Improving Visual Question Answering by Referring to Generated Paragraph Captions

Hyounghun Kim **Mohit Bansal**
UNC Chapel Hill
{hyounghk, mbansal}@cs.unc.edu

Abstract

Paragraph-style image captions describe diverse aspects of an image as opposed to the more common single-sentence captions that only provide an abstract description of the image. These paragraph captions can hence contain substantial information of the image for tasks such as visual question answering. Moreover, this textual information is complementary with visual information present in the image because it can discuss both more abstract concepts and more explicit, intermediate symbolic information about objects, events, and scenes that can directly be matched with the textual question and copied into the textual answer (i.e., via easier modality match). Hence, we propose a combined Visual and Textual Question Answering (VTQA) model which takes as input a paragraph caption as well as the corresponding image, and answers the given question based on both inputs. In our model, the inputs are fused to extract related information by cross-attention (early fusion), then fused again in the form of consensus (late fusion), and finally expected answers are given an extra score to enhance the chance of selection (later fusion). Empirical results show that paragraph captions, even when automatically generated (via an RL-based encoder-decoder model), help correctly answer more visual questions. Overall, our joint model, when trained on the Visual Genome dataset, significantly improves the VQA performance over a strong baseline model.

1 Introduction

Understanding visual information along with natural language have been studied in different ways. In visual question answering (VQA) (Antol et al., 2015; Goyal et al., 2017; Lu et al., 2016; Fukui et al., 2016; Xu and Saenko, 2016; Yang et al., 2016; Zhu et al., 2016; Anderson et al., 2018), models are trained to choose the correct answer given a question about an image. On the other hand, in image captioning tasks (Karpathy and Fei-Fei, 2015; Johnson et al., 2016; Anderson et al., 2018; Krause et al., 2017; Liang et al., 2017; Melas-Kyriazi et al., 2018), the goal is to generate sentences which should describe a given image. Similar to the VQA task, image captioning models should also learn the relationship between partial areas in an image and the generated words or phrases. While these two tasks seem to have different directions, they have the same purpose: understanding visual information with language. If their goal is similar, can the tasks help each other?

In this work, we propose an approach to improve a VQA model by exploiting textual information from a paragraph captioning model. Suppose you are assembling furniture by looking at a visual manual. If you are stuck at a certain step and you are given a textual manual which more explicitly describes the names and shapes of the related parts, you could complete that step by reading this additional material and also by comparing it to the visual counterpart. With a similar intuition, paragraph-style descriptive captions can more explicitly (via intermediate symbolic representations) explain what objects are in the image and their relationships, and hence VQA questions can be answered more easily by matching the textual information with the questions.

We provide a VQA model with such additional 'textual manual' information to enhance its ability to answer questions. We use descriptive captions generated from a paragraph captioning model which capture more detailed aspects of an image than a single-sentence caption (which only conveys the most obvious or salient single piece of information). We also extract properties of objects, i.e., names and attributes from images to create simple sentences in the form of "[object name] is [attribute]". Our VTQA model takes

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3606–3612
Florence, Italy, July 28 - August 2, 2019.

these paragraph captions and attribute sentences as input in addition to the standard input image features. The VTQA model combines the information from text and image with early fusion, late fusion, and later fusion. With early fusion, visual and textual features are combined via cross-attention to extract related information. Late fusion collects the scores of candidate answers from each module to come to an agreement. In later fusion, expected answers are given an extra score if they are in the recommendation list which is created with properties of detected objects. Empirically, each fusion technique provides complementary gains from paragraph caption information to improve VQA model performance, overall achieving significant improvements over a strong baseline VQA model. We also present several ablation studies and attention visualizations.

2 Related Work

Visual Question Answering (VQA): VQA has been one of the most active areas among efforts to connect language and vision (Malinowski and Fritz, 2014; Tu et al., 2014). The recent success of deep neural networks, attention modules, and object plus salient region detection has made more effective approaches possible (Antol et al., 2015; Goyal et al., 2017; Lu et al., 2016; Fukui et al., 2016; Xu and Saenko, 2016; Yang et al., 2016; Zhu et al., 2016; Anderson et al., 2018).

Paragraph Image Captioning: Another thread of research which deals with combined visual and language problem is the translation of visual contents to natural language. The first approach to this included using a single-sentence image captioning model (Karpathy and Fei-Fei, 2015). However, this task is not able to accommodate the variety of aspects of a single image. Johnson et al. (2016) expanded single-sentence captioning to describe each object in an image via a dense captioning model. Recently, paragraph captioning models (Krause et al., 2017; Liang et al., 2017; Melas-Kyriazi et al., 2018) attempt to capture the many aspects in an image more coherently.

3 Models

The basic idea of our approach is to provide the VQA model with extra text information from paragraph captions and object properties (see Fig. 1).

3.1 Paragraph Captioning Model

Our paragraph captioning module is based on Melas-Kyriazi et al. (2018)'s work, which uses CIDEr (Vedantam et al., 2015) directly as a reward to train their model. They make the approach possible by employing self-critical sequence training (SCST) (Rennie et al., 2017). However, only employing RL training causes repeated sentences. As a solution, they apply n-gram repetition penalty to prevent the model from generating such duplicated sentences. We adopt their model and approach to generate paragraph captions.

3.2 VTQA Model

3.2.1 Features

Visual Features: We adopt the bottom-up and top-down VQA model from Anderson et al. (2018), which uses visual features from the salient areas in an image (bottom-up) and gives them weights using attention mechanism (top-down) with features from question encoding. Following Anderson et al. (2018), we also use Faster R-CNN (Ren et al., 2015) to get visual features $V \in \mathbb{R}^{O \times d}$, where O is #objects detected and d is the dimension of each visual feature of the objects.

Paragraph Captions: These provide diverse aspects of an image by describing the whole scene. We use GloVe (Pennington et al., 2014) for the word embeddings. The embedded words are sequentially fed into the encoder, for which we use GRU (Cho et al., 2014), to create a sentence representation, $s_i \in \mathbb{R}^d$: $s_i = \text{ENC}_{sent}(w_{0:T})$, where T is the number of words. The paragraph feature is a matrix which contains each sentence representation in each row, $P \in \mathbb{R}^{K \times d}$, where K is the number of sentences in a paragraph.

Object Property Sentences: The other text we use is from properties of detected objects in images (name and attribute), which can provide explicit information of the corresponding object to a VQA model. We create simple sentences like, "[object name] is [attributes]". We then obtain sentence representations by following the same process as what we do with the paragraph captions above. Each sentence vector is then attached to the corresponding visual feature, like 'name tag', to allow the model to identify objects in the image and their corresponding traits.

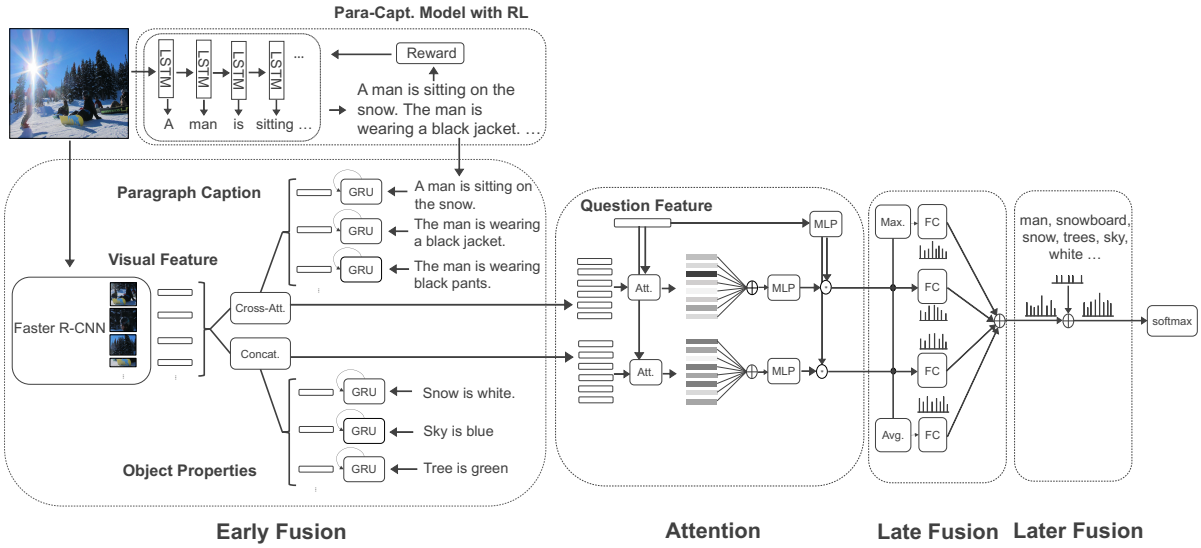

Figure 1: VTQA Architecture: Early, Late, and Later Fusion between the Vision and Paragraph Features.

3.2.2 Three Fusion Levels

Early Fusion: In the early fusion stage, visual features are fused with paragraph caption and object property features to extract relevant information. For visual and paragraph caption features, cross-attention is applied to get similarity between each component of visual features (objects) and a paragraph caption (sentences). We follow Seo et al. (2016)'s approach to compute the similarity matrix, $S \in \mathbb{R}^{O \times K}$. From the similarity matrix $V^p = \text{softmax}(S^T)V$ and the new paragraph representation, P^f is obtained by concatenating P and $P * V^p$: $P^f = [P; P * V^p]$, where $*$ is element-wise product operation. For visual feature and object property feature C, they are already aligned and the new visual feature V^f becomes $V^f = [V; V * C]$. Given the fused representations, the attention mechanism is applied over each row of the representations to weight more relevant features to the question.

$$a_i = w_a^T(\text{ReLU}(W_{sa}s_i^f) * \text{ReLU}(W_{qa}q)) \quad (1)$$

$$\alpha = \text{softmax}(a) \quad (2)$$

where, s_i^f is a row vector of new fused paragraph representation and q is the representation vector of a question which is encoded with GRU unit. w_a^T, W_{sa}, and W_{qa} are trainable weights. Given the attention weights, the weighted sum of each row vector, s_i^f leads to a final paragraph vector $p = \sum_{i=1}^{K} \alpha_i s_i^f$. The paragraph vector is fed to a nonlinear layer and combined with question vector by element-wise product.

$$p^q = \text{ReLU}(W_p p) * \text{ReLU}(W_q q) \quad (3)$$

$$L_p = \text{classifier}(p^q) \quad (4)$$

where W_p and W_q are trainable weights, and L_p contains the scores for each candidate answer. The same process is applied to the visual features to obtain $L_v = \text{classifier}(v^q)$.

Late Fusion: In late fusion, logits from each module are integrated into one vector. We adopt the approach of Wang et al. (2016). Instead of just adding the logits, we create two more vectors by max pooling and averaging those logits and add them to create a new logit $L_{new} = L_1 + L_2 + ... + L_n + ... + L_{max} + L_{avg}$, where L_n is nth logit, and L_{max} and L_{avg} are from max-pooling and averaging all other logits. The intuition of creating these logits is that they can play as extra voters so that the model can be more robust and powerful.

Answer Recommendation or 'Later Fusion': Salient regions of an image can draw people's attention and thus questions and answers are much more likely to be related to those areas. Objects often denote the most prominent locations of these salient areas. From this intuition, we introduce a way to directly connect the salient spots with candidate answers. We collect properties (name and attributes) of all detected objects and search over answers to figure out which answer can be extracted from the properties. Answers in this list of expected answers are given extra credit to enhance the chance to be selected. If logit L_{before} from the final layer contains scores of each answer, we want to raise the scores to logit L_{after} if the correspond-

ing answers are in the list l_c:

$$L_{\text{before}} = \{a_1, a_2, ..., a_n, ..\}$$
$$L_{\text{after}} = \{\hat{a}_1, \hat{a}_2, ..., \hat{a}_n, ..\} \quad (5)$$

$$\hat{a}_n = \begin{cases} a_n + c \cdot \text{std}(L_{before}) & \text{if } n \in l_c \\ a_n & \text{otherwise} \end{cases} \quad (6)$$

where the std$(\cdot)$ operation calculates the standard deviation of a vector and c is a tunable parameter. l_c is the list of the word indices of detected objects and their corresponding attributes. The indices of the objects and the attributes are converted to the indices of candidate answers.

4 Experimental Setup

Paragraph Caption: We use paragraph annotations of images from Visual Genome (Krishna et al., 2017) collected by Krause et al. (2017), since this dataset is the only dataset (to our knowledge) that annotates long-form paragraph image captions. We follow the dataset split of 14,575 / 2,487 / 2,489 (train / validation / test).

Visual Question Answering Pairs: We also use the VQA pairs dataset from Visual Genome so as to match it with the provided paragraph captions. We almost follow the same image dataset split as paragraph caption data, except that we do not include images that do not have their own question-answer pairs in the train and evaluation sets. The total number of candidate answers is 177,424. Because that number is too huge to train, we truncate the question-answer pairs whose answer's frequency are under 30, which give us a list of 3,453 answers. So, the final number of question-answering pairs are 171,648 / 29,759 / 29,490 (train / validation / test).

Training Details: Our hyperparameters are selected using validation set. The size of the visual feature of each object is set to 2048 and the dimension of the hidden layer of question encoder and caption encoder are 1024 and 2048 respectively. We use AdaMax (Kingma and Ba, 2014) for the optimizer and a learning rate of 0.002. We modulate the final credit, which is added to the final logit of the model, by multiplying a scalar value c (we tune this to 1.0).

5 Results, Ablations, and Analysis

VQA vs. VTQA As shown in Table 1, our VTQA model increases the accuracy by 1.92% from the baseline VQA model for which we employ Anderson et al. (2018)'s model and apply multi-modal factorized bilinear pooling (MFB) (Yu et al., 2017). This implies that our textual data helps improve VQA model performance by providing clues to answer questions. We run each model five times with different seeds and take the average value of them. For each of the five runs, our VTQA model performs significantly better ($p < 0.001$) than the VQA baseline model.

	Model	Test accuracy (%)
1	VQA baseline	44.68
2	VQA + MFB baseline	44.94
3	VTQA (EF+LF+AR)	46.86

Table 1: Our VTQA model significantly outperforms ($p < 0.001$) the strong baseline VQA model (we do not apply MFB to our VTQA model, since it does not work for the VTQA model).

	Model	Val accuracy (%)
1	VTQA + EF (base model)	45.41
2	VTQA + EF + LF	46.36
3	VTQA + EF + AR	46.95
4	VTQA + EF + LF + AR	47.60

Table 2: Our early (EF), late (LF), and later fusion (or Answer Recommendation AR) modules each improves the performance of our VTQA model.

Late Fusion and Later Fusion Ablations As shown in row 2 of Table 2, late fusion improves the model by 0.95%, indicating that visual and textual features complement each other. As shown in row 3 and 4 of Table 2, giving an extra score to the expected answers increases the accuracy by 1.54% from the base model (row 1) and by 1.24% from the result of late fusion (row 2), respectively. This could imply that salient parts (in our case, objects) can give direct cues for answering questions.[1]

Ground-Truth vs. Generated Paragraphs We manually investigate (300 examples) how many questions can be answered only from the ground-truth (GT) versus generated paragraph (GenP) captions. We also train a TextQA model (which uses cross-attention mechanism between question and caption) to evaluate the performance of the GT and GenP captions. As shown in Table 3, the GT captions can answer more questions correctly than GenP captions in TextQA model evaluation. Human evaluation with GT captions also shows better performance than with GenP captions as seen in Table 4. However, the results from the man-

[1]Object Properties: Appending the encoded object properties to visual features improves the accuracy by 0.15% (47.26 vs. 47.41). This implies that incorporating extra textual information into visual features could help a model better understand the visual features for performing the VQA task.

	Model	Val accuracy (%)
1	TextQA with GT	43.96
2	TextQA with GenP	42.07

Table 3: TextQA with GT model outperforms TextQA with GenP (we run each model five times with different seeds and average the scores. GT: Ground-Truth, GenP: Generated Paragraph).

	Human Eval.	Accuracy (%)
1	with GT	55.00
2	with GenP	42.67

Table 4: Human evaluation only with paragraph captions and questions of the validation dataset. Human evaluation with GT shows better performance than human evaluation with GenP.

ual investigation have around 12% gap between GT and generated captions, while the gap between the results from the TextQA model is relatively small (1.89%). This shows that paragraph captions can answer several VQA questions but our current model is not able to extract the extra information from the GT captions. This allows future work: (1) the TextQA/VTQA models should be improved to extract more information from the GT captions; (2) paragraph captioning models should also be improved to generate captions closer to the GT captions.[2]

Attention Analysis Finally, we also visualize the attention over each sentence of an input paragraph caption w.r.t. a question. As shown in Figure 2, a sentence which has a direct clue for a question get much higher weights than others. This explicit textual information helps a VQA model handle what might be hard to reason about only-visually, e.g., 'two (2) cows'. Please see Appendix A for more attention visualization examples.

6 Conclusion

We presented a VTQA model that combines visual and paragraph-captioning features to significantly improve visual question answering accuracy, via a model that performs early, late, and later fusion. While our model showed promising results, it still used a pre-trained paragraph captioning model to obtain the textual symbolic information. In future work, we are investigating whether the VTQA model can be jointly trained with the paragraph captioning model.

[2]We also ran our full VTQA model with the ground truth (GT) paragraph captions and got an accuracy value of 48.04% on the validation dataset (we ran the model five times with different seeds and average the scores), whereas the VTQA result from generated paragraph captions was 47.43%. This again implies that our current VTQA model is not able to extract all the information enough from GT paragraph captions for answering questions, and hence improving the model to better capture clues from GT captions is useful future work.

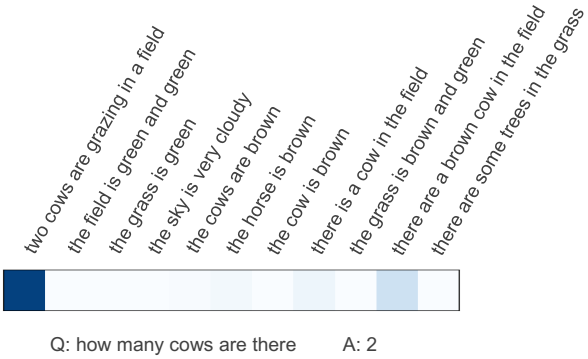

Figure 2: Attention Visualization for an example answered correctly by our model.

Acknowledgments

We thank the reviewers for their helpful comments. This work was supported by NSF Award #1840131, ARO-YIP Award #W911NF-18-1-0336, and faculty awards from Google, Facebook, Bloomberg, and Salesforce. The views, opinions, and/or findings contained in this article are those of the authors and should not be interpreted as representing the official views or policies, either expressed or implied, of the funding agency.

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 *International Conference on Computer Vision (ICCV)*.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734.

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. In

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 457–468.

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. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Justin Johnson, Andrej Karpathy, and Li Fei-Fei. 2016. Densecap: Fully convolutional localization networks for dense captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. 2017. A hierarchical approach for generating descriptive image paragraphs. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 3337–3345. IEEE.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.

Xiaodan Liang, Zhiting Hu, Hao Zhang, Chuang Gan, and Eric P Xing. 2017. Recurrent topic-transition gan for visual paragraph generation. *arXiv preprint arXiv:1703.07022*.

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

Mateusz Malinowski and Mario Fritz. 2014. A multi-world approach to question answering about real-world scenes based on uncertain input. In *Advances in neural information processing systems*, pages 1682–1690.

Luke Melas-Kyriazi, Alexander Rush, and George Han. 2018. Training for diversity in image paragraph captioning. *EMNLP*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7008–7024.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Kewei Tu, Meng Meng, Mun Wai Lee, Tae Eun Choe, and Song-Chun Zhu. 2014. Joint video and text parsing for understanding events and answering queries. *IEEE MultiMedia*, 21(2):42–70.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. 2016. Temporal segment networks: Towards good practices for deep action recognition. In *European Conference on Computer Vision*, pages 20–36. Springer.

Huijuan Xu and Kate Saenko. 2016. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *European Conference on Computer Vision*, pages 451–466. Springer.

Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. 2016. Stacked attention networks for image question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–29.

Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. 2017. Multi-modal factorized bilinear pooling with co-attention learning for visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 1821–1830.

Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. 2016. Visual7w: Grounded question answering in images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4995–5004.

Appendices

A Attention Visualization

As shown in Figure 3, paragraph captions contain direct or indirect clues for answering questions.

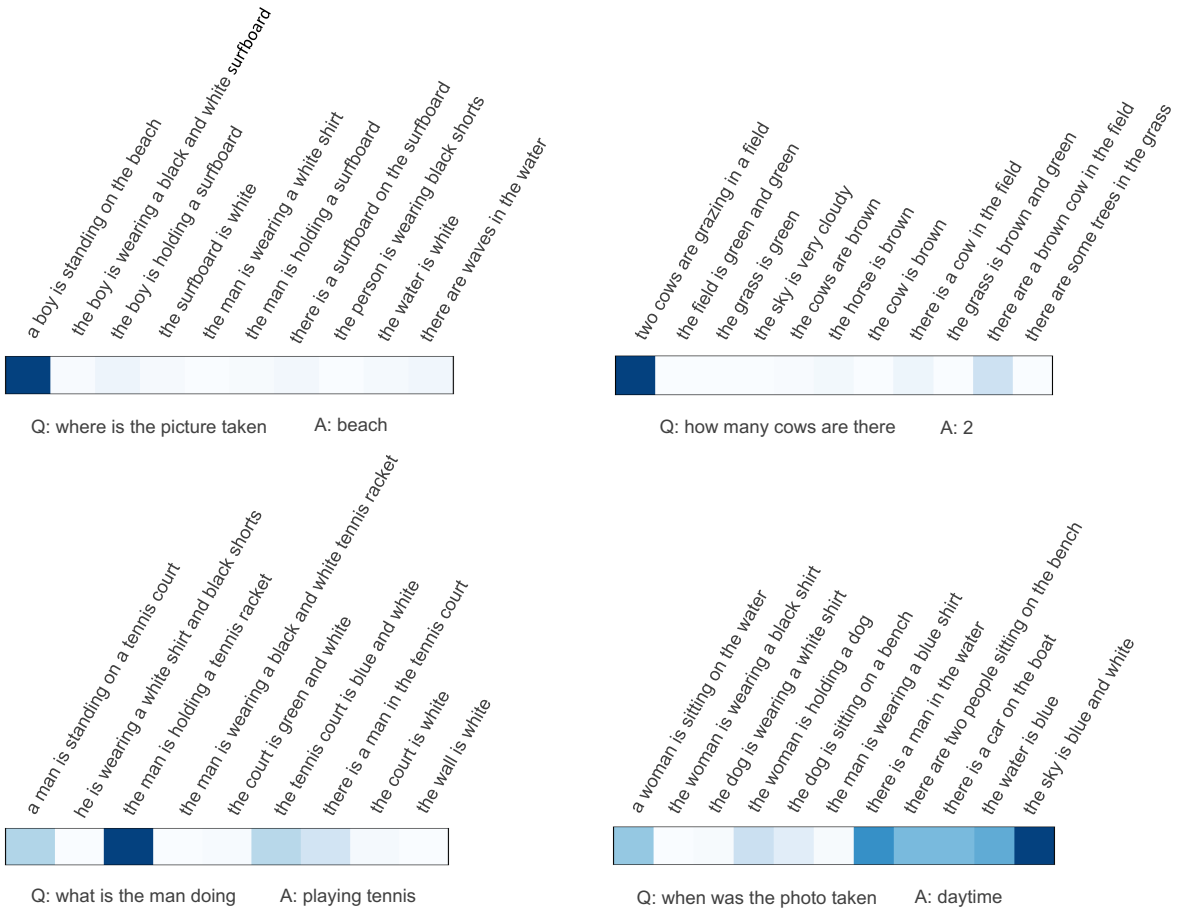

Figure 3: Attention Visualization: For all examples, our model answers correctly.

The upper left figure This is the case that a sentence in the paragraph caption can give obvious clue for answering the given question. By looking at the sentence "a boy is standing on the beach", this question can be answered correctly.

The upper right figure The sentence "two cows are grazing in a field " gives the correct answer "2" directly.

The bottom left figure There is no direct clue like "he is playing tennis", but the correct answer can be inferred by integrating the information from different sentences such as "the man is holding a tennis racket" and "a man is standing on a tennis court".

The bottom right figure This case seems tricky, but the answer can be inferred by associating the blue sky with daytime.