Error Analysis for Visual Question Answering

Artur Podtikhov[1], Makhmud Shaban[1], Alexey K. Kovalev[1,2], and Aleksandr I. Panov[2,3(✉)]

[1] National Research University Higher School of Economics, Moscow, Russia
[2] Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, Russia
[3] Moscow Institute of Physics and Technology, Moscow, Russia
panov.ai@mipt.ru

Abstract. In recent years, the task of visual question answering (VQA) at the intersection of computer vision and natural language processing is gaining interest in the scientific community. Even though modern systems achieve good results on standard datasets, these results are far from what is achieved in Computer Vision or Natural Language Processing separately, for example, in tasks of image classification or machine translation. One of the reasons for this phenomenon is the problem of modelling the interaction between modalities, which is partially solved by using the attention mechanism, as, for example, in the models used in this paper. Another reason lies in the statement of the problem itself. In addition to the problems inherited from CV and NLP, there are problems associated with the variety of situations shown in the picture and the possible questions for them. In this paper, we analyze errors for the state-of-the-art approaches and separate them into several classes: text recognition errors, answer structure, entity counting, type of the answer, and ambiguity of an answer. Text recognition errors occur when answering a question like "what is written in ..?" and associated with the representation of the image. Errors in the answer structure are associated with the reduction of the VQA to the classification task. Entity counting is a known weakness of current models. A typical situation of errors in the type of answer is when the model answers the "Yes/No" question in a different way. Errors from the ambiguity of an answer class occur when the model produces an answer that is correct in meaning but does not coincide with the formulation of the ground truth. Addressing these types of errors will lead to the overall improvement of VQA systems.

Keywords: Visual question answering · Error analysis · Faster R-CNN · Mask R-CNN · Attention · Bert

A. Podtikhov and M. Shaban—Equal contribution.

B. Kryzhanovsky et al. (Eds.): NEUROINFORMATICS 2020, SCI 925, pp. 283–292, 2021.
https://doi.org/10.1007/978-3-030-60577-3_34

1 Introduction

The Visual Question Answering (VQA) task was introduced in 2015 [1] by Agrawal et al. by formalizing the problem and providing the dataset for learning and evaluation. The task is to get an answer to a question on a given image. The image and the open-ended question related to it are passed as an input of the algorithm. The output should be a relevant answer to the question asked, which takes into account information collected from the input image and common-sense knowledge. The VQA task combines data processing tasks with visual and linguistic processing to answer questions regarding image data. Therefore, VQA can be called the multi-discipline Artificial Intelligence (AI) research problem, which consists of three different tasks: Computer Vision, Natural Language Processing, and Reasoning. In recent years, unimodal algorithms, that deal with some of these three parts have achieved great quality performance. However cross-modal tasks are more difficult because they require a simultaneous semantic understanding of two or more unimodal tasks, as well as the interaction between data of different modalities. The reasoning part of the problem is what makes VQA task unique, as it requires deeper understanding of the context than other cross-modal tasks such as image captioning. Machine learning for computer vision and natural language processing accelerates the advancement of artificial intelligence. Open-ended question answering entail a set of AI capabilities as it requires common-sense knowledge. That is why building robust algorithms for VQA that perform at near-human levels would be an important step towards solving the Artificial Intelligence problem.

Recent years have witnessed a growing academic interest in solving VQA task. That is why, VQA competitions, as well as an annual workshop dedicated to the latest developments in this direction, [1] are held annually starting from 2016. A dataset, consisting of 265,016 images, each of which contains an average of 5.6 relevant questions and 10 ground-truth answers, as well as 3 plausible (but likely incorrect) answers per question, was presented for this competition. Since annual VQA competition was launched, the development of the models can be traced to the winners and runners-up of the competitors.

There are number of approaches of solving VQA task, such as neural-symbolic [16,22], graph-based [13] or attention-based [2,24]. Despite the variety of possible approaches, an approach based on the attention mechanism is the most popular. The key idea of attention in artificial neural networks is to weigh object features based on a query, which can be a question, a word, or even the image itself. In this article, we review the most successful attention-based approaches and propose some modifications to improve model quality.

2 Base Models

Winner (Deep Modular Co-Attention Networks [23]) and runner-up (Bilinear Attention Networks [9]) solutions of VQA Challenge 2019 were chosen as base models.

2.1 Bilinear Attention Networks

Bilinear Attention Networks (BAN) [9] were introduced as an efficient way to utilize multimodal attention, maintaining the interaction between inputs of different modality. The architecture achieved state-of-the-art on VQA 2.0 dataset at the time of publication, and a runner-up solution for VQA 2019 Challenge was generated by a modified Bilinear Attention Network.

One of the core concepts of the proposed architecture is **low-rank bilinear pooling**, where the modified bilinear form that takes two inputs (for instance, question and word embeddings) and returns single output vector of fixed length. Replacing a single weight matrix of a bilinear form by multiplication of two smaller matrices significantly reduces the computational cost of the operation, and the pooling matrix provides a way to encode rich features without inflating the number of model parameters.

The aforementioned operation is used on the question and word embeddings to create a two-dimensional attention matrix A, where A_{ij} is the attention weight for an i-th word in question and j-th object in the image. Then another low-rank bilinear pooling is applied to question and word embeddings, considering attention weights provided by previously calculated bilinear attention matrix.

The proposed block can be effectively stacked with similar blocks if the output of the low-rank bilinear pooling is set to question embedding vector length. Residual connections between the start and the end of each block are added for signal preservation.

The image features are extracted using bottom-up attention Faster R-CNN [2], which is a slight modification of Faster R-CNN [18] with added "attribute of the object" loss term, which enforces rich image feature encoding. The object detection model is pretrained on Visual Genome [11] dataset.

The words of a question are encoded using GloVe [17] embeddings, and are fed into one-layer unidirectional GRU [4].

2.2 Deep Modular Co-Attention Networks

The next model is Deep Modular Co-Attention Networks (MCAN) [23]. It is the winner solution for VQA 2019 Challenge. This method is the development of the idea of attention, inspired by the Transformer [20] – translation model with multi-head attention.

The input image is represented by the set of region proposals features obtained in a bottom-up manner [2]. This method is based on ResNet [6] as backbone and Faster R-CNN [18] object detection model, trying to focus on the objects from Visual Genome dataset [11]. The input question is encoded using 300-D GloVe [17] embeddings pretrained on large scale corpus. The sequence of embedded words then passes through one-layer LSTM [7].

The authors came up with a new layer of modular co-attention between the regions of the image and the words of the question. They implemented two new attention units: Self-Attention and Guided-Attention. Self- Attention learns pairwise attention between paired input samples (between words in the question) and

Guided-Attention learns pairwise attention between paired input samples from different inputs (between words in question and image regions). Furthermore, they have invented deep encoder-decoder architecture to get a better representation of attention weights. Input features from image regions passed through 6 cascaded Self-Attention units, the last layer output is used in 6 cascaded Guided-Attention with inserts of Self-Attention under question features. The resulting feature matrices contain rich information about attention weights. Attention weights are obtained as two-layer MLP fed to a softmax layer. The attended feature is the dot product of attention weights and encoded features.

3 Modification of Base Models

To extract more information from experiments and address the lack of error analysis of current model and its further modifications, we implemented a more detailed evaluation, where the VQA accuracy is calculated for three categories of questions based on the type of an answer - a "Yes/No" question, a "number" question and other questions.

Image and question representations have a huge impact on the final performance. In this paper, we propose to improve the performance of base models by extracting more powerful visual and text features. Due to the inherent modularity of base models, modifications implemented in the form of replacing default modules for images and question representations with new ones.

The modified models in the following experiments were compared by official VQA metric.

Image Features. Faster R-CNN, which is used in the base model to obtain visual features of objects, is already outdated and does not show competitive quality in solving the Object Detection problem. Cascade Mask R-CNN [3] is the current state-of-the-art for the task. Its main difference from Faster R-CNN is that after predicting the areas and classes of objects, the signs are sent to another network in which the mask of the objects is predicted. Moreover, the architecture is improved further by stacking three ResNet models for usage as a backbone. Cascade Mask R-CNN, pre-trained on ImageNet is used as a visual features extractor in the following experiments.

Since Mask R-CNN struggles to find all the objects in the image, for example, the background, the Panoptic Feature Pyramid Network [10] was selected as the second option of the image features extraction model. In this model, the authors combine Mask R-CNN to select objects with a semantic segmentation model, to split the entire image into objects. It is proposed to add semantic segmentation to each FPN layer [14]. At the output of the model is an image, each pixel of which corresponds to a class. The model is pre-trained on ImageNet. Due to the combination of semantic segmentation with the selection of objects, it is impossible to obtain the attributes of objects in a standard way, therefore, ResNet-152 without the last layer is used to obtain 2048-D embedding. For each image segment found in the Panoptic model, all pixels except the pixels

included in the mask of the found object are used, then ResNet-152, pre-trained on ImageNet, is applied to the resulting image with one object.

During experiments we noticed that models without pre-training on VisualGenome dataset show worse results, that is why we tried one more object extraction model - Faster R-CNN with ResNetXt101 backbone [8] pre-trained on VisualGenome.

Question Features. BERT [5], a multi-layer bidirectional Transformer based on [21], is currently widely used in Natural Language Processing due to superior performance on almost every NLP task, compared to more dated alternatives, such as GRU or LSTM. The model consists of embedding layers, 12 Transformer blocks with a dimension of 768, and 12 Self-Attention Heads. Contextual BERT embeddings of words in question are used as input features of the final model. The maximum question length is increased from 14 to 36 since at the input the model requires words tokenized using WordPiece [19], in which unknown words are replaced with a combination of several tokens.

BERT, as well as its recent improvements, such as RoBERTa [15] and ALBERT [12], were used as a question feature extractor in the following experiments.

Counting Module. In the base model, the soft-attention mechanism is used, which limits the model's ability to count. That is why in [25] proposed a new Counting module. The input of this module is soft-attention weights and bounding boxes. Weights and boxes are converted into a graph representation, thus counting task is reduced to the task of finding duplicate edges. The output is a one-hot-encoded array that is sent to the linear layer with ReLU activation and sum with previous attention weight.

The results are presented in Table 1 and 2 of Sect. 4.

4 Results and Error Analysis

After a series of experiments, we concluded that "number" questions are the ones that the model has the most trouble with, even though this issue is already addressed by a counting module.

Table 1. BAN. Experiment results on VQA v2.0 validation split.

Object Detection	Backbone	Question features	Counter	Accuracy	Yes/No	Number	Other
Faster R-CNN (baseline)	ResNet101	GloVe + GRU	+	**66.60**	**79.00**	**49.56**	**55.73**
Faster R-CNN	ResNet101	BERT	+	61.85	74.70	42.85	51.54
Faster R-CNN	ResNet101	RoBERTa	+	62.90	75.10	44.41	52.86
Faster R-CNN	ResNet101	ALBERT	+	54.61	68.01	37.07	44.11
Faster R-CNN	ResNet101	ALBERT	–	55.20	68.38	37.80	44.77
Faster R-CNN	ResNeXt-101	ALBERT	–	54.64	67.32	36.22	44.91

Table 2. MCAN. Experiment results on VQA v2.0 validation split

Object Detection	Backbone	Question features	Counter	Accuracy	Yes/No	Number	Other
Faster R-CNN (baseline)	ResNet-101	GloVe + LSTM	–	67.38	83.81	48.78	59.88
Faster R-CNN	ResNet-101	BERT	–	67.85	83.89	48.63	60.82
Cascade Mask R-CNN	ResNetXt-152-32x8d	GloVe + LSTM	–	58.37	76.91	46.91	47.38
Cascade Mask R-CNN	ResNetXt-152-32x8d	BERT	–	58.42	76.24	47.31	47.80
Cascade Mask R-CNN	ResNetXt-152-32x8d	BERT	+	58.20	75.93	46.70	47.83
Panoptic FPN	ResNet-101	GloVe + LSTM	–	59.43	78.22	42.32	49.75
Panoptic FPN	ResNet-101	BERT	–	59.78	78.31	41.34	50.66
Faster R-CNN	ResNet-101	RoBERTa	–	66.78	82.86	47.49	59.24
Faster R-CNN	ResNet-101	ALBERT	–	68.29	84.47	**49.84**	60.95
Faster R-CNN	ResNeXt-101	ALBERT	–	**68.57**	**84.57**	49.78	**61.45**

Changing image features yields a positive effect only when the model is pre-trained on the VisualGenome dataset. Probably the reason is that VisualGenome has 3000 classes while ImageNet pre-trained models have only 300.

As far as modifications are concerned, replacing question feature extractor with the BERT-style language model does not lead to improvements in all cases. One possible explanation for this could be a lack of question feature extractor finetuning for the task, yet the solution for this problem is not trivial - attempts to train BAN while finetuning BERT-style backbones in the end-to-end fashion led to degraded performance compared to frozen models.

To gain useful insights from model predictions, we performed an error analysis on a set of baseline model predictions, where the predicted answer does not match any of the 10 assessors' answers provided in the dataset. We further refer to it as an "error set". The analysis was performed manually - given a subset of 1000 image-question pairs from an error set, each prediction was categorized as a member of one of 6 common mistakes categories, which were derived from a quantitative analysis of error sets:

- **Text recognition errors.** Such errors occur when the question requires extracting the text from the image. Since the current solution only extracts features of known objects, text regions are not taken into account.
- **Answer structure.** The main purpose of the model is to make classification on a known set of answers. That is why errors often occur in syntax structure of an answer such as statement of prepositions or pluralizing.
- **Type of the answer errors.** There are cases when the type of answer does not match the type of question. For example, in Fig. 1.c Appendix I, to a yes/no question corresponds 'answer 'answer.
- **Entity counting.** In Table 1 and Table 2 there is common trend in models. All current VQA models have difficulties with counting. Furthermore, this category includes questions about telling time on a clock.
- **Ambiguity of an answer.** This type of error is independent with model answers. Language is a complex structure and the answers of 10 people may not cover all possible correct answers. Errors of the model on precisely such

questions fell into this category, where it gives the correct answer, which does not coincide with the answers given by the assessors.
– **Wrong answer.** This category is for wrong predictions that do not fall into any of the 5 categories described above, and is most likely due to a genuine lack of reasoning.

As shown in Table 3, around the half of mistakes are made without a clear reason, a small percentage of ambiguity errors are troublesome to fix, but other types of errors can be fixed by modifications of the model. Text recognition failures can be fixed by making use of the latest advances in Optical Character Recognition by extracting the text from the image and using it to answer to the question. Answer structure errors appear due to a limited number of answers-classes used for solving the task. Consequently using a language model to generate final answer based on the output of the main model can be helpful. To solve the type-of-the-answer errors, a simple classifier on top of the model prediction completely fix the problem. The idea is to only look at the probabilities which correspond to the answers of the current category.

Table 3. Error analysis on validation set for MCAN and BAN models. Errors are divided into 6 categories.

Model	Text recognition	Answer structure	Type of the answer	Number	Ambiguity	Wrong answer
MCAN	12.6%	9.4%	4.8%	15.6%	8.4%	49.2%
BAN	9.01 %	4.1%	4.9%	14.8%	6.6%	49.2%

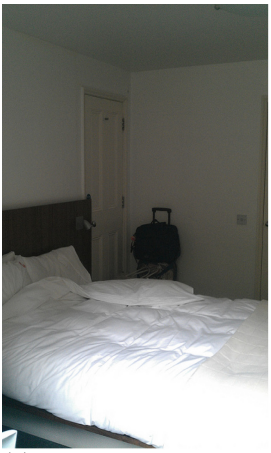
5 Conclusion

Visual Question Answering task requires understanding of both the given image and a question on that image. Recent advancements in Natural Language Processing and Computer Vision as separate disciplines help solve most of the problems that researchers can meet on their path. Nevertheless, questions that need sophisticated reasoning, such as counting of a specific type of objects, demand new techniques involving modality interaction.

The error analysis gave insight into possible shortcomings in current state-of-the-art VQA models, its struggles with entity counting, text recognition, answer structure and type. We hope that the analysis coupled with the proposed ways to solve these problems will help aspiring researchers to improve the existing VQA systems.

Acknowledgement. The reported study was supported by RFBR, research Projects No. 19–37-90164 and 18–29-22047.

6 Appendix I: Samples of Wrong Answers by Error Categories



(a) Sample of structure error.
Question: What is in the corner?
Model answer: bag;
Real answer: bags;

(b) Sample of classification error.
Question: What has two blue doors?
Model answer: yes;
Real answer: toilet;

(c) Sample of number error.
Question: How many blue benches are visible in the photo?
Model answer: 3;
Real answer: 2;

(d) Sample of text recognition.
Question: What is in the word written on the field?
Model answer: tennis court;
Real answer: polo or jp morgan;

(e) Sample of ambiguity error.
Question: Why is there a fire extinguisher in the kitchen?
Model answer: safety;
Real answer: prevent fire;

References

1. Agrawal, A., Lu, J., Antol, S., Mitchell, M., Zitnick, C.L., Batra, D., Parikh, D.: VQA: visual question answering. arXiv e-prints arXiv:1505.00468, May 2015
2. Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., Zhang, L.: Bottom-up and top-down attention for image captioning and visual question answering. arXiv e-prints arXiv:1707.07998, July 2017

3. Cai, Z., Vasconcelos, N.: Cascade R-CNN: high quality object detection and instance segmentation. arXiv e-prints arXiv:1906.09756, June 2019
4. Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv e-prints arXiv:1406.1078, June 2014
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. arXiv e-prints arXiv:1810.04805, October 2018
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. arXiv e-prints arXiv:1512.03385, December 2015
7. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–80 (1997). https://doi.org/10.1162/neco.1997.9.8.1735
8. Jiang, Y., Natarajan, V., Chen, X., Rohrbach, M., Batra, D., Parikh, D.: Pythia v0.1: the winning entry to the VQA challenge 2018. arXiv e-prints arXiv:1807.09956, July 2018
9. Kim, J.H., Jun, J., Zhang, B.T.: Bilinear attention networks. arXiv e-prints arXiv:1805.07932, May 2018
10. Kirillov, A., Girshick, R., He, K., Dollár, P.: Panoptic feature pyramid networks. arXiv e-prints arXiv:1901.02446, January 2019
11. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.J., Shamma, D.A., Bernstein, M.S., Li, F.F.: Visual genome: connecting language and vision using crowdsourced dense image annotations. arXiv e-prints arXiv:1602.07332, February 2016
12. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: ALBERT: a lite berT for self-supervised learning of language representations. arXiv e-prints arXiv:1909.11942, September 2019
13. Li, L., Gan, Z., Cheng, Y., Liu, J.: Relation-aware graph attention network for visual question answering. arXiv e-prints arXiv:1903.12314, March 2019
14. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. arXiv e-prints arXiv:1612.03144, December 2016
15. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: RoBERTa: a robustly optimized bert pretraining approach. arXiv e-prints arXiv:1907.11692, July 2019
16. Mao, J., Gan, C., Kohli, P., Tenenbaum, J.B., Wu, J.: The neuro-symbolic concept learner: interpreting scenes, words, and sentences from natural supervision. arXiv e-prints arXiv:1904.12584, April 2019
17. Pennington, J., Socher, R., Manning, C.D.: Glove: global vectors for word representation. In: EMNLP (2014)
18. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. arXiv e-prints arXiv:1506.01497, June 2015
19. Schuster, M., Nakajima, K.: Japanese and korean voice search. In: International Conference on Acoustics, Speech and Signal Processing, pp. 5149–5152 (2012)
20. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. arXiv e-prints arXiv:1706.03762, June 2017
21. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. ArXiv **abs/1706.03762** (2017)

22. Yi, K., Wu, J., Gan, C., Torralba, A., Kohli, P., Tenenbaum, J.B.: Neural-symbolic VQA: disentangling reasoning from vision and language understanding. arXiv e-prints arXiv:1810.02338, October 2018
23. Yu, Z., Yu, J., Cui, Y., Tao, D., Tian, Q.: Deep modular co-attention networks for visual question answering. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6274–6283 (2019)
24. Yu, Z., Yu, J., Xiang, C., Fan, J., Tao, D.: Beyond bilinear: generalized multimodal factorized high-order pooling for visual question answering. arXiv e-prints arXiv:1708.03619, August 2017
25. Zhang, Y., Hare, J., Prügel-Bennett, A.: Learning to count objects in natural images for visual question answering. arXiv e-prints arXiv:1802.05766, February 2018