Artificial Neural Networks and Deep Learning 2021

Third Competition – Visual Question Answering

Team Members: Leonardo Giusti, Francesco Govigli, Lorenzo Mainetti

SCOPE

Given a dataset made of images, questions, and answers, implement a performant algorithm using Neural Networks to predict the most probable and feasible answer for a specific question on a specific image. The dataset is composed by synthetic scenes in which people and objects interact, and by question about evident details of an image.

The possible answers/labels are:

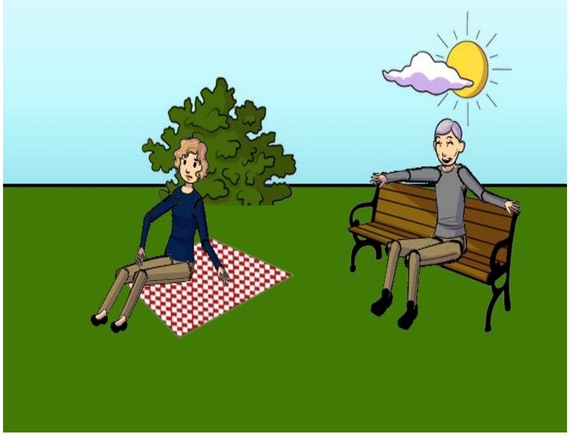
['0', '1', '2', '3', '4', '5', 'apple', 'baseball', 'bench', 'bike', 'bird', 'black', 'blanket', 'blue', 'bone', 'book', 'boy', 'brown', 'cat', 'chair', 'couch', 'do’, 'floor', 'food', 'football', 'girl', 'grass', 'gray', 'green', 'left', 'log', 'man', 'monkey bars', 'no', 'nothing', 'orange', 'pie', 'plant', 'playing', 'red', 'right', 'rug', 'sandbox', 'sitting', 'sleeping', 'soccer', 'squirrel', 'standing', 'stool', 'sunny', 'table', 'tree', 'watermelon', 'white', 'wine', 'woman', 'yellow', 'yes']

The evaluation is based on Multiclass Accuracy, which is simply the average number of observations with the correct label.

EXAMPLE:

Input:
- Question: “Who looks happier?”
- Figure shown on the left

Output:
- Answer: “man "



MORE DETAILS (Dataset Structure)

The dataset is made of a set of images and a json file containing the questions and the corresponding answers for training and another json containing just questions for testing. To be noticed is the fact that some questions can refer to the same image.

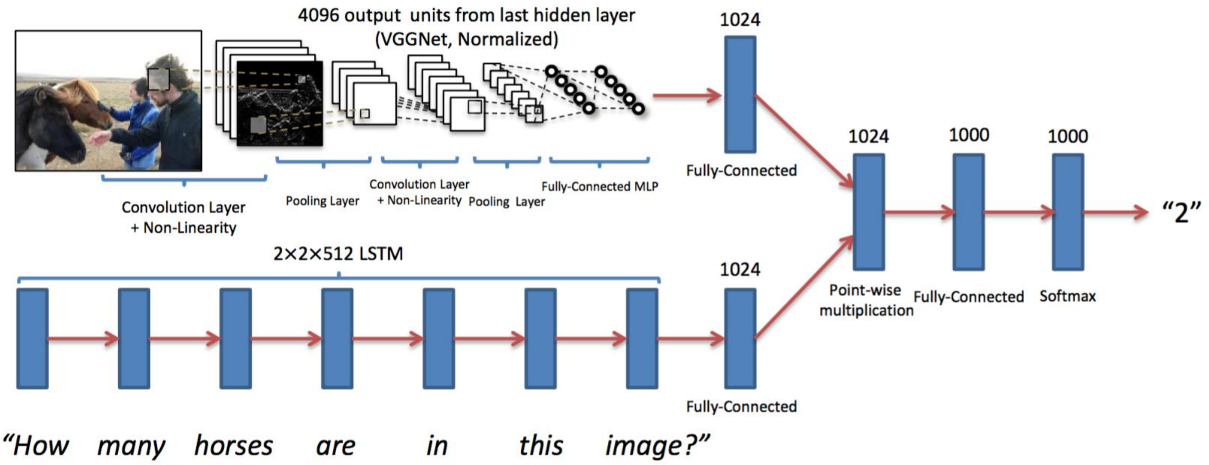
Dataset preprocessing

Initially, to make all reproducible, we inserted a random seed in our notebook. We used a dataframe to encapsule the data tuple (Image, Question, Answer) and split it in a training set made of 90% of the original set and a validation set made of 10% of the samples.

Since the dataset has a considerable size, we decided to reserve just a 10% of it for the validation and we did not apply any kind of data augmentation. Also, we kept the images to their original size 400x700 in all our experiments.

Building a baseline

Our baseline was built following the typical structure of a VQA model (in particular, we took as a starting point the structure described in this paper: https://arxiv.org/pdf/1505.00468v6.pdf). Our first model was built using a pretrained ResNet50 to handle the features extraction from the images, where the fully connected layers were removed, and all layers were frozen. In parallel, to handle text (questions), we added an LSTM layer made of 512 units, and after adding a Dense layer on the top of both the CNN and the LSTM, we merged the two through concatenation. Eventually we added just two more Dense layers, being the last one a Softmax to be able to classify the output.



From baseline to our best model performance

Our baseline obtained an accuracy of 0.44 on the test set, so we started to tune the parameters. We first switched to the VGG-16 as transfer learning network, following the model presented in the paper and we got a slight improvement. We also unfroze the last layers and re-trained them to learn high level features specific for our problem. We then increased the learning rate from 1e-4 to 1e-3 and tried different combinations of sequence_length and vocab_size. We noticed that a greater vocab_size (i.e. closer to the actual number of different words in the question vocabulary) improves the performance, so we set it to 4000. We also focused on the batch size, 16, 32, 64 were tried, but we eventually opted for 16, mainly because of computational issues. Indeed, with more computational power we could probably get a better result.

But what really made the difference was the introduction of the Ensemble Method. We trained 3 different networks with the same structure for 7 epochs each and on different datasets. After the training we averaged the results obtaining an improvement of around 0.02 with reference to the single model. Thanks to the Ensemble we got our best accuracy result, 0.62335 on the test set.

Possible improvements

A possible improvement to our model could be:

- Attention Mechanism => probably it would not give such a great improvement if we apply it to the questions, due to their length. Indeed, attention is a powerful tool when we have long sequences, and this is not the case. We could still try it. Or we could also try to apply it to the CNN feature maps, to better focus on a specific region of the image (Attention => weighted feature maps)

Furthermore, we could totally change our model and try the state of the art for NLP, the Transformer, even though it would probably be overkill in this case.