LSTM in VQA-Med, is it really needed? JCE study on the ImageCLEF 2019 dataset

Avi Turner[1] and Assaf B. Spanier[1]

Department of Software Engineering of Azrieli College of Engineering Jerusalem, Israel assaf.spanier@mail.huji.ac.il

Abstract. This paper describes the contribution of the Department of Software Engineering at the Azrieli College of Engineering, Jerusalem, Israel to the ImageCLEF VQA-Med 2019 task. This task was inspired by the recent ever greater success of visual question answering (VQA) in the general domain. Given medical images accompanied by clinically relevant questions, participating systems were tasked with answering questions based on the image content.We explored and implemented a two-stage model. The first stage predicts the category of the textual question, while the second stage is comprised of 5 sub-models. Each sub-model is a classic VQA deep learning module with two branches for feature extraction, the first using CNN to extract image features, and the second using embedding (and optionally LSTM) to extract textual features. The network then combines the two feature branches to predict the appropriate answer. We found that most sub-models didnt need LSTM to achieve high scores on the validation and test data-sets. We submitted 10 models for the challenge, our best submission overall ranked 9th out of 17. All source codes are available at https://github.com/turner11/VQA-Med

Keywords: VQA-Med · LSTM · ImageCLEF-2019.

1 Introduction

The ever increasing demand for automated computer systems (AI) to assist clinical medical practice addresses two main audiences: Doctors who use these systems to get a second opinion on their diagnosis; and patients who increasingly have easy access to comprehensive and detailed medical data which they find bewildering. Thus, addressing patients, the systems motivation is to help them have a better understanding of their medical condition, by providing detailed explanations of the results of their medical tests and scans, which is something that doctors, naturally, are unable to do for each data item of each patients file. The current access to ones detailed medical file without explanation leads to the unfortunate situation that patients turn to searching the Internet and online forums to better understand their condition, reaching misleading information

and false conclusions. Consequently, this often worries patients, either because insufficiently specific details of their health are considered, or even worse, because irrelevant, false, inexpert information is found.

Visual question answering (VQA) [1] is a subfield of automated systems (AI) relevant to these kinds of problems. The task of VQA is to produce textual answers to textual questions asked in the context of a specific image. This is illustrated in Fig 1: given an image and a question, a VQA system should supply an answer relevant to the question in the context of the image.

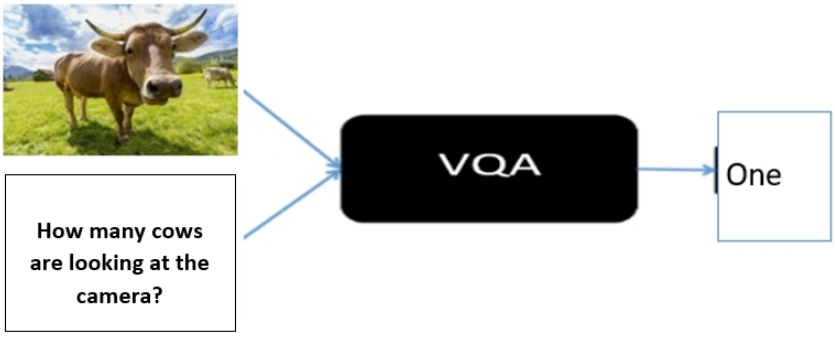

Fig. 1. Given an image and a question, a VQA system should supply an answer in the context of the question and the given image.

A VQA [2] system questions takes textual questions as input with the images they refer to, and combines data from the image and the question text to arrive at the most relevant answer. To produce answers to specific questions, VQA systems combine natural language processing methods with advanced computer vision techniques. The application of VQA to the field of medicine is a twofold challenge, not only are medical texts and images significantly different from those in the general computer vision field, but the resources and labelled data available in the medical field are quite limited relative to what is available in the general field. Evidenced by the 260,000 image COCO-QA Challenge dataset of general images, this quantity contrasted with the 5,000 VQA-Med medical image dataset. Following the recent successes of VQA in the general computer vision field and the challenge posed by the medical field, as of 2018, ImageClef 2019 [3] published a second round of the VQA-Med Challenge [4]. This paper deals with the problems of VQA in the medical field. The rest of this paper is organized as follows. First, we describe some related work. Next, we describe the database and challenge characteristics. Then, we describe our method in detail. Lastly, results are presented in Section 4, followed by conclusions and future work in the last Section.

2 Related Work

The VQA COCO-QA Challenge is studying a problem very similar to the VQA-Med task. VQA [1] has been held every year since 2016. The dataset is public domain based. The prevalent approach to VQA uses recurrent neural networks, such as LSTMs [5], to encode the textual questions, and deep convolution networks, such as VGG-16, to encode and extract features from the images [6]. Based on these ideas, a plethora of other methods have been proposed in the literature: including attention, dynamic models, and even incorporating external databases.

In this study, we took a different approach: our objective was to use classic VQA methods [7]. We analyzed those methods in order to determine their advantages and limitations with respect to the necessity of the LSTM layer and other parameters. We utilized conventional VQA approaches, optimizing their parameters, to find the best prediction method and its corresponding imaging and text features, which provided the best evidence as to whether or not the LSTM layer is necessary to achieve a high score or not.

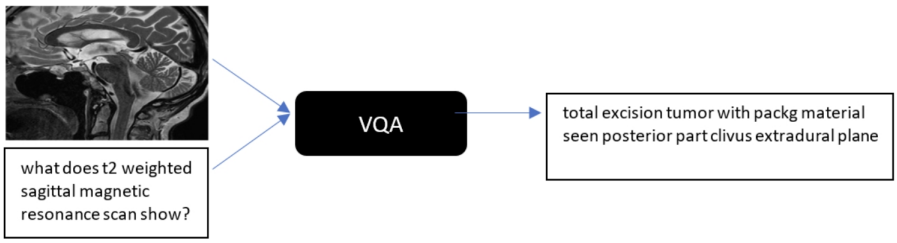

Fig. 2. VQA-Med, texts as well as images pertaining to the medical field are significantly different and more complex.

3 Task Description and Dataset

The Challenge dataset comprised of a training dataset of 3,200 medical images and 12,792 Question and Answer (QA) pairs, a validation dataset of 500 medical images and 2,000 QA pairs, and a test dataset of 500 medical images and 500 questions with answers withheld. The questions are divided into 4 categories: Modality, Plane, Organ System and Abnormality.

The evaluation of the participant systems of the VQA-Med 2019 task was conducted based on two metrics: BLEU, and Accuracy (Strict). Accuracy (Strict) is an adapted version of the accuracy metric from the general domain VQA task that considers exact matching of a participant provided answer and the ground truth answer. BLEU [10] is used to capture the similarity between a system generated answer and the ground truth answer. Each answer is converted to

lower-case, all punctuation was removed, and the answer was tokenized to individual words. Stopwords were removed using NLTKs9 English stopword list. Snowball stemming10 was applied to increase the coverage of overlaps.

4 Methods

The input to our method is an image and a questions referring to it. The output is an answer for the question in the context of the given image. Fig 2. The system is comprised of two stages: The first predicts the category of the textual question Fig 3, while the second is a classic VQA module which combines the question and image to predict a relevant answer (see Fig 4 below). The first stage classifies the question into 5 question categories: Modality, Plane, Organ System and 2 Abnormality categories: Note, we subdivide the tasks given Abnormality class into two categories: Questions with a Yes or No answer, and All Other Questions. This stage uses embedding and an MLPClassifier, and it optimizes the log-loss function using LBFGS or stochastic gradient descent. We used the sklearn package [8] for this, with its default parameters.

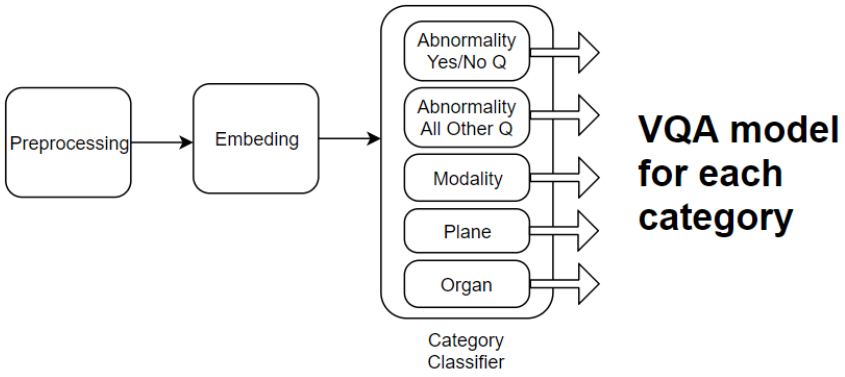

Fig. 3. The first stage predicts the category of the textual question. Classifying into the following 5 question categories: Modality, Plane, Organ System, Abnormality, Yes/No and Abnormality, Other

The second stage is a classic VQA deep learning module, which takes the question and image as a combined input and predicts the appropriate answer. The text undergoes preprocessing and embedding the output of this branch is treated as features. We investigated whether LSTM was needed at the next stage or not. Image features are extracted using a CNN network (VGG 19) [9]. Next, the features from both branches are merged using fully-connected layers

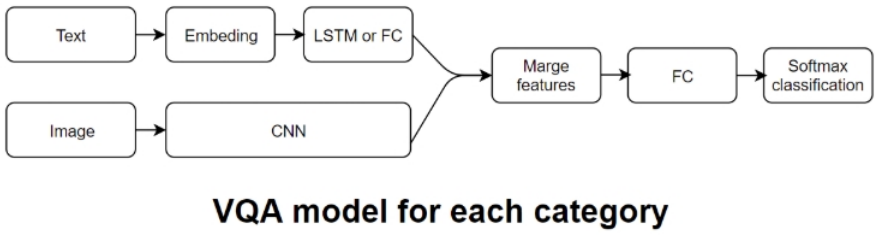

Fig. 4. A classic VQA deep learning module, which takes the question and image as a combined input and predicts the appropriate answer

5 Results

We will start with a short comparison of our results with those of other groups, then we will focus in on our own submissions. When participating groups were compared using only the best performing submission from each group we placed 9th out of 17 groups, achieving a strict accuracy score of 0.53 compared to 0.62 achieved by the best performing groups submission. Looking at the number of submissions we submitted 10, while the average number of submissions per group was 4.7 (80 submissions by 17 groups). However, in term of average performance across all submissions we placed 8th, trading places with LIST due to the small variation in performance between our submissions, which were all between places 25 and 44 (of 80 total submissions), while LIST groups submissions were between places 24 and 57.

We will review and analyze the submissions in order of the scores they achieved on the test set, looking particularly at the following features of the submitted models: 1) Optimization Function, 2) Activation Function, 3) Loss Function, 4) batch size, 5) size of fully-connected layers, 6) number of units in the LSTM layer, and 7) whether Class Weights were used.

Since this paper is focused on finding whether LSTM is needed for VQA tasks, and we wanted the effect and contribution of LSTM to be highlighted, we chose to work with a very simple convolution network, and used the VGG network. when evaluating optimization functions we found that RMSprop produced the best results across all the submitted models. Results clearly indicated using Softmax for Optimization alongside Categorical Crossentropy as the Loss Function was the best option, which was expected as they are the most natural choice for a task like this [10]. These were the parameters used in the three highest performing submissions (by test set scores). Batch size was 32 for all question categories, except the Abnormal – Yes/No category which required a batch size of 75, submissions with lower batch sizes produced less accurate results.

Lets turn to the last three parameters:

- Size of fully-connected layers
- Number of units in the LSTM layer
- Whether Class Weights were used

5.1 Submissions Details

We will review our ten submissions in order of their Challenge test set results. Note that each submission is comprised of five sub-models – one per question category.

In the tables presented per model, each row represents a sub-model (for a question category), the columns are:

- Column 1 – question category that sub-model was trained for
- Column 2-3 – sub-models validation set scores (strict-accuracy and BLEU)
- Column 4 – size of fully-connected layers
- Column 5 – number of units in the LSTM layer (LM in short)
- Column 6 – Loss Function used
- Column 7 – Activation Function used (act, in short)
- Column 8 – batch size
- Column 9 – number of epochs (epo, in short)
- Column 10 – whether Class Weights were used

Best Performing Submission The submission with the highest test set score had the following characteristics: Fully-connected layer size of 14 for all sub-models This is the highest number used among our submissions, and our findings indicate that a higher number of fully-connected layers was more successful in generalizing from the validation set. An LSTM layer was used only in the sub-model handling the Abnormality – Other question category. In the training and validation datasets Yes and No answer frequencies were not balanced for the Abnormality Yes/No category. We therefore investigated whether class weights would improve accuracy and found that they did. See test set results in Table 4 and validation results in Table 1

Table 1. Best performing submission. cross. stands for categorical crossentropy, Abnorm for Abnormality, LM for LSTM, epo for epochs, act for activation

category	acc	bleu	FC	LM	loss	act.	batch size	epo	class weight
Organ	0.7	0.70	14	0	cross	softmax	32	7	NO
Plane	0.74	0.74	14	0	cross	softmax	32	7	NO
Modality	0.82	0.82	14	0	cross	softmax	32	10	NO
Abnorm.	0.724	0.76	21	128	cross	softmax	32	3	NO
Abnorm. yes no	0.02	0.05	14	0	cross	softmax	75	2	yes

2nd and 3rd best performing submissions The differences between these models and the best submission were not great, nor were the differences in the scores they both achieved 0.53 strict and 0.55 BLEU. Compared to the best

performing submission, these two had fewer epochs and smaller fully-connected layer sizes. See test set results in Table 4 and validation results in Table 2 and Table 3

Table 2. 2nd performing submission. cross. stand for categorical crossentropy, Abnorm for Abnormality, LM for LSTM, epo for epochs, act for activation

Category	Accuracy	BLEU	FC	LM	Loss	act.	batch size	epo	class weight
Organ	0.66	0.68	14	0	cross	softmax	32	7	NO
Plane	0.74	0.74	8	0	cross	softmax	32	5	NO
Modality	0.72	0.75	14	0	cross	softmax	32	10	NO
Abnorm.	0.02	0.04	21	128	cross	softmax	32	3	NO
Abnorm. yes no	0.78	0.78	14	0	cross	softmax	75	2	YES

Table 3. 3rd performing submission. cross. stand for categorical crossentropy, Abnorm for Abnormality, LM for LSTM, epo for epochs, act for activation

Category	Accuracy	BLEU	FC	LM	Loss	act.	batch size	epo	class weight
Organ	0.63	0.65	14	0	cross	softmax	32	9	NO
Plane	0.72	0.72	14	0	cross	softmax	32	7	NO
Modality	0.72	0.75	14	0	cross	softmax	32	7	NO
Abnorm.	0.02	0.02	21	128	cross	softmax	32	3	NO
Abnorm. yes no	0.78	0.78	14	0	cross	softmax	75	2	YES

Table 4. Test accuracy of all 10 summations

submission rank	accuracy	BLEU
1	0.54	0.55
2	0.53	0.55
3	0.52	0.57
4	0.53	0.558
5	0.52	0.55
6	0.52	0.55
7	0.52	0.57
8	0.50	0.54
9	0.50	0.51
10	0.50	0.56

Submissions 4,5,6,8,9 These submissions either did not use Class Weights at all or did not use them exclusively for the Abnormality – Yes/No category, and the size of fully-connected layers was smaller, emphasizing the importance of these elements to the network. See test set results in Table 4,

Submissions performing 7th and 10th These submissions did not include LSTM, their low scores proving the importance of this layer for handling complex tasks such as the Abnormality – Other question category. See test set results in Table 4,

6 Conclusions

This paper presents research done in the context of participation in the VQA-Med Challenge. We analyzed VQA classifiers and feature extraction methods for image and text classification in the context of medical images in the VQA-Med 2019 task. We found that none of the sub-models needed LSTM, except the one handling the Abnormality Other questions category, the most complex task, which also required fully-connected layers of size 21, unlike all the other categories, for which fully-connected layers of size 14 were sufficient. Class weights are needed only in cases were a significant imbalance between answer class frequency exists as there was in this challenge in the Abnormality Yes/No question category. We submitted 10 models, our best submission ranking 9th out of 17. All source codes are available at https://github.com/turner11/VQAMED,

7 Future Work

This paper focused on the question of whether and when LSTM may be useful for VQA tasks. We therefore chose to work with a very simple convolution network, the VGG network. Further research on the effect and contribution of the LSTM module is needed in order to look at a broader range of convolution networks, including more advanced versions, such as ResNet and Inception, and their effect on results. We intend to investigate the effects of using larger size fully-connected layers and more epochs. Looking at batch size, found that our best performing submissions had a batch size of 32, with 75 for the Abnormality Yes/No sub-model, while all lower batch sizes produced less accurate results. The batch size was limited by computing resources, and we intend to examine larger batch sizes with stronger processors.

References

1. Kushal Kafle and Christopher Kanan. Visual question answering: Datasets, algorithms, and future challenges. *Computer Vision and Image Understanding*, 163:3–20, 2017.

2. Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015.
3. Bogdan Ionescu, Henning Müller, Renaud Péteri, Duc-Tien Dang-Nguyen, Luca Piras, Michael Riegler, Minh-Triet Tran, Mathias Lux, Cathal Gurrin, Yashin Dicente Cid, et al. Imageclef 2019: Multimedia retrieval in lifelogging, medical, nature, and security applications. In *European Conference on Information Retrieval*, pages 301–308. Springer, 2019.
4. Asma Ben Abacha, Sadid A. Hasan, Vivek V. Datla, Joey Liu, Dina Demner-Fushman, and Henning Müller. VQA-Med: Overview of the medical visual question answering task at imageclef 2019. In *CLEF2019 Working Notes*, CEUR Workshop Proceedings, Lugano, Switzerland, September 09-12 2019. CEUR-WS.org.
5. Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
6. Kushal Kafle and Christopher Kanan. Answer-type prediction for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4976–4984, 2016.
7. Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C Lawrence Zitnick, Devi Parikh, and Dhruv Batra. Vqa: Visual question answering. *International Journal of Computer Vision*, 123(1):4–31, 2017.
8. Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.
9. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
10. Pieter-Tjerk De Boer, Dirk P Kroese, Shie Mannor, and Reuven Y Rubinstein. A tutorial on the cross-entropy method. *Annals of operations research*, 134(1):19–67, 2005.