Deep Learning for Visual Question Answering

Avi Singh
avisingh599@gmail.com

Abstract

This project deals with the problem of Visual Question Answering (VQA). We develop neural network-based models to answer open-ended questions that are grounded in images. We used the newly released VQA dataset (with about 750K questions) to carry out our experiments. Our model makes use of two popular neural network architecture: Convolutional Neural Nets (CNN) and Long Short Term Memory Networks (LSTM). We use state-of-the-art CNN features for encoding images, and word embeddings to encode the words. Our Bag-of-word + CNN model obtained an accuracy of 44.47%, while our CNN+LSTM model obtained an accuracy of 47.80% on the validation set of the VQA dataset. The code has been open sourced under the MIT License, and is the first open-source project to work with the VQA dataset.

1 Introduction

In recent years, there has been a lot of progress in AI problems at the intersection of Natural Language Processing (NLP) and Computer Vision. One problem that has garnered a lot of attention recently is Image Captioning[1, 2, 3]. However, the task is not well suited to track the progress of AI since image captions are nonspecific, and their automatic evaluation is still an open problem[4]. Another such problem is Visual Question Answering[5, 6]. In this task, the input is an image and a question based on the image, and the output is one or more words that answer the question. Open-ended question answering requires one to solve several lower-level problems like fine-grained recognition, object detection, activity recognition, common-sense reasoning, and knowledge-based reasoning[6]. Due to the specificity of the task, it can also be evaluated automatically, making it easier to track progress. These characteristics make it an ideal "AI-complete" problem, suitable for a Turing Test[7].

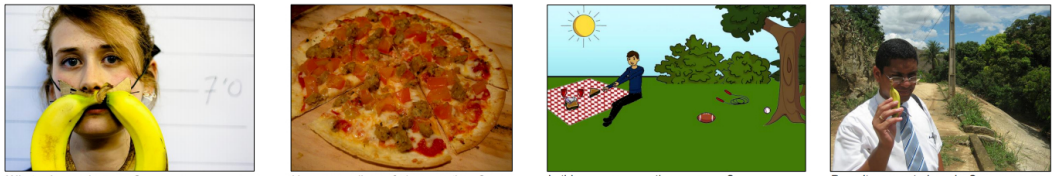


Figure 1: A teaser from the VQA dataset[6]

1.1 Motivation

Apart from helping us keep track of progress in AI, the problem also has more direct applications. Apps such as VizWiz[8] have been used by thousands of visually impaired people. Using the app, a person can take a picture, and ask a question about that picture. The image and the question get uploaded to a server, where a human answers the question. Due to the human being in the loop, the

Figure 2: Some outputs generated by the Neural Networks

users have to wait for some time before they get their answers. A reliable VQA system can automate this task, thus enabling instantaneous answers at no recurring costs.

1.2 Overview

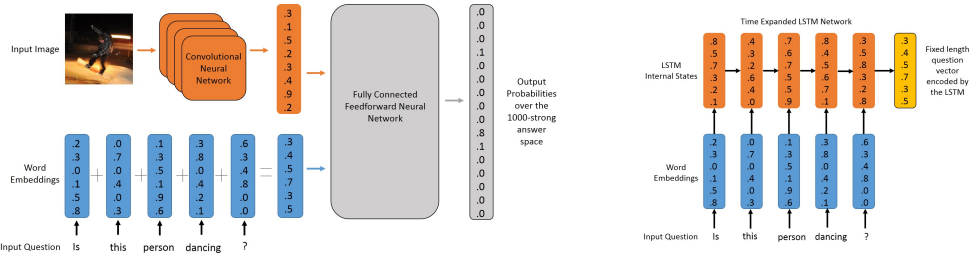
In this project, we provide end-to-end learnable models for answering questions about images. We have made use of the recently released, large-scale VQA dataset, which has about 750K questions over 250K images. In Section 2, we briefly describe all the work that has been done on the VQA problem so far, and present a survey of all image question answering datasets released so far. In Section 3, we introduce the major building blocks of our system. In Section 4, we present two end-to-end trainable models that solve the problem of VQA. In Section 5, we demonstrate the performance that these models achieve. We show several directions for future improvement in Section 6, and we conclude in Section 7.

2 Related Work

Almost all of the work in Visual Question Answering has been done in the last two years. Malinowski et al.[5] released the first image Q&A dataset, DAQUAR. They initially used a multi-word symbolic approach to the problem, but their subsequent work was based on a combination of Long Short Term Memory Networks (LSTM) and Convolutional Neural Networks (CNN)[9]. A research group at Baidu has reported results on a dataset with Chinese questions[10], automatic translations of which are also available, and results are reported on both. The most recent result is that of Ren et al.[11], and it shows that even simple Bag of Words models are hard to beat using LSTMs. They reported result on a dataset of automatically generated questions. Recently, the VQA[6] dataset was released, and it came with strong baselines based on CNN features combined LSTM models. Our model is different from them in the sense that it uses concatenation of image and question features, instead of fusing them by point wise multiplication. Also, we make use of word embeddings, whereas [6] uses one-hot feature vectors. Apart from the original paper that introduces the dataset, no results have been published on it. Industrial labs at Baidu and Facebook have released an extended abstract and a demo video respectively, but they are yet to publish their results.

A survey of publicly available datasets released on VQA has been presented here.

1. Visual Question Answering (VQA) dataset[6]: Based on images from the COCO dataset,it currently has 360K questions on 120K images. There are plans of releasing questions on the rest of the COCO images and an additional 50K abstract images. All the questions are human-generated, and were specifically designed to stump a ”smart robot”.Along with this dataset comes the VQA challenge, which is being organized for the first time this year.



(a) CNN + BOW Model

(b) CNN + LSTM Model

Figure 3: The architecture of the two models presented in the paper.

2. Visual Madlibs[12]: It contains fill-in-the-blank type questions along with standard question-answer pairs. It has 360K questions on 10K images from the COCO dataset. A lot of questions require high-level human cognition, such as describing what one feels on seeing an image.
3. Toronto COCO-QA Dataset[11]: Automatically generated questions from the captions of the MS COCO dataset. At 115K questions, it is smaller than the VQA dataset. Answers are all one word.
4. DAQUAR[5] - DAtaset for QUestion Answering on Real-world images: A much smaller dataset, with about 12K questions. This was one of the earliest datasets on image question and answering.

3 Preliminaries

This section briefly describes some of the main building blocks of the models that we present in Section 4.

- **Convolutional Neural Networks** In the last three years, CNNs have been established as the clear state-of-the art when it comes to visual recognition tasks like scene classification and object detection. Typical CNN architectures involve a sequence of convolutional layers followed by pooling layers, and this combination is repeated several times over, with Dropout layers added in the mix for regularization. For the purpose of this project, we have used the VGG[13] architecture which secured the first and second position in the localization and classification task respectively at ILSVRC 2014.
- **Long Short Term Memory Networks** Long-Short Term Memory (LSTM) networks are a variant of Recurrent Neural Netoworks that are capable of dealing with sequential data by "remembering" certain features of the history of inputs passed to them. LSTMs have given state-of-the-art performance when it comes to tasks like speech recognition and machine translation. Variants of these recurrent models have also been used in conversational models and question-answering systems.
- **Word Embeddings** Word Embeddings such as Google's Word2Vec and Stanford's GloVe are unsupervised techniques that convert words to dense continuous vectors of a fixed length (typically 300-1000). The words embedded using these techniques show interesting properties like similar words having high cosine similarity.

4 Models

We present our Neural Network based models in this section. The first one is based on a Convolutional Neural Network (see 3a), and the second is based on a combination of a Convolutional Neural Network and a Long Short Term Memory Network (see 3b).

Finite Answer Space: We work with a finite answer space by selecting the 1000 most frequent answers from the training set. The problem reduces to multi-class classification. These top-1000 answers still cover more than 80% of the training set, and we can thus expect the system to give a reasonable performance even with the finite answer space.

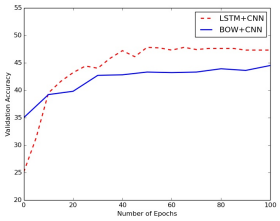
4.1 Convolutional Neural Network and Bag of Words

The image is passed through the VGG ConvNet[13], and the activations before the softmax layer are extracted, giving us a 4096-dimensional vector representing the image. The question is converted to a vector by summing up the word vectors corresponding to all the word present in the question. The two vectors (image and question) are concatenated, and passed through a Multi-Layer Perceptron with two fully connected layers and 50% dropout for regularization. A softmax layer is attached at the end, and it gives us a probability distribution over the entire answer space.

4.2 Convolutional Neural Network and Long Short Term Memory Network

The previous model ignores the order in which the words appear in the question, and there is a loss of information when summing up the word vectors. To capture the sequential nature of language data, we model the questions using LSTMs. Every word in the question is first converted to its embedding, and these embeddings are passed into the LSTM in a sequential fashion. The final output of the LSTM is used as the *question embedding*. This question embedding is concatenated with the 4096-dimensional image vector, and we then apply the same Multi-Layer Perceptron architecture that we used in the previous BOW model. The entire network is trained end-to-end, except the Convolutional Neural Net.

5 Experiments



Model	Accuracy
BOW + CNN	44.47%
LSTM - Language Only	42.51%
LSTM + CNN	47.80%

Figure 4: The LSTM outperforms the BOW model

All our experiments were conducted on the VQA dataset[6] with 240K training set questions for learning and 120K validation set questions for evaluating performance. There are ten ground truth answers available for every question in the VQA dataset, and we used only the most frequent answer for every question while training. We ran our experiments for 100 epochs, and the learning curves are shown in Figure 1.

5.1 Hyperparameters

The hyperparameters used for the two models are reported here:

- CNN+BOW Model: Two fully-connected hidden layers of with 1024 hidden units each. Dropout is 50%. The activation function used is tanh. A softmax layer is attached at the end.
- CNN + LSTM Model: There is one layer of LSTM with 512 hidden units. After concatenation with the output of the VGG Network, this is followed by two fully-connected hidden layers with 1024 hidden units and 50% dropout. Hard sigmoid function is used for inner activation in LSTM, while tanh is used for in both the outside activation in LSTM and the fully connected layers. A softmax layer is attached at the end.

5.2 Evaluation Methodology

We have used the same evaluation as has been outlined in the VQA challenge. For every answer produced by the neural network, it is matched against the ten ground truth answers provided by humans. If the answer generated by the neural net exactly matches against three of the the ground truth answer, then the answer generated is assumed to be correct.

5.3 Discussion

It is interesting to see that the system has a fairly high accuracy (more than 40%) when working with only language data. This shows that the system is capable of recognizing the type of questions (yes/no question, what color question etc.), and this information imposes strong priors on the answer that is to be generated.

5.4 Implementation Details

The code is written in Python using the Keras library with a Theano backend. This enabled it to take advantage of GPU hardware (whenever available). A speed-up of as much as 10X was observed when using a GPU instead of a CPU on mini-batches. The entire code for training and evaluating models has been released along with some pre-trained models on the VQA dataset. A demo script to work with arbitrary new questions is also available. It is released under the MIT license on Github[14].

Model	Nvidia GTX 760	Interl Core i7
BOW + CNN	140 seconds/epoch	900 seconds/epoch
LSTM + CNN	200 seconds/epoch	1900 seconds/epoch

Table 1: The GPU leads to upto 10x performance improvements with a batch size of 128

6 Future Work

Some possible directions of future work have been highlighted in this section.

1. **Visual Attention**: Currently, the system uses only high-level image information in the form of CNN features. Ideally, we would like to have an attention-based model which pays more attention to selected regions of the image, and the selection of these regions should be conditioned on the question. For example, a question asking about the color of a ball should pay more attention on the region containing the ball. This technique has been successfully applied to image captioning[15].
2. **Sequence to Sequence Models**: A major limitation of the current approach is that the answer space is finite. It is desirable to have a sequence-to-sequence model that can generate arbitrary answers over the entire English language. One possible technique to do this is to use LSTM-based encoders and decoders, as shown in [16] for the task of machine translation. Currently, our system uses an LSTM only for encoding the question, not for generating the answers.
3. **Visual Semantic Alignment**: Karpathy and Fei-Fei [1] have shown that is possible to semantically align images and their descriptions, as shown in Figure 5. This approach could also be applied to image-question pairs, and could possibly be combined with attention models in order to get fine-grained information from the image in context of the question.

7 Conclusion

This project demonstrates how we can use combinations of CNN and LSTM for answering questions about images. Experiments were conducted on the VQA dataset, and have shown promising results. The code has been open-sourced under the MIT License, and is the first open-source project to work with the VQA dataset. We have also indicated several possible future directions that can be taken to better model the problem.

Acknowledgement: We would like to thank Prof. Amitabha Mukherjee for his encouragement, guidance, and also for the resources provided in terms of the GPU and system storage space. Without his support, this project would not have been possible.

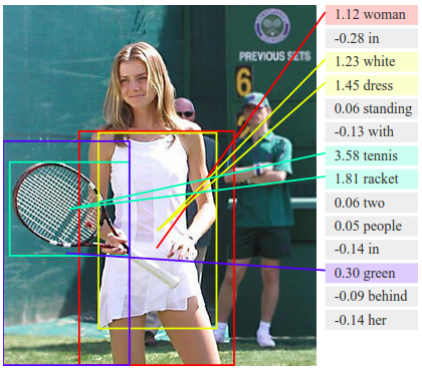


Figure 5: An example of image-sentence alignment generated by [1]

8 References

[1] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[2] Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[3] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[4] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[5] Mateusz Malinowski and Mario Fritz. A multi-world approach to question answering about real-world scenes based on uncertain input. *Neural Information Processing Systems (NIPS)*, 2014.

[6] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. *International Conference on Computer Vision (ICCV)*, 2015.

[7] Mateusz Malinowski and Mario Fritz. Towards a visual turing challenge. *NIPS Workshop*, 2014.

[8] Jeffrey P. Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C. Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samual White, and Tom Yeh. Vizwiz: Nearly real-time answers to visual questions. In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, pages 333–342, New York, NY, USA, 2010. ACM.

[9] Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. Ask your neurons: A neural-based approach to answering questions about images. *International Conference on Computer Vision (ICCV)*, 12/2015 2015.

[10] Haoyuan Gao, Junhua Mao, Jie Zhou, Zhiheng Huang, and Alan Yuille. Are you talking to a machine? dataset and methods for multilingual image question answering. *Neural Information Processing Systems (NIPS)*, 2015.

[11] Mengye Ren, Ryan Kiros, and Richard S. Zemel. Exploring models and data for image question answering. *Neural Information Processing Systems (NIPS)*, 2015.

[12] Licheng Yu, Eunbyung Park, Alexander C. Berg, and Tamara L. Berg. Visual Madlibs: Fill in the blank Image Generation and Question Answering. *arXiv preprint arXiv:1506.00278*, 2015.

[13] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[14] A. Singh. https://github.com/avisingh599/visual-qa. 2015.

[15] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*, 2015.

[16] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. *Neural Information Processing Systems (NIPS)*, abs/1409.3215, 2014.