Evaluating Modern Unimodal Baselines for Visual Question Answering

Andriamarolahy Rabetokotany
(arabetokotany@aimsammi.org)
African Master for Machine Intelligence (AMMI)
Rwanda

Supervised by: Pr. Stefan Lee

August 24, 2021
Submitted in Partial Fulfillment of a Masters II at AMMI

Abstract

Declaration

I, the undersigned, hereby declare that the work contained in this essay is my original work, and that any work done by others or by myself previously has been acknowledged and referenced accordingly.

Andriamarolahy Rabetokotany, August 24, 2021

Contents

1. Introduction

In recent years tasks that involve both vision and language problems have grabed the attention from Computer Vision (CV) and Natural Language Processing (NLP) communities. One such task which has become very popular is Visual Question Answering (VQA) [1] [2]. It is the task consists of building a system that can answer natural language questions about a given image. For example, given the image in Figure 1.1 and the question "How many horses are in this image?", a VQA system should be able to give the right answer which is "two".

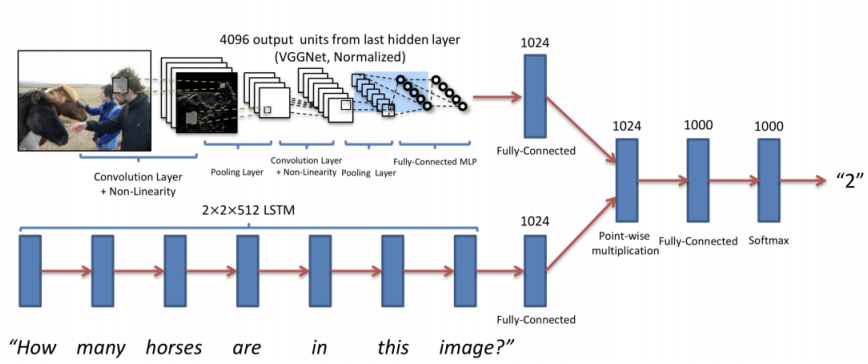

Figure 1.1: An example of VQA system [1]

This is by nature a multimodal task as it seems to require both language understanding and visual processing. For instance, in the above example the system must understand that this is a "how many" question, so the answer must be a number, and then detect and recognize a horse based on its appearence. To validate that hypothesis, dataset papers for these tasks often report single-modality baselines in order to know how well can a model learn to answer questions from just vision or from just language.

When the VQA dataset was released, Recurrent Neural Network (RNN) like Long Short Term Memory ($LSTM$) [5] were the dominant way language was processed so the language-only baselines are based on $LSTM$s. Since then, the field of NLP has seen many breakthroughs, especially the shift to transfer learning. Large-scale pretrained language models like Bidirectional Encoder Representations from Transformers ($BERT$) [8] are now available and can be reused and fine-tuned for smaller tasks, rather than training a new model from scratch. Accordingly $BERT$ is a deep learning based model that has given state-of-the-art results on a wide variety of NLP tasks from sentiment analysis to question answering.

The purpose of this project is to retrain a language-only baseline VQA model using a pre-trained $BERT$ model and evaluate how much improvements in VQA come from better language models.

This work is composed of four chapters with the following contents: chapter 2 provides a general literature survey about VQA task, chapter 3 gives some backgrounds study of $BERT$ model, chapter 4 describes the methods used for this project, chapter 5 discusses the experimental results and finally gives conclusion and some perspectives.

2. Literature survey

In their seminal paper of 2016, S. Antol et al. [1] have introduced the task of Visual Question Answering (VQA) with a large amount of dataset containing over $250K$ images, $760K$ questions, and around $10M$ answers in which different baseline models were benchmarked. Since then, additional datasets have been released and a large number of algorithms have been proposed. For example, Y. Goyal et al. released a more balanced version of the VQA dataset when they tried to aleviate the problem of language biases in the original dataset [2].

In general most of the existing methods consist of 1) extracting features from image and question then combine the obtained features to produce an answer and 2) framing VQA as a classification problem in such way the image and question features are the inputs to the classification system and each answer represents a class. Pre-trained CNNs model like $VGGNet$ [3] and $ResNet$ [4] are the most common image features extractor while there are a wide variety of question features extractor such as bag-of-words (BOW), long short term memory ($LSTM$) encoders [5], gated recurrent units (GRU) [6], and skip-thought vectors [7]. Thus the main difference between them is how they integrate the question and image features before feeding them to a linear classification or neural network. Several techniques ranging from trivial like concatenation, elementwise multiplication, or elementwise addition [10] to the more sophisticated like using approach that maps questions and images to vector representations in a common feature space [9] have been used to combine the question and image features.

Several researchers have highlighted the performance and limitation of the existing datasets and techniques. For instance C. Kanan et al. found that $ResNet$ outperforms $VGGNet$ or $GoogLeNet$ across multiple algorithms. [11] [12] stated that attention based mechanisms, modular and memory-augmented architectures improve model's performance. However existing datasets and models have major shortcoming: evaluating answers generated by algorithms is a challenging problem, and existing datasets suffer from biases and other problems [11]. Various approaches have been hypothesized to solve these issues for future research. [11] [13] [14] suggest to develop more sophisticated deep learning models as well as more challenging datasets for VQA. C. Kanan et al. [11] underline that evaluating different kinds of questions individually in a nuanced manner, rather than using naive accuracy alone, will significantly help. Q. Wu et al. [12] propose to scale up the inclusion of additional external knowledge from structured knowledge bases, as well as a continued exploration of the potential of NLP tools.

The focus of this study will be based on the suggestion of Q. Wu et al. to explore the potential of NLP tools by getting the power of $BERT$ inolve into the VQA workflow.

3. Background: BERT

3.1 Generality

The shift to transfer learning is one of the major breakthrough in the field of NLP. Instead of training new models from scratch pre-trained models like $ELMo$ [15] and OpenAI's GPT [16] are now available and can be fine-tuned on smaller task-specic datasets such as question answering and sentiment analysis. This approach results in great accuracy improvements compared to training from scratch. $BERT$ is one of the best among these techniques that has given state-of-the-art results on a wide variety NLP tasks [8].

3.2 The core idea behind BERT

The key difference of $BERT$ from other popular embedding models, such as word2vec, is that it is a context-based embedding model. This means it generates a representation of each word based on the other words in the sentence. For example, consider the two following sentences: *"I am learning python."* and *"I am scared of the big python."*. The word *python* which refers to a programming language will have the same representation as the word *python* which is a snake using the context-free models. In contrast, $BERT$ will understand the context and generate the representation of the word accordingly. So for the two example sentences, the word *python* will have different representations based on the context.

Moreover, $BERT$ has been pre-trained on Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) [17] tasks. These techniques help $BERT$ to understand respectively the context in a sentence and the relationship between two sentences.

Masked Language Modeling

Language Modeling is the task of predicting the next word given a sentence. In MLM a percentage of input tokens is masked at random and the model predicts only those masked tokens instead of predicting every next token. This approach is implemented in $BERT$ by randomly mask 15% of the tokens in the input sentence in which: 80% of the tokens are actually replaced with the token [MASK], 10% of the time tokens are replaced with a random token and 10% of the time tokens are left unchanged.

Next Sentence Prediction

Next sentence prediction task is a binary classification task in which, given a pair of sentences, it is predicted if the second sentence is the actual next sentence of the first sentence. A pre-trained model with this kind of understanding is helpful for tasks like question answering and Natural Language Inference (NLI).

3.3 BERT architecture and functionality

Architecture

$BERT$ builds on top of transformer [18]: a clever idea consists of encoding and decoding the input text to produce a prediction by learning contextual relationships between words from text. But $BERT$ needs only the encoder part since its goal is to generate a language representation model. Thus it is a multi-layer bidirectional Transformer encoder.

There are two types of pre-trained versions of $BERT$ depending on the scale of the model architecture [8]: $BERT$-base contains 12 transformer blocks (layers), 12 attention heads and 110 million parameters and $BERT$-large contains 24 layers, 16 attention heads and 340 million parameters.

Functionality

The input to the encoder for BERT is a sequence of tokens, which are first converted into vectors and then processed in the neural network. But before processing can start, BERT needs the input to be massaged and decorated with some extra metadata:

- Token embeddings
- Segment embeddings
- Positional embeddings

The input representation for BERT: the input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings.

A word starts with its embedding representation from the embedding layer. Every layer does some multi-headed attention computation on the word representation of the previous layer to create a new intermediate representation. All these intermediate representations are of the same size. In a 12-layers BERT model a token will have 12 intermediate representations.

In this project we are taking advantage of the power of $BERT$ to train a language-alone baseline for the VQA task.

4. Methods

In attempt to measure the performance that a better language model can provide to VQA, we essentially have the same method as [1] and [2] when training a language-alone model. Specifically, in our procedure a pre-trained $BERT$-base model is used instead of $LSTM$ to obtain 1024-dim embeddings for the question. This question embeddings is then fed into a feed-forward classifier with 2 fully connected layers and $ReLU$ non-linearity followed by a softmax layer to obtain a distribution over 1000 answers. And finally, the answer with the highest activation among them is selected as the predicted answer.

For the purpose of training a language-alone baseline, our model accepts only the questions as inputs and the top $K = 1000$ most frequent answers as possible outputs. Our model was trained on the training set and tested on the validation set from the balanced version of the VQA dataset [2] using Pytorch and Hugging Face's Transformers. The model is then learned end-to-end with a binary cross entropy with logits loss and optimized using AdamW with careful chosen set of hyper-parameters.

$$accuracy \quad = \quad min\left(\frac{n}{3}, 1\right) \tag{4.1}$$

where n: number of the matching answers.

To evaluate the performance of our model, we use the standard evaluation metrics [1] defined in the equation 4.1. It shows that a predicted answer is considered 100% accurate if the number of its matching among the 10 provided answers is at least 3.

For testing, the dataset papers for VQA suggest to report results on test-standard accuracies and compare their methods' accuracies. However, for the sake of time restriction our results are reported on validation only.

5. Results

The experimental curves during training and validation time are shown in Figure **??**.

	d-$LSTM$	$BERT$-base
accuracy	43.01	43.9

Table 5.1: Accuracy of $BERT$ compared to d-$LSTM$ (reported in [2]) on VQA $v2.0$ val

Table 5.1 shows our experimetal results. At a glance, we notice a small improvement about 1% of the accuracy from using pre-trained $BERT$ which is slightly lower than we expected. However there is certainly room for improvement.

Despite this, we can still state that in general VQA task benefits from having a better language model like $BERT$.

6. Conclusion

The findings of this study support the idea that VQA task benefits from better language model like $BERT$.

Acknowledgements

I gratefully acknowledge the help provided by Pr. Stefan Lee for supervising me along this work.

Bibliography

[1] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. *VQA: Visual Question Answering*, 2016.

[2] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra and D. Parikh. *Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering*, 2017.

[3] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, in International Conference on Learning Representations (ICLR), 2015.

[4] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[5] S. Hochreiter and J. Schmidhuber, *Long short-term memory*, 1997.

[6] K. Cho, B. Van Merri¨enboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, *Learning phrase representations using rnn encoder-decoder for statistical machine translation*, in Conference on Empirical Methods on Natural Language Processing (EMNLP), 2014.

[7] R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, A. Torralba, R. Urtasun, and S. Fidler, *Skip-thought vectors*, in Advances in Neural Information Processing Systems (NIPS), 2015.

[8] J. Devlin, M. Chang, K. Lee and K. Toutanova, *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, 2019.

[9] Z. Yang, X. He, J. Gao, L. Deng, and A. J. Smola, *Stacked attention networks for image question answering*, in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[10] B. Zhou, Y. Tian, S. Sukhbaatar, A. Szlam, and R. Fergus, *Simple baseline for visual question answering*, arXiv preprint arXiv:1512.02167, 2015.

[11] K. Kafle and C. Kanan. *Visual Question Answering: Datasets, Algorithms, and Future Challenges*, 2017.

[12] Q. Wu, D. Teney, P. Wang, C. Shen, A. Dick and A. van den Hengel. *Visual Question Answering: A Survey of Methods and Datasets*, 2016.

[13] A. K. Gupta. *Survey of Visual Question Answering: Datasets and Techniques*, 2017.

[14] Y. Srivastava, V. Murali, S. R. Dubey, and S.Mukherjee. *Visual Question Answering using Deep Learning: A Survey and Performance Analysis*, 2020.

[15] Matthew Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power, *Semi-supervised sequence tagging with bidirectional language models*, 2017.

[16] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever, *Improving language understanding with unsupervised learning*. Technical report, OpenAI, 2018.

[17]

[18]