INTERNSHIP REPORT

ON

VQA USING DEEP LEARNING

Under the supervision of

Dr AK DHAMIJA

By

HEMANT DHAMIJA

2016A7PS0031G

Internship Completed At

DIPR, Defence R&D Organisation, Ministry of Defence, Delhi

During the degree of B.Tech from

Birla Institute of Technology & Science, Pilani

AUGUST 2019

Certificate

The internship report titled **"VQA using Deep Learning"** being submitted by Mr. **Hemant Dhamija** to the DIPR, Defence R&D Organisation, Ministry of Defence, Delhi, is a record of bonafide work carried out by him. He has worked under my guidance and supervision, and has fulfilled all the requirements for the submission of this report, which has attained the standard required for such report of this institute. The data/results presented in this report have not been submitted elsewhere.

(AK Dhamija)
Scientist 'F', Head Human Engineering Division
DIPR, Lucknow Road, Timarpur
Ministry of Defence, Delhi

Acknowledgments

I would like to express my deep sense of gratitude to **Dr AK Dhamija** and **Dr K Ramachandran, Director, DIPR** for their invaluable help, guidance and inspiration during the course of Practice School-I. I am thankful to them for constantly encouraging me by giving their constructive and critical evaluation on my work.

I acknowledge all other members of DIPR, Delhi, who have enlightened me about the nuances of Deep Learning process and report writing and inspired me throughout this internship, especially at difficult times.

I am grateful to my fellow intership team members for enthralling technical discussions all through the intership. I am grateful to my friends in BITS Pilani, Goa Campus and elsewhere, who have contributed a lot in my learning during the internship.

Hemant Dhamija

July 2018

Abstract

Visual Question Answering (VQA) is challenging task which combines two important branches of Artificial Intelligence viz Computer Vision and Natural Language Processing. Computer Vision enables computers to understand and process images in the same way that a human does and Natural Language Processing enables computers to understand and analyze human language. The objective of VQA systems is to predict the right answer given both image and question about that image in a natural language. The VQA task can be a classification problem if the answer is chosen from given choices or as a generation problem if the answer is generated as a well-formed textual description.

In this study, features of total 14782 images are extracted and saved in features.pkl. Total 135020 questions are extracted along with their answers. 20 epochs are used for training of 135020 questions in each epoch and the epoch model giving minimum loss is considered for predicting the answers. In sample testing is giving an accuracy of 48.69%.

Contents

List of Figures

Glossary

Convolutional Neural Network(CNN)

Regularized versions of multilayer perceptrons most commonly applied to analyzing visual imagery.

Long Short Term Memory(LSTM)

An artificial recurrent neural network (RNN) architecture having feedback connections along with standard feedforward neural networks. It is mainly used to process entire sequences of data like such a speech or video data.

Recurrent Neural Network(RNN)

A class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence.

Visual Geometry Group(VGG)

Deep convolutional network for object recognition developed and trained by Oxford's renowned Visual Geometry Group, which achieved very good performance on the ImageNet dataset.

Visual Question Answering(VQA)

Use of video cameras to transmit a signal to a specific place, on a limited set of monitors.

Acronyms

CNN

Convolutional Neural Network.

LSTM

Long Short Term Memory.

RNN

Recurrent Neural Network.

VGG

Visual Geometry Group.

VQA

Visual Question Answering.

Chapter 1

Course Work

Following topics are studied and presented along with worked out examples.

1. Basic Statistics including Data Visualization, z, t and χ^2 distributions
2. Sampling, central limit theorem and inferential statistics
3. ANOVA, MANOVA, ANCOVA, MANCOVA
4. Multivariate regression along with regression diagnostics
5. Randomized (Permutation) t-test and ANOVA
6. Hadoop and HDFS setup involving 1 name node and 2 data nodes and worked out examples of Map/reduce framework on sample data after saving the data into 2 data nodes of HDFS

Chapter 2

Project Work

2.1 Introduction

Visual Question Answering (VQA) is challenging task which combines two important branches of Artificial Intelligence viz Computer Vision and Natural Language Processing. Computer Vision enables computers to understand and process images in the same way that a human does and Natural Language Processing enables computers to understand and analyze human language. The objective of VQA systems is to predict the right answer given both image and question about that image in a natural language. The VQA task can be a classification problem if the answer is chosen from given choices or as a generation problem if the answer is generated as a well-formed textual description.

In the last few years, Deep Neural Networks have achieved success in various problem areas like image recognition (Simonyan and Zisserman, 2014), (Krizhevsky et al., 2012), machine translation (Cho et al., 2014), (Krizhevsky et al., 2012),image captioning (Sutskever et al., 2014) and Visual Question Answering (Malinowski et al., 2016), (Zhou et al., 2015). This internship report presents my work for the problem of VQA using a combination of Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM) which is a version of Recurrent Neural Network (RNN). I consider the task of VQA as multi-label classification problem, where each label corresponds to a unique word in the answer dictionary that was built from the training set. The project, thus, combines the CNN and LSTM models in order to learn to answer the questions pertaining to an image.

2.2 Execution Requirements

1. Keras with graphviz installed

2. "Training" folder in the current working directory containing the $Quest\_answers.json$ and images folders containing training images for training and in-sample testing

3. "Test" folder in the current working directory containing the $Quest\_answers.json$ and images folders containing test images for out of sample testing

4. "testimages" folder n the current working directory containing containing test images for self_test()

Saved files are

1. Tokenizer.pickle for tokenizer
2. descriptions.txt for questions
3. trainimages.txt for storing names of files
4. model_14.h5 for trained model
5. model.png for model architecture
6. features.pkl for saving features of all images
7. vgg16model.png for saving
8. max_length.pkl to store the maximum length of questions after training

2.3 Network Architecture

1. Training
   (a) Image Features - CNN
      i. The image features are computed using pre-computed weights of VGG16 (by popping out the last layer from this. The architecture for this CNN portion is depicted in the figure. All images are initially converted to $224 \times 224 \times 3$ size and input layer is $224 \times 224 \times 3$.
      ii. After this there are two convolution layers of 64 filters each and thus creating output of $224 \times 224 \times 64$. Next layer is Max pooling layer of size 2×2 and thus creating output of $112 \times 112 \times 64$.

iii. After this there are two convolution layers of 128 filters each and thus creating output of $112 \times 112 \times 128$. Next layer is Max pooling layer of size 2×2 and thus creating output of $56 \times 56 \times 128$.

iv. After this there are three convolution layers of 256 filters each and thus creating output of $56 \times 56 \times 256$. Next layer is Max pooling layer of size 2×2 and thus creating output of $28 \times 28 \times 256$.

v. After this there are three convolution layers of 512 filters each and thus creating output of $28 \times 28 \times 512$. Next layer is Max pooling layer of size 2×2 and thus creating output of $14 \times 14 \times 512$.

vi. After this there are three convolution layers of 512 filters each and thus creating output of $14 \times 14 \times 512$. Next layer is Max pooling layer of size 2×2 and thus creating output of $7 \times 7 \times 512$.

vii. After this there is a flatten layer creating output of $7 \times 7 \times 512 = 25088$ processing elements.

viii. After this there are two fully connected ie. dense layer containing 4096 processing elements each. Thus the final layer creates 4096features for each image.

The CNN-VGG16 Layers are shown in Figure 2.1. These extracted features are stored in features.pkl so that it can be loaded again to extract the features of test images which essentially form a subset of training images. The Visual Geometry Group (VGG) pre computed weights will have to be used to extract the image features if there are new image is to be tested.

The overall architecture of combined LSTM-CNN is shown in Figure 2.2

i. Question Features - LSTM: The questions are read from $Quest\_answers.json$ file and These questions are then saved in the form of pair of triplet of (image, question, answer) in the file descriptions.txt, which can be utilized later on for feeding into the LSTM and the merged network. .

ii. Since the maximum length of question is 42, so input layer of 42 is used for LSTM. The 42 words are tokenized and mapped to unique numbers. The next layer is

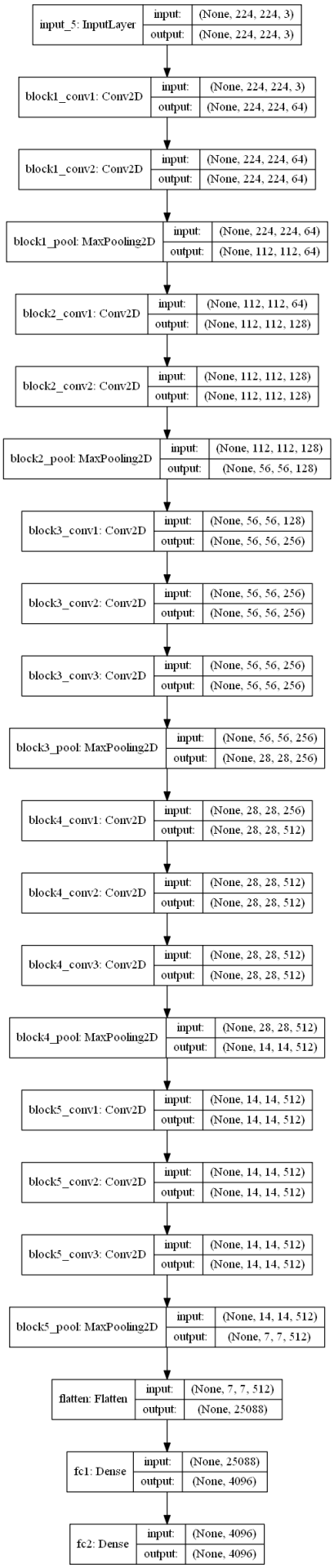

Figure 2.1: CNN-VGG16 Layers

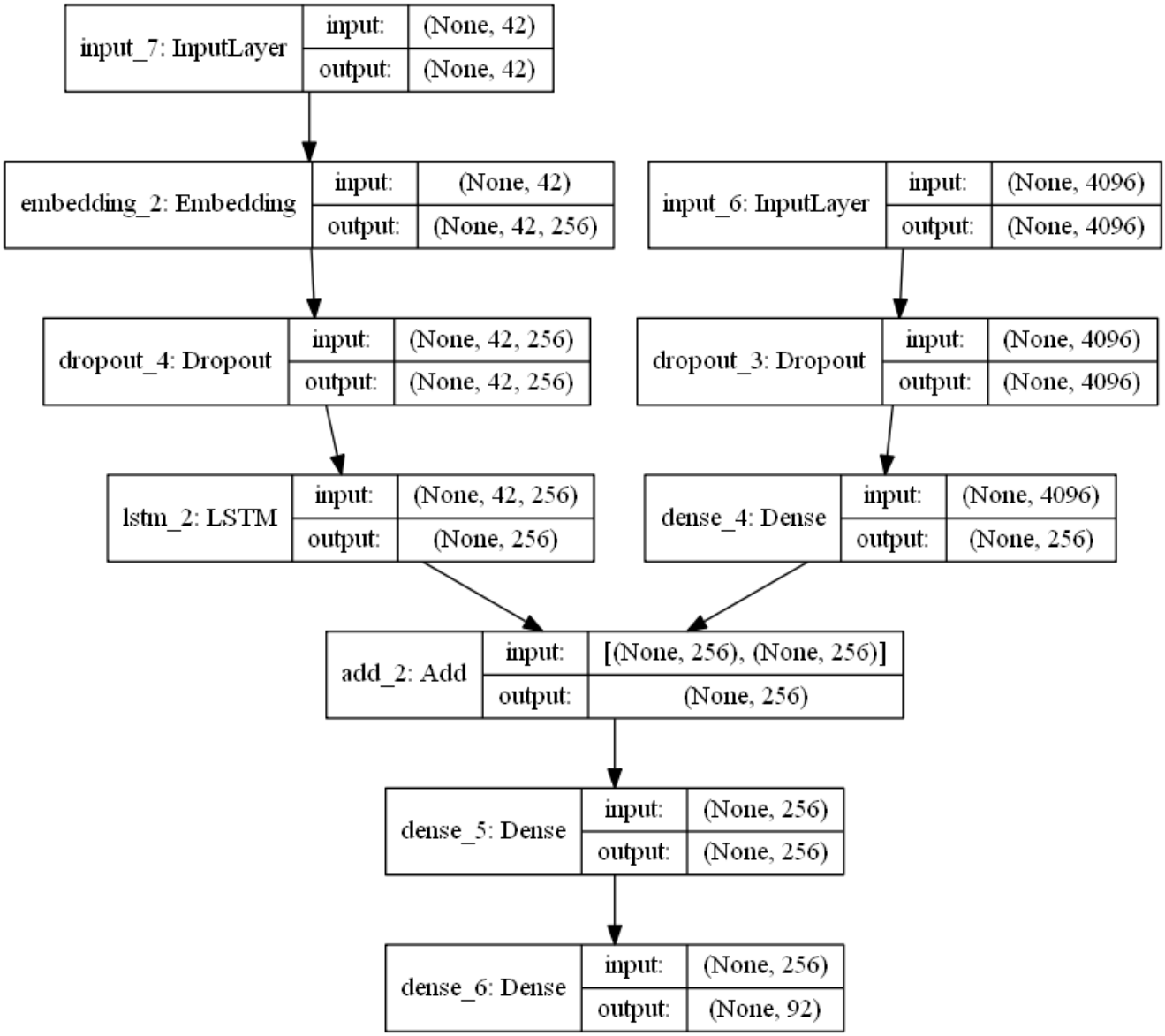

Figure 2.2: LSTM-CNN combined architecture

embedding layer which transforms each word i.e. the associated number to a 256 dimensional vector space, thus each word will have 256 dimensions and the embedding layer is 42×256 dimensional

iii. The next layer is a Dropout layer with 50% elements dropout as regularization parameter and this layer is also 42×256 dimensional. Next is 256 dimensional LSTM layer (effectively creating 256 layers of LSTM.

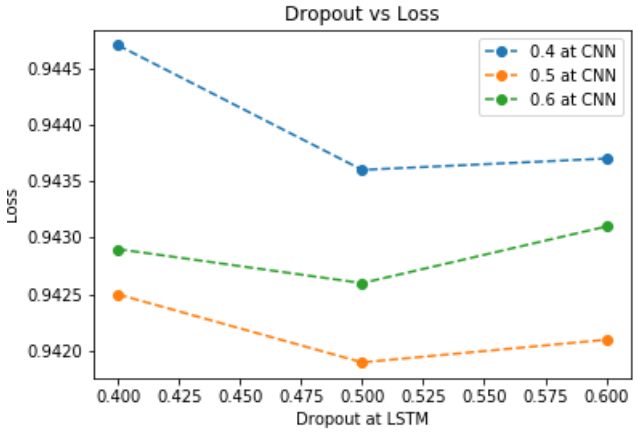

Figure 2.3: Dropout vs Loss

iv. On the CNN side also next layer is a 4096 dimensional dropout layer. The model is trained for various values of dropout parameters at LSTM and CNN final layers (combinations of (0.4, 0.5, 0.6) (0.4, 0.5, 0.6) are tried because training time is very large) and and 50% dropouts were found to be optimums (Figure **??**) contributing to least loss and best accuracy. The next layer is dense fully connected 256 dimensional layer. So both 256 dimensional LSTM and CNN layers are then merged and connected to 256 dimensional output layer.

v. The next layer is 256 dimensional dense fully connected layer and the final layer is 92 dimensional fully connected layer, because the total number of distinct labels (i.e. answers to the questions) are 92.

2. Testing: testing can be done in three ways

   (a) In sample testing using function test_testset()

(b) Out of sample testing by function test_new_images(). This assumes a test directory "Test" containing "Quest_Answers.json" file for questions and a subdirectory named "images" containing test images

(c) by manually providing the images names and the questions in the function self_test()

2.4 Results

Features of total 14782 images are extracted and saved in features.pkl. Total 135020 questions are extracted along with their answers. 20 epochs are used for training of 135020 questions in each epoch and the epoch model giving minimum loss is considered for predicting the answers. In sample testing is giving an accuracy of 48.69%.

REFERENCES

Cho, K., B. van Merrienboer, Ç. Gülçehre, F. Bougares, H. Schwenk, and Y. Bengio (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR abs/1406.1078*.

Krizhevsky, A., I. Sutskever, and G. E. Hinton (2012). Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, USA, pp. 1097–1105. Curran Associates Inc.

Malinowski, M., M. Rohrbach, and M. Fritz (2016). Ask your neurons: A deep learning approach to visual question answering. *CoRR abs/1605.02697*.

Simonyan, K. and A. Zisserman (2014). Very deep convolutional networks for large-scale image recognition. *CoRR abs/1409.1556*.

Sutskever, I., O. Vinyals, and Q. V. Le (2014). Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, Cambridge, MA, USA, pp. 3104–3112. MIT Press.

Zhou, B., Y. Tian, S. Sukhbaatar, A. Szlam, and R. Fergus (2015). Simple baseline for visual question answering. *CoRR abs/1512.02167*.