Stacked Self-Attention Networks for Visual Question Answering

Qiang Sun, Yanwei Fu
Academy for Engineering & Technology, Fudan University
School of Data Science, Fudan University
Fudan-Xinzailing joint research centre for big data
ZheJiang Xin ZaiLing Technology Co. LTD
Shanghai, China
sunqiang85@gmail.com,yanweifu@fudan.edu.cn

ABSTRACT

Given a photograph, the task of Visual Question Answering (VQA) requires joint image and language understanding to answer a question. It is challenging in effectively extracting the visual representation of images, and efficiently embedding the textual sentences of questions. To address these challenges, we propose a VQA model that utilizes the stacked self-attention for visual understanding, and the BERT-based question embedding model. Particularly, the stacked self-attention mechanism proposed enables the model to not only focus on a simple object but also the relations between objects. Furthermore, the BERT model is learned in an end-to-end manner to better embed the question sentences. Our model is validated on the well-known VQA v2.0 dataset, and achieves the state-of-the-art results.

CCS CONCEPTS

• **Computing methodologies → Scene understanding**.

KEYWORDS

visual question answering, scene understanding, language understanding

ACM Reference Format:
Qiang Sun, Yanwei Fu. 2019. Stacked Self-Attention Networks for Visual Question Answering. In *International Conference on Multimedia Retrieval (ICMR '19), June 10–13, 2019, Ottawa, ON, Canada.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3323873.3325044

1 INTRODUCTION

Visual Question Answering (VQA) [2] is a task that, given an image and question pair, the VQA model predicts an answer. The performance is evaluated by comparing the generated answer to the human annotated answers. It is an AI-complete challenge that is similar to a Turing test. Especially, the VQA task has to combine the techniques from different communities, including computer vision, natural language processing and knowledge reasoning.

In VQA, it is desirable and natural for the researchers to ask and validate whether their proposed AI systems have truly understood the visual scene and question meanings. In the visual part, the VQA model should effectively recognize the objects of images and identify the relations between objects [18]. In the textual part, the VQA model should be able to abstract the meanings of questions that are expressed by natural language in the questions [7]. The most difficult part is that the model should extract the visual information by the guidance of question, then use the knowledge reasoning and common senses for answering [17].

Extensive previous efforts have been made in addressing the VQA tasks. Object-level visual features extracted by Faster R-CNN [13] are very informative for VQA [1]. The Stacked Attention Network (SAN) [21] allows the multi-step reasoning for VQA, by making the query question vector evolve as processing. During the procedure, the visual features as a value set in attention mechanism stay unchanged. Thus, the attention mechanism in SAN mostly focuses on the initial visual features. This actually limits the reasoning and expressive power of SAN. In contrast, our attention mechanism further applies the self-attention to the evolved features that extracted in the previous step. The relations between objects can also be inferred by the self-attention.

We also introduce a novel way of efficiently embedding the question sentences by learning a BERT-based language model. Particularly, the new language model – Bidirectional Encoder Representation from Transformer (BERT) [4] obtained the state-of-the-art results on 11 NLP tasks. Among these tasks, the reading comprehension task on the Stanford Question Answering (SQuAD) dataset is similar to VQA. Inspired by the efficiency of BERT model, we learn a BERT-based question embedding model for VQA. To the best of our knowledge, there is no previous effort in successfully utilizing the BERT model in the VQA task.

Formally, this paper proposes a model that uses the stacked self-attention for both evolving visual features, and learning a BERT-based question embedding model. Critically, inspired by the human ability in recognizing different regions or objects gradually by the guide of first glimpse, our model keeps the "first glimpse" as a fixed query vector, and learns the evolved visual features by the attention of query vector, as illustrated in Fig. 1. The evolved visual features generated by self-attention layers are accumulated for the final answer prediction. Rather than directly using the Recurrent Neural Networks (RNNs) based models in previous works [1, 2, 6, 21], the BERT-based language model is learned to embed the question sentences. Our final model is trained in an end-to-end manner, in order to effectively combine the advantages of both stacked self-attention, and BERT-based question embedding. Experimental

ICMR '19, June 10–13, 2019, Ottawa, ON, Canada

ACM ISBN 978-1-4503-6765-3/19/06…$15.00
https://doi.org/10.1145/3323873.3325044

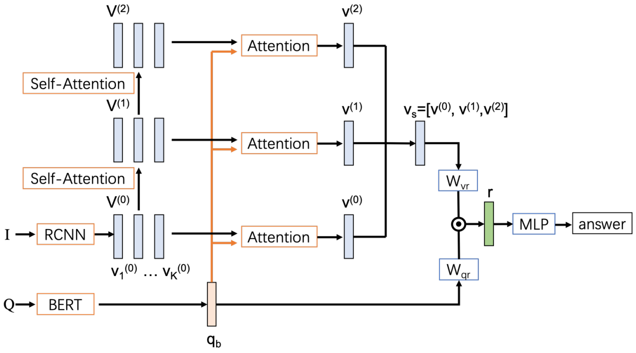

Figure 1: Overview of stacked self-attention networks.

results evaluated on VQA v2.0 dataset demonstrate the efficacy of our proposed model.

To sum up, we list the contributions. Firstly, we propose a novel stacked self-attention network to not only understand a single object but also the relations between objects. Secondly, for the first time, the BERT-based question embedding model is learned to address the VQA task. Finally, our proposed model is trained in an end-to-end way, which can effectively integrate the knowledge from both stacked self-attention visual features and BERT-based language knowledge.

2 RELATED WORK

There has been significant recent interest in addressing the problems of VQA. A huge amount of efforts have been made to improve the VQA model in different aspects, including feature extraction, feature fusion, attention mechanism, and external knowledge reasoning [19]. The visual features have been considered from coarse granularity to fine granularity, such as global image-level [23], grid region-level [7], and object-level [1]. The potential features of question embedding can come from the simple bag-of-words [23], Convolutional Neural Networks (CNNs) [11], and RNNs, (*e.g.*, LSTM [2, 25] and GRU [1]). Furthermore, the feature fusion is explored from the simple element operation [23] to MLB [8], MCB [5] and MUTAN [3]. Inspired by the recent success of the attention mechanism in image caption [20], various attention mechanisms have been applied in VQA task, such as question-guided attention on visual features [1, 7, 14], co-attention [10] and stacked attention [21]. External knowledge based model [17] is introduced to answer the question that needs complex reasoning or common sense.

The attention mechanism [16] uses a query vector to extract related information from a set of key-value pairs by similarity. For the question-guided attention mechanism [1, 7, 14] in VQA, the question embedding acts like a query vector, and the visual features serve as both the key and value vectors. The results are consistent with the intuition that question-guided attention works better with object-level visual features. Similarly, in the image-guided attention mechanism, image feature can be used as a query vector, and the question embedding vectors in the different steps serve as both the key and value vectors. Thus the image-guided attention can be viewed as extracting related words from the question sentence. In co-attention [10], the image-guided attention mechanism is a supplementary to the question-guided attention mechanism. In dual attention mechanism [12], the query vector is fused both by the image and question vectors, then the fused feature is further used to extract related information from both the visual and question features. Instead of treating attention as a single step, the models in SAN [21] , DAN [12] and CoR [18] compute the attention by multiple steps. We are aware that the self-attention used in transformer [16] has achieved remarkable success in natural language processing. By adopting the deep bidirectional transformer, the BERT [4] model can beat the other competitors in language understanding. Remarkably, the self-attention mechanism has demonstrated to be very effective in extracting more representative features in the computer vision task, *e.g.*, scoring sports video [22].

3 METHOD

This section describes the framework and pipeline of our approach. Fig. 1 shows that our approach is composed of four steps: feature extraction, attention, feature fusion, and classification. Particularly, In Sec. 3.1, we extract the initial visual features and question embedding. In Sec. 3.2, we use the stacked self-attention mechanism to extract the evolved visual features. Then we further employ the question-guided attention mechanism to extract the related visual features from the initial and evolved visual features. In Sec.3.3, we fuse the visual features and question embedding into a fusion space and predict the answer by a MLP classifier.

3.1 Feature Extraction

The object-level visual features are extracted from image by Faster R-CNN. For a fair comparison to [1], we use their provided visual features. For each image $\mathbf{I}$, a set of K object-level visual features is denoted as in Eq. (1):

$$\mathbf{V}^{(0)} = \mathrm{RCNN}\left(\mathbf{I}\right) \tag{1}$$

where $\mathbf{V}^{(0)} = \left[\mathbf{v}_1^{(0)}, \ldots, \mathbf{v}_K^{(0)}\right]^T \in \mathbb{R}^{K \times D_v}$ represents the set of top K object-level visual features, and D_v represents the dimension of visual feature.

The question feature is embedded by the BERT-based language model as shown in Fig. 2. We adopt the BERT [4] model, which is pre-trained on Wikipedia and BookCorpus, then futher fine-tune the BERT model on the training question sentences in the VQA dataset.

We feed the question into the BERT to get the hidden states $\mathbf{H}$ for all layers. Then we average the last four hidden layers to get the final question feature $\mathbf{q}_b$. The padding tokens are excluded from averaging.

$$\mathbf{H} = \mathrm{BERT}\left(\mathbf{Q}, \mathit{attention\_mask}\right) \tag{2}$$

$$\mathbf{q}_b = \mathrm{Avg}\left(\left[\mathbf{H}_{-1}; \mathbf{H}_{-2}; \mathbf{H}_{-3}; \mathbf{H}_{-4}\right]\right) \tag{3}$$

where $Q \in \mathbb{R}^{L_q}$ represents the token index sequence of question sentence. L_q is the max length of the token index sequence. The $\mathit{attention\_mask} \in \mathbb{R}^{L_q}$ indicates the corresponding element in token index sequence is a token or padding. $H \in \mathbb{R}^{L_b \times L_q \times D_b}$ represents the output of BERT model. L_b is the total layer number of BERT model. D_b denotes the dimension of each hidden layer in

BERT. $\mathbf{q}_b \in \mathbb{R}^{D_q}$ is the final question embedding. D_q denotes the dimension of the final question feature, which is four times as D_b.

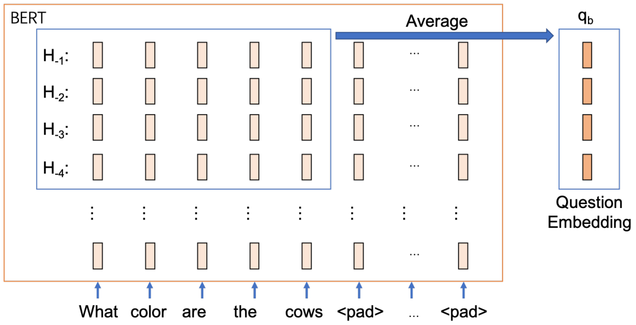

Figure 2: Question Embedding by BERT. We average the last 4 hidden layers as question embedding.

3.2 Attention mechanism

Both the self-attention and question-guided attention mechanisms are employed in our model. In the self-attention mechanism, the evolved visual features that generated by self-attention can serve as a summary of multi-step relations. In the question-guided attention mechanism, both the initial and evolved visual features are attended by question embedding.

In self-attention mechanism, the object-level visual feature set $\mathbf{V}^{(0)}$ has two roles: one is the query vector set, *i.e.*, "first glimpse"; the other is the 0-step evolved visual features, *i.e.*, the initial visual feature set.

Similarly, the the n-step evolved visual features are denoted as $\mathbf{V}^{(n)} = \left[\mathbf{v}_1^{(n)}, \ldots, \mathbf{v}_K^{(n)}\right]$. The feature $\mathbf{v}_k^{(n)}$ can be viewed as a summary of the n-step relations of the k-th initial object. Intuitively, the 0-step evolved visual feature $\mathbf{v}_k^{(0)}$ is the k-th object itself; the 1-step evolved visual feature $\mathbf{v}_k^{(1)}$ is a summary of direct relations of the k-th object.

Suppose the n-step evolved visual features are obtained, the $n+1$ step evolved visual features can be generated as Eq. (4)~(6).

$$\mathbf{s}_k^{(n)} = \mathbf{V}^{(n)}\mathbf{v}_k^{(0)} \tag{4}$$

$$\mathbf{a}_k^{(n)} = \text{softmax}\left(\mathbf{s}_k^{(n)}\right) \tag{5}$$

$$\mathbf{v}_k^{(n+1)} = \mathbf{a}_k^{(n)}\mathbf{V}^{(n)} \tag{6}$$

Firstly, Eq. (4) calculates the similarity $\mathbf{s}_k^{(n)} \in \mathbb{R}^K$ between the k-th initial object feature $\mathbf{v}_k^{(0)}$ and the the n-step visual features $\mathbf{V}^{(n)}$. Secondly, the attention vector $\mathbf{a}_k^{(n)}$ is obtained by the softmax function in Eq. (5). Finally, the next step visual feature $\mathbf{v}_k^{(n+1)}$ is obtained by a weighted averaging in Eq. (6). Thus the $n+1$ step self-attention visual feature set is formed as $\mathbf{V}^{(n+1)} = \left[\mathbf{v}_1^{(n+1)}, \ldots, \mathbf{v}_K^{(n+1)}\right]$.

In the question-guided attention mechanism, the question related n-step visual feature $\mathbf{v}^{(n)}$ is calcuated as in Eq. (7)~(11).

$$\mathbf{e}_k^{(n)} = \text{ReLU}\left(\mathbf{v}_k^{(n)}\mathbf{W}_{ve}\right) \odot \text{ReLU}\left(\mathbf{q}_b\mathbf{W}_{qe}\right) \tag{7}$$

$$\mathbf{z}_k^{(n)} = \mathbf{e}_k^{(n)}\mathbf{W}_{ez} \tag{8}$$

$$\mathbf{z}^{(n)} = \left[\mathbf{z}_1^{(n)}, \mathbf{z}_2^{(n)}, \ldots, \mathbf{z}_K^{(n)}\right] \tag{9}$$

$$\mathbf{w}^{(n)} = \text{softmax}\left(\mathbf{z}^{(n)}\right) \tag{10}$$

$$\mathbf{v}^{(n)} = \mathbf{w}^{(n)}\mathbf{V}^{(n)} \tag{11}$$

where Eq. (7) projects both the visual features and question embedding into a common latent feature space by $\mathbf{W}_{ve} \in \mathbb{R}^{D_v \times D_e}$ and $\mathbf{W}_{qe} \in \mathbb{R}^{D_q \times D_e}$. D_e denotes the dimension of common latent feature space. Then element-wise multiplication is applied to obtain $\mathbf{e}_k^{(n)}$. Eq. (8) further linearly projects $\mathbf{e}_k^{(n)}$ into a scalar $\mathbf{z}_k^{(n)}$ by $\mathbf{W}_{ez} \in \mathbb{R}^{D_e \times 1}$. $\mathbf{z}_k^{(n)}$ denotes the similarity of $\mathbf{v}_k^{(n)}$ and $\mathbf{q}_b$ when they are projected into the common latent feature space in attention. In Eq. (10), the similarity is normalized by the softmax function to obtain the weight vector $\mathbf{w}^{(n)}$. Finally, the question related n-step visual feature $\mathbf{v}^{(n)}$ is generated by a weighted average.

3.3 Feature fusion and classification

The output visual features of question-guided attention are concatenated and fused into a MLP classifier to predict the answers as in Eq. (12)~(14).

$$\mathbf{v}_s = \left[\mathbf{v}^{(0)}, \mathbf{v}^{(1)}, \ldots, \mathbf{v}^{(L)}\right] \tag{12}$$

$$\mathbf{r} = \text{ReLU}\left(\mathbf{v}_s\mathbf{W}_{vr}\right) \odot \text{ReLU}\left(\mathbf{q}_b\mathbf{W}_{qr}\right) \tag{13}$$

$$\mathbf{y} = \text{softmax}\left(\text{ReLU}\left(\mathbf{r}\mathbf{W}_{rh}\right)\mathbf{W}_{hy}\right) \tag{14}$$

where $\mathbf{v}_s$ can be viewed as the summary of visual objects and relations related to the question. L denotes the number of total stacked self-attention layers. Eq. (13) fuses the visual features and question embedding to produce the representation $\mathbf{r}$ by $\mathbf{W}_{qr} \in \mathbb{R}^{D_q \times D_r}$ and $\mathbf{W}_{vr} \in \mathbb{R}^{D_v L \times D_r}$. The classifier is a MLP with one hidden Layer, where $\mathbf{W}_{rh} \in \mathbb{R}^{D_r \times D_h}$ and $\mathbf{W}_{hy} \in \mathbb{R}^{D_h \times D_y}$. D_y is the size of candidate answer set. Therefore, the final output $\mathbf{y}$ is the possibility of each answer for the given image and question pair.

4 EXPERIMENT

4.1 Experimental Setup

We set up our parameters of model as follows. In feature extraction, the initial object visual set size K is set to 36. The dimension of visual feature D_v is set to 2048. The max length of question L_q is set to 14. The dimension of question embedding D_q is set to 3072. For attention, the number of stacked self-attention layers L is set to 2. The the common latent feature space dimension D_e is set to 1024. For feature fusion, the fusion feature dimension D_r is set to 3072. We train our networks by adamax optimizer with a learning rate of $5e^{-5}$, gradient clipping at 0.25, and batch-size at 32.

	Test-dev				Test-std			
Models	Yes/No	Number	Other	Overall	Yes/No	Number	Other	Overall
MCB [5]	-	-	-	-	78.82	38.28	53.36	62.27
MF-SIG-VG [24]	81.29	42.99	55.55	64.73	-	-	-	-
bottom-up [15]	81.82	**44.21**	56.05	65.32	82.20	**43.90**	56.26	65.67
our full model	**82.81**	42.60	**56.58**	**65.80**	**83.24**	42.01	**56.78**	**66.14**

Table 1: Results on the test-dev and test-std splits of VQA v2.0 dataset.

4.2 Dataset

We evaluate our model on the VQA v2.0 dataset [6], which contains 204k images from COCO [9]. The images are split into 3 folders: 83K for training, 40K for validation, and 81K for test. And the complete dataset contains 443K training, 214K validation, and 453K test image and question pairs. For each pair, there is a complementary pair with the same question but different image and ground-truth answer, which aims to eliminate the language bias. It is a challenging open-ended task with questions varied from "what", "where" to even "why". In evaluation, for each image and question pair, the model predicts the answer. The predicted answer accuracy is evaluated by comparing to 10 human labeled answers as shown in Eq. (15).

$$Acc(ans) = \min\{\frac{\#humans\ that\ said\ ans}{3}, 1\} \qquad (15)$$

4.3 Results and Analysis

In Tab. 1, we evaluate our stacked self-attention model in comparison to other competitors on the test-dev and test-std splits of VQA 2.0 dataset. The results show that our model is improved by 0.48% on test-dev, 0.47% on test-std overall comparing to the bottom-up model [15] . The bottom-up model is the winner of VQA 2.0 Challenge 2017. It is composed of Faster R-CNN, GRU, question guided attention, and gated tanh layers. Thus it is a suitable baseline for comparison. The results demonstrate the efficacy of our model that can beat the baselines.

Clearly, we notice that our model is more competitive on the "Yes/No" and "Other" questions. We argue that the "Yes/No" and "Other" questions actually require more reasoning ability in relations. This implies the advance of our model.

We also note that the performance of our model is worse than that of bottom-up model on "Number" questions. This makes sense, since the "Number" questions are about counting distinct objects. But the relation information extracted by stacked self-attention mechanism, intrinsically is not necessarily useful in addressing the counting task.

4.4 Ablation Study

To further reveal the insight of our model, we conduct two ablation experiments on the test-dev split in VQA v2.0 dataset. Our model is composed of two components: the stacked self-attention layers and BERT-based language model. To quantify the contribution of the each component, we build and retrain 2 variants: 1) "w/o Self-Attention": the model merely uses the question-guided attention, without self-attention layers. 2) "w/o BERT": the model employs the GRU for question sentence embedding instead of using the BERT-based language model.

Models	Yes/No	Number	Other	Overall
our full model	**82.81**	42.60	**56.58**	**65.80**
w/o BERT	78.49	40.67	53.46	62.32
w/o Self-Attention	82.31	**43.03**	56.49	65.60

Table 2: Ablation study on the VQA v2.0 test-dev dataset. "w/o BERT" stands for the model using GRU instead of BERT component; "w/o Self-Attention" stands for the model without self-attention mechanism.

Tab.2 shows that the result of our full model outperforms "w/o BERT" model on all the three question types: "Yes/No" by 4.32%, "Number" by 1.93%, "Other" by 3.12%. This may be caused by the fact that BERT model can provide the VQA model with a better question embedding than that of GRU, since the BERT model utilizes the bidirectional context in training, and it also can solve the long-range dependency problem.

Tab.2 also shows that the performance of our full model surpasses the "w/o Self-Attention" model with a slight margin of 0.20% overall. Specifically, our full model outperforms the "w/o Self-Attention" model on "Yes/No" and "Other" questions that more rely on relation information. Meanwhile, the self-attention mechanism decreases the performance on "Number" questions that do not need the relation information when counting.

5 CONCLUSION

This paper proposes a model based on the stacked self-attention and BERT language embedding for visual question answering. The stacked self-attention mechanism generates the new visual features which can be taken as a summary of relations of objects. BERT-based language model captures more information from the question sentence. Our model achieves the state-of-the art on VQA v2.0 dataset. The ablation study validates the two components of our model: stacked self-attention and BERT language model. As the future work, the proposed framework can be combined with a more sophisticated feature fusion mechanism to further improve the performance on VQA. The stacked self-attention mechanism could be applied to the other tasks that involving language and vision, such as image captioning.

REFERENCES

[1] 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*.

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

ICCV.

[3] Hedi Ben-Younes, Rémi Cadene, Matthieu Cord, and Nicolas Thome. 2017. Mutan: Multimodal tucker fusion for visual question answering. In *ICCV*.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[5] 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. *arXiv preprint arXiv:1606.01847* (2016).

[6] 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 *CVPR*.

[7] Vahid Kazemi and Ali Elqursh. 2017. Show, Ask, Attend, and Answer: A Strong Baseline For Visual Question Answering. *arXiv preprint arXiv:1704.03162* (2017).

[8] Jin-Hwa Kim, Kyoung Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. 2017. Hadamard Product for Low-rank Bilinear Pooling. In *ICLR*.

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

[10] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. In *NIPS*.

[11] Lin Ma, Zhengdong Lu, and Hang Li. 2016. Learning to Answer Questions from Image Using Convolutional Neural Network.. In *AAAI*.

[12] Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. 2017. Dual attention networks for multimodal reasoning and matching. In *ICCV*.

[13] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *NIPS*.

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

[15] 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* (2017).

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

[17] Peng Wang, Qi Wu, Chunhua Shen, Anthony Dick, and Anton Van Den Henge. 2017. Explicit knowledge-based reasoning for visual question answering. In *IJCAI*.

[18] Chenfei Wu, Jinlai Liu, Xiaojie Wang, and Xuan Dong. 2018. Chain of Reasoning for Visual Question Answering. In *NIPS*.

[19] Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2017. Visual question answering: A survey of methods and datasets. *CVIU* (2017).

[20] 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 *ICML*.

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

[22] Bing Zhang, Chengming Xu, Chang Mao Cheng, Yanwei Fu, Yu-Gang Jiang, and Xiangyang Xue. 2018. Learning to score and summarize figure skating sport videos. *arXiv preprint arXiv:1802.02774* (2018).

[23] Bolei Zhou, Yuandong Tian, Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. 2015. Simple baseline for visual question answering. *arXiv preprint arXiv:1512.02167* (2015).

[24] Chen Zhu, Yanpeng Zhao, Shuaiyi Huang, Kewei Tu, and Yi Ma. 2017. Structured attentions for visual question answering. In *ICCV*.

[25] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. 2016. Visual7w: Grounded question answering in images. In *CVPR*.