000 vs 1,280), which poses a limitation on the model's efficiency. The main advantage of our framework over other methods is that it not only predicts an answer to the question, but also generates human-readable attributes and captions to explain the answer. These explanations can help us understand what the system extracts from an image and their relevance to the question. As explanations improve, so would our system.

5 Discussions and Conclusions

In this work, we break up the end-to-end VQA pipeline into **explaining** and **reasoning**, and achieve comparable performance with the baselines. Different from previous work, our method first generates attributes and captions as explanations for an image and then feed these explanations to a question answering module to infer an answer. The merit of our method lies in that these attributes and captions allow a peek into the process of visual question answering. Furthermore, the relevance between these explanations and the question can act as indication whether the system really understands the image content.

It is worth noting although we also use the CNN-RNN combination, we generate words and captions as the explanations of images, thus allowing the VQA system to perform *reasoning on semantics* instead of unexplainable CNN features. Since the effectiveness of CNN for generating attributes and captions is well established, the use of CNN as a component does not contradict our high-level objective for explainable VQA. Our goal is not to immediately make a big gain in performance, but to propose a more powerful framework for VQA. Our current implementation already matches the baselines, but more importantly, provides the ability to explain and to improve.

References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *ICCV*, pages 2425–2433.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *ICLR*.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server. *CoRR*.

Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh K Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C Platt, et al. 2015. From captions to visual concepts and back. In *CVPR*.

Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. Multimodal compact bilinear pooling for visual question answering and visual grounding. *EMNLP*.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. *CVPR*.

Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *CVPR*.

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

Qing Li, Qingyi Tao, Shafiq Joty, Jianfei Cai, and Jiebo Luo. 2018. Vqa-e: Explaining, elaborating, and enhancing your answers for visual questions. *ECCV*.

Junwei Liang, Lu Jiang, Liangliang Cao, Li-Jia Li, and Alexander G. Hauptmann. 2018. Focal visual-text attention for visual question answering. In *CVPR*.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. In *NIPS*, pages 289–297.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2018. Neural baby talk. In *CVPR*, pages 7219–7228.

Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. 2017. Dual attention networks for multimodal reasoning and matching. *CVPR*.

Damien Teney, Peter Anderson, Xiaodong He, and Anton van den Hengel. 2018. Tips and tricks for visual question answering: Learnings from the 2017 challenge. In *CVPR*.

Damien Teney, Lingqiao Liu, and Anton van den Hengel. 2017. Graph-structured representations for visual question answering. In *CVPR*.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *CVPR*, pages 3156–3164.

Peng Wang, Qi Wu, Chunhua Shen, and Anton van den Hengel. 2017. The vqa-machine: Learning how to use existing vision algorithms to answer new questions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Qi Wu, Chunhua Shen, Lingqiao Liu, Anthony Dick, and Anton van den Hengel. 2016a. What value do explicit high level concepts have in vision to language problems? In *CVPR*.

Qi Wu, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2016b. Ask me anything: Free-form visual question answering based on knowledge from external sources. *CVPR*.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C Courville, Ruslan Salakhutdinov, Richard S Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, volume 14, pages 77–81.

Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. 2016. Stacked attention networks for image question answering. In *CVPR*, pages 21–29.

Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. 2016. Image captioning with semantic attention. In *CVPR*.

Dongfei Yu, Jianlong Fu, Tao Mei, and Yong Rui. 2017. Multi-level attention networks for visual question answering. In *CVPR*.

Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. 2016. Visual7w: Grounded question answering in images. In *CVPR*, pages 4995–5004.