# Multi-grained Attention with Object-level Grounding for Visual Question Answering

**Pingping Huang, Jianhui Huang, Yuqing Guo, Min Qiao** and **Yong Zhu**
Baidu Inc., Beijing, China
{huangpingping,huangjianhui,guoyuqing,qiaomin,zhuyong}@baidu.com

## Abstract

Attention mechanisms are widely used in Visual Question Answering (VQA) to search for visual clues related to the question. Most approaches train attention models from a coarse-grained association between sentences and images, which tends to fail on small objects or uncommon concepts. To address this problem, this paper proposes a multi-grained attention method. It learns explicit word-object correspondence by two types of word-level attention complementary to the sentence-image association. Evaluated on the VQA benchmark, the multi-grained attention model achieves competitive performance with state-of-the-art models. And the visualized attention maps demonstrate that addition of object-level groundings leads to a better understanding of the images and locates the attended objects more precisely.

## 1 Introduction

Visual Question Answering (Antol et al., 2015; Goyal et al., 2017a) is a multi-modal task requiring to provide an answer to the question with reference to a given image. Most current VQA systems resort to deep neural networks and solve the problem by end-to-end learning. First the question and the image are encoded into semantic representations independently. Then the multi-modal features are fused into one unified representation for which the answer is predicted (Malinowski et al., 2015; Fukui et al., 2016; Anderson et al., 2018).

A key point to a successful VQA system is to discover the most relevant image regions to the question. This is commonly resolved by attention mechanisms, where a spatial attention distribution highlighting the visual focus is computed according to the similarity between the whole question and image regions (Xu et al., 2015; Yang et al., 2016; Lu et al., 2016). Although such coarse sentence-image alignment reports promising results in general, it sometimes fails to locate small objects or understand a complicated scenario. For the example in Figure 1, the question is "*What is the man wearing around his face*". Human has no difficulty in finding the visual clue on the people's faces, and accordingly provide the correct answer "*glasses*". However, by visualizing the attention map of a state-of-the-art VQA model, we find that the attention is mistakenly focused on the men's body rather than their faces.


Figure 1: An example of VQA and the attention maps produced by a state-of-the-art model and our model.

In order to identify related objects more precisely, this paper proposes a multi-grained attention mechanism that involves object-level grounding complementary to the sentence-image association. Specifically, a matching model is trained on an object-detection dataset to learn explicit correspondence between the content words in the question and their visual counterparts. And the labels of the detected objects are considered and their similarity with the questions are computed. Besides, a more sophisticated language model is adopted for better representation of the question. Finally the three types of word-object, word-label and sentence-image attention are accumulated to enhance the performance.

The contributions of this paper are twofold. First, this paper proposes a multi-grained attention mechanism integrating two types of object features that were not previously used in VQA atten-



*Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3595–3600
Florence, Italy, July 28 - August 2, 2019. 

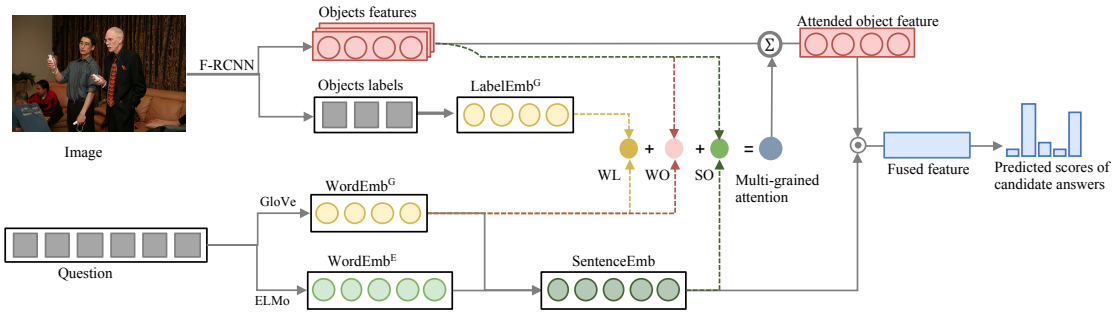


Figure 2: The architecture of our proposed model. The enhanced modules are illustrated in dot lines.

tion approaches. Second, the deep contextualized word representation ELMo (Peters et al., 2018) is firstly adopted in the VQA task to facilitate a better question encoding.

## 2 Proposed Model

The flowchart of the proposed model is illustrated in Figure 2. We start from the bottom-up top-down (up-down) model (Teney et al., 2017; Anderson et al., 2018), which is the winning entry to the 2017 VQA challenge. Then this model is enhanced with two types of object-level groundings to explore fine-grained information, and a more sophisticated language model for better question representation.

### 2.1 Image Features

We adopt the object-detection-based approach to represent the input image. Specifically, following Anderson et al. (2018), a state-of-the-art object detection model Faster R-CNN (Ren et al., 2015) with ResNet-101 (He et al., 2016) as its backbone is trained on the Visual Genome (VG) (Krishna et al., 2016) dataset. Then the trained model[1] is applied to identify instances of objects with bounding boxes belonging to certain categories. The target categories of this detection model contain 1600 objects and 400 attributes.

For each input image, the top-$K$ objects with the highest confidence scores are selected to represent the image. For each object, the output of ResNet's pool-flat-5 layer is used as its visual feature, which is a 2048-dimensional vector $\boldsymbol{v}_k$. Besides, the label of each object's category $\boldsymbol{c}_k$ is also kept as a visually grounded evidence. $\boldsymbol{c}_k$ is a $N$-dimensional one-hot vector, where $N$ is the vocabulary size. Then the input image is represented by both its object features $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_K] \in \mathbb{R}^{2048\times K}$ and object labels $\boldsymbol{C} = [\boldsymbol{c}_1, \boldsymbol{c}_2, ..., \boldsymbol{c}_K] \in \mathbb{R}^{N\times K}$.

### 2.2 Text Features

In our model, text features include token features and sentence features for the question, which are respectively used for fine-grained and coarse-grained attention computation.

**Word Features** Let $\boldsymbol{Q} = [\boldsymbol{q}_1, ..., \boldsymbol{q}_T] \in \mathbb{R}^{N\times T}$ denote the one-hot representation for the input question tokens, where $T$ is the question length, and $N$ is the vocabulary size. Then each token $q_t$ is turned into two word embeddings: GloVe (Pennington et al., 2014) $\boldsymbol{x}_t^G = \boldsymbol{q}_t\boldsymbol{E}^G \in \mathbb{R}^{D_1}$, and ELMo $\boldsymbol{x}_t^E = ELMo(\boldsymbol{q}_t) \in \mathbb{R}^{D_2}$. $D_1$ and $D_2$ are the dimensions of GloVe embedding and ELMo embedding respectively. $\boldsymbol{E}^G$ is the GloVe matrix pre-trained on the Wikipedia & Gigaword[2]. The ELMo embedding is dynamically computed by a $L$-layer bi-LSTM language model (Hochreiter and Schmidhuber, 1997). We use the publicly available pre-trained ELMo model[3] to get the contextualized embeddings.

**Sentence Features** The above two sets of token embeddings are then concatenated $\boldsymbol{x}_t = [\boldsymbol{x}_t^G; \boldsymbol{x}_t^E] \in \mathbb{R}^{D_1+D_2}$, and fed into a GRU (Cho et al., 2014) to encode the question sentence. The final hidden state of the GRU *i.e.*, $\boldsymbol{h}_T \in \mathbb{R}^{D_3}$ is taken as sentence feature, where $D_3$ is the hidden state size for GRU.

### 2.3 Multi-grained Attentions

**Word-Label Matching Attention (WL)** Object category labels are high-level semantic representation compared to visual pixels, and have proven to

[1]The model is available at https://github.com/peteanderson80/bottom-up-attention

[2]http://nlp.stanford.edu/projects/glove/

[3]https://github.com/allenai/allennlp

be useful for both visual tasks like scene classification (Li et al., 2010) and multi-modal tasks like image caption and VQA (Wu et al., 2018).

For VQA task, we observed that the semantic similarity between the object category labels and the words in the question helps to locate the referred objects. For the input image in Figure 1, Faster-RCNN detected objects with labels of "*man*", "*head*". Some labels are exactly the same as or are semantically close to the words in the question "*What is the man wearing around his face?*". Therefore, we compute the WL attention vector, that indicates how much weight we should give to each of the $K$ objects in the image, in terms of the semantic similarity between the category labels of the objects and the words in the question. For the $k$-th object with label $\boldsymbol{c}_k$ we encode it into GloVe embedding[4] $\boldsymbol{l}_k^G = \boldsymbol{c}_k \boldsymbol{E}^G$, and compute its attention score by measuring its similarity to the question GloVe embedding as follows:

$$
\begin{aligned}
s^{WL}(\boldsymbol{X}^G, \boldsymbol{l}_k^G) &= \arg\max_t \cos(\boldsymbol{x}_t^G, \boldsymbol{l}_k^G) \\
\boldsymbol{a}^{WL}(\boldsymbol{X}^G, \boldsymbol{L}^G) &= softmax\left(s^{WL}(\boldsymbol{X}^G, \boldsymbol{l}_k^G)\right)
\end{aligned} \tag{1}
$$

where $\boldsymbol{X}^G = \left[\boldsymbol{x}_1^G, ..., \boldsymbol{x}_T^G\right] \in \mathbb{R}^{D_1 \times T}$ is the GloVe embeddings for the question tokens. $\boldsymbol{L}^G = \left[\boldsymbol{l}_1^G, ..., \boldsymbol{l}_k^G\right] \in \mathbb{R}^{D_1 \times K}$ is the GloVe embeddings for the objects labels. $\boldsymbol{a}^{WL} \in \mathbb{R}^K$ is the WL attention vector. In contrast to Anderson et al. (2018) that only use objects' visual features without the labels, and unlike Wu et al. (2018) that discard the visual features once the labels are generated, we utilize both category labels and the visual features to enhance the fine-grained attention with object-level grounding.

**Word-Object Matching Attention (WO)** A word-object matching module is exploited to directly evaluate how likely a question word matches a visual object. The pairwise training structure of the module is shown in Figure 3. The training set is constructed on the VG object detection data. Let $(\boldsymbol{c}, b)$ be a positive sample consisting of the annotated object bounding-box $b$ with category label $\boldsymbol{c}$, then a negative sample $(\boldsymbol{c}, \bar{b})$ is constructed by randomly replacing $b$ with the object $\bar{b}$ in the same image, if $\bar{b}$ is not labelled with

[4]The reason why GloVe embedding alone is used instead of ELMo for object labels, is that object labels have no context sentence to derive the context-sensitive ELMo embeddings.

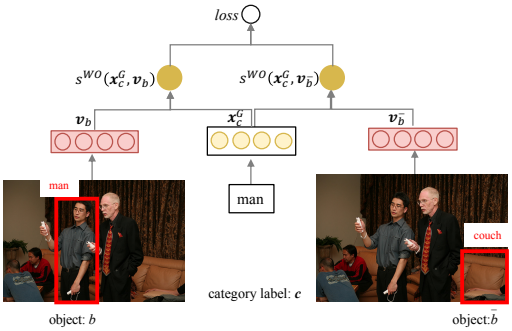


Figure 3: Label-object matching module trained on VG object annotation data.

$\boldsymbol{c}$. Then, each sample $(\boldsymbol{c}, b)$ is represented as feature vectors $(\boldsymbol{x}_c^G, \boldsymbol{v}_b)$, where $\boldsymbol{x}_c^G$is the GloVe embedding of $\boldsymbol{c}$, and $\boldsymbol{v}_b$ is extracted with the same Faster R-CNN model as described in section 2.1. At last, a margin-based pairwise ranking loss is used to train the model:

$$
\begin{aligned}
s^{WO}(\boldsymbol{x}_c^G, \boldsymbol{v}_b) &= \sigma\left(\boldsymbol{W}_s\left[f(\boldsymbol{W}_c \boldsymbol{x}_c^G) \circ f(\boldsymbol{W}_v \boldsymbol{v}_b)\right]\right) \\
loss &= \max\left\{0, \lambda - s^{WO}(\boldsymbol{x}_c^G, \boldsymbol{v}_b) + s^{WO}(\boldsymbol{x}_c^G, \boldsymbol{v}_{\bar{b}})\right\}
\end{aligned} \tag{2}
$$

where $f$ is ReLU and $\sigma$ is sigmoid activation function, $\circ$ means element wise multiplication. $\boldsymbol{W}_c, \boldsymbol{W}_v, \boldsymbol{W}_s$ are weight parameters[5]. And the margin is set $\lambda = 0.5$.

After $s^{WO}$ is pre-trained, we forwardly select at most $B$ noun tokens in the question and compute the WO attention $\boldsymbol{a}^{WO}(\boldsymbol{X}, \boldsymbol{V})$ over the $K$ objects as follows:

$$
\boldsymbol{a}^{WO}(\boldsymbol{X}^G, \boldsymbol{V}) = softmax\left(\sum_{b=1}^{B} s^{WO}(\boldsymbol{x}_b^G, \boldsymbol{v}_k)\right) \tag{3}
$$

where the parameters of $s^{WO}$ are fine-tuned in down-streaming VQA task.

**Sentence-Object Attention (SO)** Following previous methods of sentence-level question guided visual attention, we also use the global semantic of the whole sentence to guide the focus on relevant objects. Taking sentence feature $\boldsymbol{h}_T$ and objects features $\boldsymbol{V}$ as input, SO attention vector $\boldsymbol{a}^{SO}$ is computed as follows:

$$
\begin{aligned}
s^{SO}(\boldsymbol{h}_T, \boldsymbol{v}_k) &= \sigma\left(\boldsymbol{W}_j\left[f(\boldsymbol{W}_v \boldsymbol{v}_k) \circ f(\boldsymbol{W}_t \boldsymbol{h}_T)\right]\right) \\
\boldsymbol{a}^{SO}(\boldsymbol{h}_T, \boldsymbol{V}) &= softmax\left(s^{SO}(\boldsymbol{h}_T, \boldsymbol{v}_k)\right)
\end{aligned} \tag{4}
$$

where $f$ is ReLU, $\sigma$ is sigmoid activation function, and $\boldsymbol{W}_j, \boldsymbol{W}_v, \boldsymbol{W}_t$ are weight parameters.

[5]All bias terms are omitted hereafter for simplicity

| Method | test-dev | | | |
|---|---|---|---|---|
| | **All** | Yes/no | Numbers | Other |
| Up-down | 65.32 | 81.82 | 44.21 | 56.05 |
| **Our Model** | **67.41** | **83.60** | **47.02** | **58.24** |

| Method | test-std | | | |
|---|---|---|---|---|
| | **All** | Yes/no | Numbers | Other |
| Prior | 25.98 | 61.20 | 0.36 | 1.17 |
| Language-only | 44.26 | 67.01 | 31.55 | 27.37 |
| d-LSTM-n-I | 54.22 | 73.46 | 35.18 | 41.83 |
| MCB | 62.27 | 78.82 | 38.28 | 53.36 |
| Up-down | 65.67 | 82.20 | 43.90 | 56.26 |
| **Our Model** | **67.73** | **83.88** | **46.60** | **58.50** |

Table 1: Result comparison on VQA v2 dataset. Results of Prior, Language-only, d-SLTM-n-I, MCB are reported in Goyal et al. (2017a). Result of up-down model is reported in Teney et al. (2017).

### 2.4 Multi-modal Fusion and Answer Prediction

The above three attentions are summed together for the final attention vector. Then we get the weighted visual feature vector $\boldsymbol{v}^a \in \mathbb{R}^{2048}$ for the image:

$$\begin{aligned} \boldsymbol{a} &= \boldsymbol{a}^{WL} + \boldsymbol{a}^{WO} + \boldsymbol{a}^{SO} \\ \boldsymbol{v}^a &= \sum_{k=1}^{K} a_k \boldsymbol{v}_k \end{aligned} \tag{5}$$

Then the question feature $\boldsymbol{h}_T$and the attended visual feature $\boldsymbol{v}^a$ are transformed into the same dimension and fused together with element-wise multiplication, to get the joint representation vector $\boldsymbol{r} \in \mathbb{R}^{D_4}$.

$$\boldsymbol{r} = f(\boldsymbol{W}_{rt}\boldsymbol{h}_T) \circ f(\boldsymbol{W}_{rv}\boldsymbol{v}^a) \tag{6}$$

where $f$ is ReLU, $\boldsymbol{W}_{rt}, \boldsymbol{W}_{rv}$ are weight parameters. Following Teney et al. (2017), we treat VQA task as a classification problem, and use the binary cross-entropy loss to take multiple marked answers into consideration:

$$\begin{aligned} \hat{s} &= \sigma\left(f(\boldsymbol{W}_a \boldsymbol{r})\right) \\ loss &= \sum_{a=1}^{A} s_a log(\hat{s_a}) - (1 - s_a) log(\hat{s}_a) \end{aligned} \tag{7}$$

where $\hat{s} \in \mathbb{R}^A$ is the predicted score over all $A$ answer candidates, $s_a$ is the target accuracy score[6].

## 3 Experiments and Analysis

### 3.1 Settings

Experiments are conducted on VQA v2 dataset (Goyal et al., 2017b). Questions are trimmed to a maximum of $T = 14$ words. We set the number of detected boxes to $K = 36$, and set the dimension of GloVe embeddings and ELMo embeddings to $D_1 = 300$ and $D_2 = 1024$, respectively. The GRU hidden size for question sentence is $D_3 = 1024$, and the joint representation $r$ is of dimension $D_4 = 2048$. Noun tokens count is set as fixed $B = 3$ with padding[7]. Candidate answers are restricted to the correct answers in the training set that appear more than a threshold, which results in a number of $A = 3129$ answer candidates. Adamax optimizer (Kingma and Ba, 2014) is used with initial learning rate of 0.002, and we use a learning rate decay schedule that reduces the learning rate by a factor of 0.1 every 3 epochs after 8 epochs. The batch size is 512.

[6]accuracy = $\min(\frac{\#\text{humans that provided that answer}}{3}, 1)$, i.e. an answer is accurate if at least 3 markers provided the answer.

| **Model** | **All** | Yes/no | Numbers | Other |
|---|---|---|---|---|
| Up-down | 63.15 | 80.07 | 42.87 | 55.81 |
| + WL | 64.29 | 81.75 | 44.34 | 56.29 |
| + WO | 64.24 | 82.00 | 43.69 | 56.18 |
| + ELMO | 64.15 | 81.86 | 44.11 | 55.98 |

Table 2: Model analysis results. Models were trained on *train* and evaluated on *val* set.

### 3.2 Comparisons with the State-of-the-arts

Table 1 shows the result comparison with the baseline up-down and other methods in single model setting. Our model outperforms these previous results, improving the up-down model from 65.32 to 67.41 on *test-dev*, and from 65.67 to 67.73 on *test-std*. This superior performance can be seen in all the answer types, especially for the most difficult ones *Numbers*, where our model gains significant +2.81/+2.70 improvement on the *test-dev*/*test-std*.

### 3.3 Model Analysis

To understand the effects of different components, the performance by adding one certain proposed component to the baseline is reported in Table 2. Adding our proposed two branches of fine-grained WL and WO attentions significantly improves the baseline performance. The result also verifies that ELMo embeddings combined with GloVe embeddings provide more sophisticated text representations, thus improves the overall performance.

### 3.4 Study on Attention Maps

To validate the effectiveness of the enhanced attention mechanism, we visualize the attentions and

[7]Though $B$ is set as a fixed value during the whole process, it can be variable with trivial modifications for the WO attention computation.

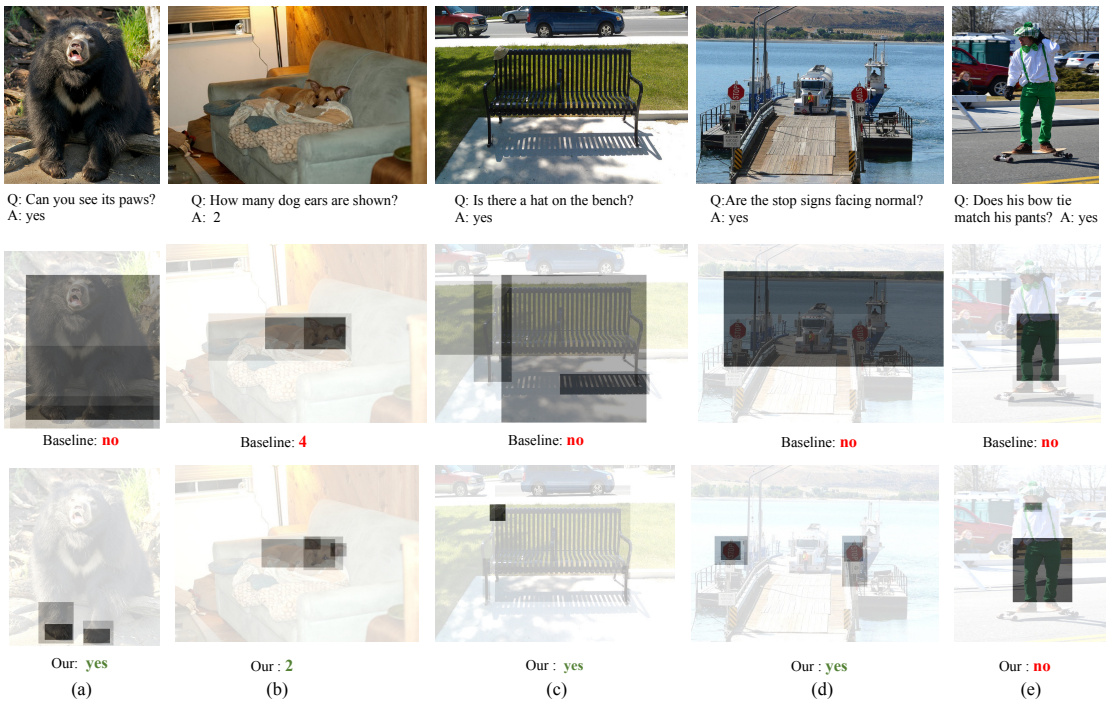


Figure 4: Attention map examples (only top-5 salient regions are shown here).

compare them versus those of the up-down model. As Figure 4 shows, the addition of object-level groundings leads to a better understanding of the images and locates the attended objects more precisely. For example, in Figure 4(a), for question "*Can you see its paws?*", the attention generated by our method is focused on the "*paws*", while the baseline does not focus on the key regions as accurate as we do. In Figure 4(b), for the *Numbers* type question "*How many dog ears are shown?*", our model gives the strongest attention on the "*ear*" part of the dog, while the baseline model attends to the whole dog body. For small object clues, our model shows more advantage. As shown in the examples in Figure 4(c), Figure 4(d).

We also notice cases where though the final answer is wrong, our model generates appropriate attention maps. As shown in Figure 4(e), for *Yes/no* question "*Does his bow tie match his pants?*", our model correctly finds "*tie*" and "*pants*" object regions, but we suspect that the model does not understand the meaning of "*match*".

A mean opinion score (MOS) test to quantitatively compare our attention mechanism with the baseline model is also performed. Specifically, we randomly select 100 cases and generate their attention maps. Then, we asked subjects to rate a score from 0 (bad quality), 0.5 (medium quality) and 1 (excellent quality) to these attention maps. The distribution of MOS ratings are summarized in Figure 5. The mean scores of our model 0.8125 wins a large margin over the baseline model 0.7315, indicating that the attention maps generated by our attention mechanism are preferred by human.

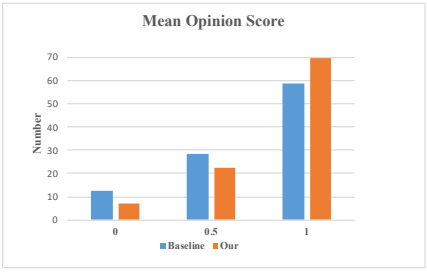


Figure 5: The distribution of Mean Opinion Score.

## 4 Conclusion

This paper proposes a multi-grained attention mechanism. It involves both word-object grounding and sentence-image association to capture different degrees of granularity and interpretability of the images. Visualizations of object-level attention show a clear improvement in the ability of the model to attend to small details in complicated scenes.

## 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086.

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, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoderdecoder for statistical machine translation. In *arXiv preprint arXiv:1406.1078*.

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

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017a. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*, volume 1, page 3.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017b. 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)*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

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

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. 2016. Visual genome: Connecting language and vision using crowdsourced dense image annotations.

Li-Jia Li, Hao Su, Yongwhan Lim, and Li Fei-Fei. 2010. Objects as attributes for scene classification. In *European Conference on Computer Vision*, pages 57–69. Springer.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. *CoRR*, abs/1606.00061.

Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. 2015. Ask your neurons: A neural-based approach to answering questions about images. In *In Proceedings of the IEEE International Conference on Computer Vision*, pages 1–9.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations.

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.

Damien Teney, Peter Anderson, Xiaodong He, and Anton van den Hengel. 2017. Tips and tricks for visual question answering: Learnings from the 2017 challenge. *arXiv preprint arXiv:1708.02711*.

Qi Wu, Chunhua Shen, Peng Wang, Anthony Dick, and Anton van den Hengel. 2018. Image captioning and visual question answering based on attributes and external knowledge. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1367–1381.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057.

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.