HOMEWORK 2, ARTIFICIAL NEURAL NETWORKS AND DEEP LEARNING
INNOCENTE – MADDES – MASCIULLI – MORREALE

We started from the notebook on recurrent neural networks seen at the exercise session with prof. Lattari, along with all the precedents.
We wrote the code to obtain the dictionary in order to encode answers and questions. The train set is formed by tuples: image name, answer, question. We then applied some basic data augmentation: horizontal flip, weight and height shift, zoom, rotation. We also tried not to augment the train set, but the results were strictly similar and we preferred a more efficient training.
We tried to split the data in training and validation following different proportions: 80-20, 75-25. Data is split in a stratified fashion, using answers as the class labels. We tried different combinations of image dimension and batch size during training in order to have a good trade off between fitting of the model and time spent for each epoch. To obtain an efficient model we tried different architectures:

- Custom model:we won't directly use the image as input into the model. The image is scaled to 200×350. The scaled image is fed into a convolutional neural network (CNN) such as VGG-16 which outputs a feature vector encoding the contents of the image and is referred to as an image embedding. The question is fed into an embedding layer, resulting in a question embedding. These embedding vectors, which compactly represent the image and question contents have different dimensions. Hence they are first projected into the same number of dimensions using corresponding fully connected layers (a linear transformation) and then combined using pointwise multiplication (multiplying values at corresponding dimensions). The final stage of the VQA model is a multilayer perceptron with a softmax nonlinearity at the end that outputs a score distribution over each of the 58 answers. Converting the answers to a 58-way classification task allows us to train the VQA model using a cross-entropy loss between the generated answer distribution and the ground truth.
- Co-attention: this model implements a mechanism that jointly reasons for visual attention and question attention. More specifically, the image representation is used to guide the question attention and the question representation(s) are used to guide image attention. It attends to the image and question simultaneously; we connected the image and question by calculating the similarity between image and question features.

We noticed an unbalanced situation on the training set regarding the different labels so we computed a weighted loss giving more importance to the less frequent answers (weights inversely proportional to class frequency).
Alternatively, we wrote a custom function to compute weights for each class and passed the generated dictionary to the function fit. We also tried to remove samples from the train set in order to balance all the classes, but the results were not good, due to underfitting.
We found out that the unbalanced data were not impacting the overall performance. On the other hand, the weighting procedures reduced the training time performance.
For all the models, during the training, we used early stopping and reduce on plateau functions to get the best results. We also set a scheduler for the learning rate based on the number of epochs but at the end we did not use it. The value of parameters of these functions are mainly based on the current learning rate and on the number of parameters of the entire model.