PROJECT-2

Image Captioning using CNN and LSTM

Group Members:

1. Bhargav Sai Bhuvanagiri - B191228EC
2. Buddala Ranama Vidya sagar -B190838EC
3. Maddi sai vardhan - B191144EC
4. Meda Phanendra Kumar - B190385EC

Aim:

1. To implement an Image captioning model using CNN and LSTM, where CNN acts as encoder(feature extractor) and LSTM acts as decoder(decodes the sequence of words that suits the feature).

Algorithm Description:

Text preprocessing:

1. Converting texts into lowercase.
2. Removing any special characters
3. Removing whitespace.
4. Removing single characters like(a, I , e.t.c).
5. Adding starting sequence and ending sequence code word.

Tokenization:

1. The words in each sentence are tokenized/separated.
2. Now the unique tokens are extracted and given an one-hot encoded vector representation,but we used an efficient word embedding which reduces the dimensionality that is word2vec.

Image Features Extraction:

1. We used a pre-trained model called DenseNet201 Architecture.
2. Since the Global Average Pooling layer is selected as the final layer of the DenseNet201 model for our feature extraction, our image embeddings will be a vector of size 1920
3. So we choose our output layer as the second last layer of densenet 201 model.Since it's a classifier the last layer is nothing but the classification part with soft max as activation function but we need only features.So we used (outputs=model.layers[-2].output) ”-2” here.

Long Short term Memory(LSTM):

1. Concatenating the image features and their respective sentence_features and sending it to the LSTM model to train.
2. Now we drop 50 percent of sentence features and add it to image features and send it to a fully connected layer(FFNN) to get output.

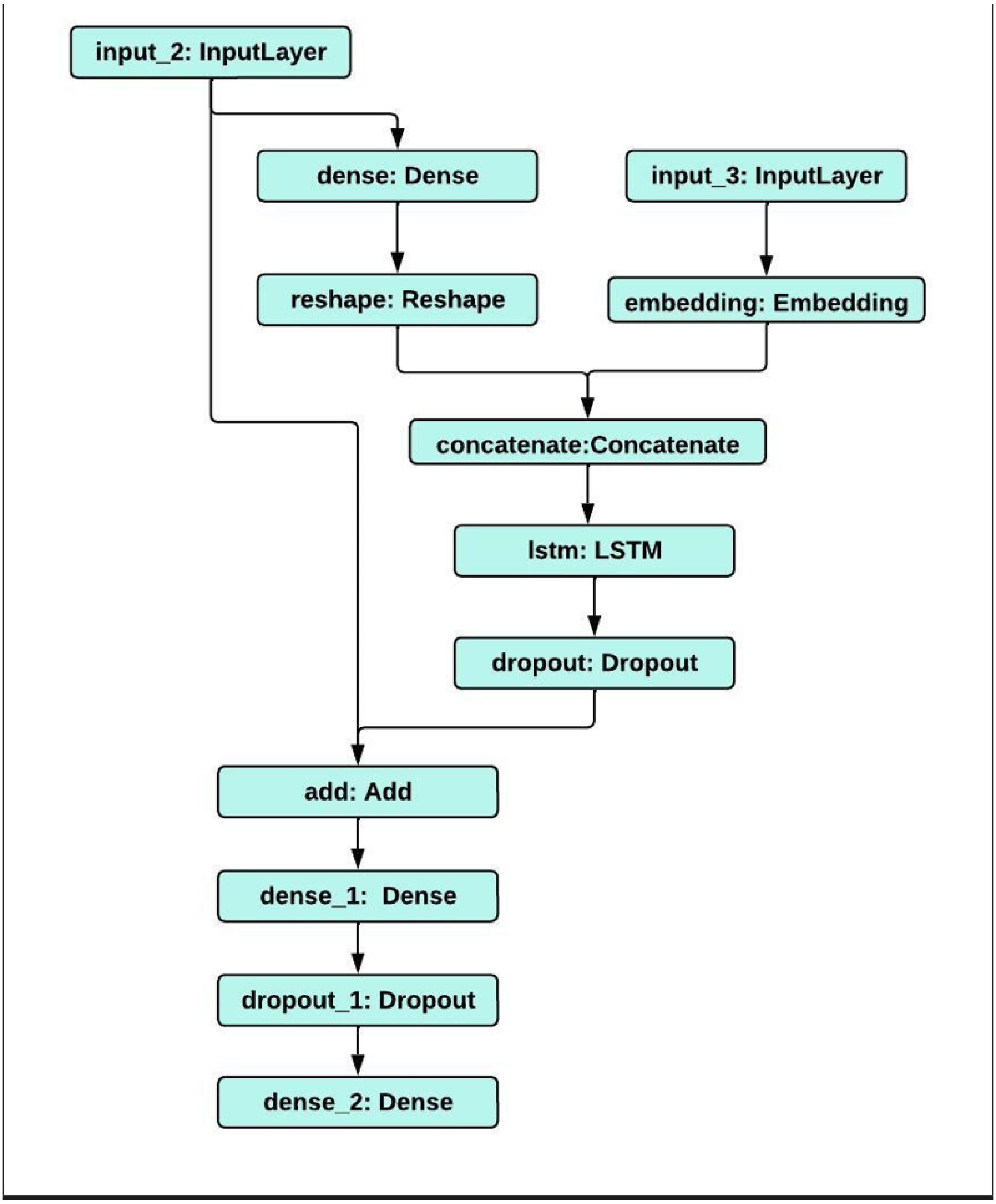


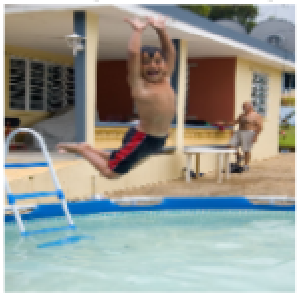
Fig: decoder LSTM overview.

Code: “Attached as .ipynb file.”

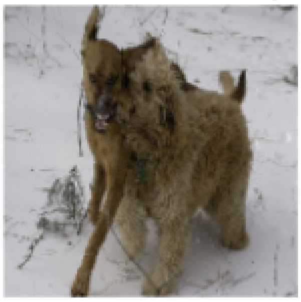
Results:

1. Correctly predicted images

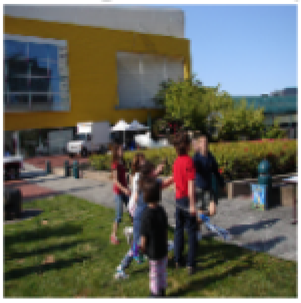
startseq boy jumping into pool endseq



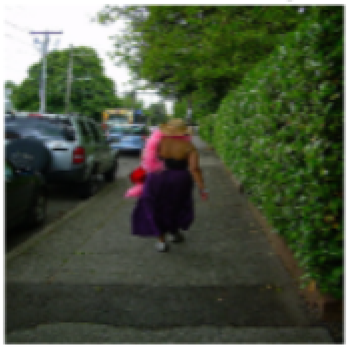
startseq two dogs are running through the snow endseq



startseq group of people are playing in the grass endseq

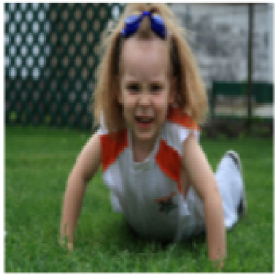


startseq the woman is walking down the street endseq



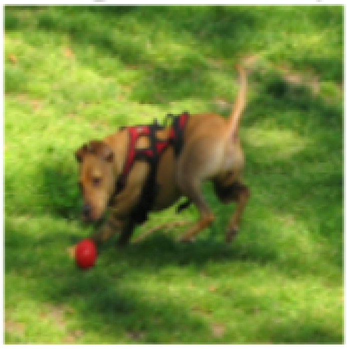
2. Wrongly predicted images
    a. Wrong colour prediction(white and some orange but it predicts pink)

startseq little girl in pink dress is playing in the grass endseq



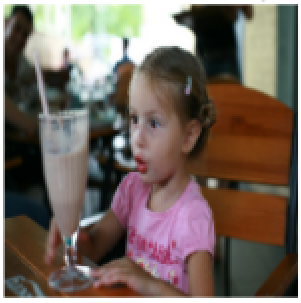
b. Number of objects predicted wrong

startseq two dogs
are running through
the grass endseq



c. Object detection went wrong

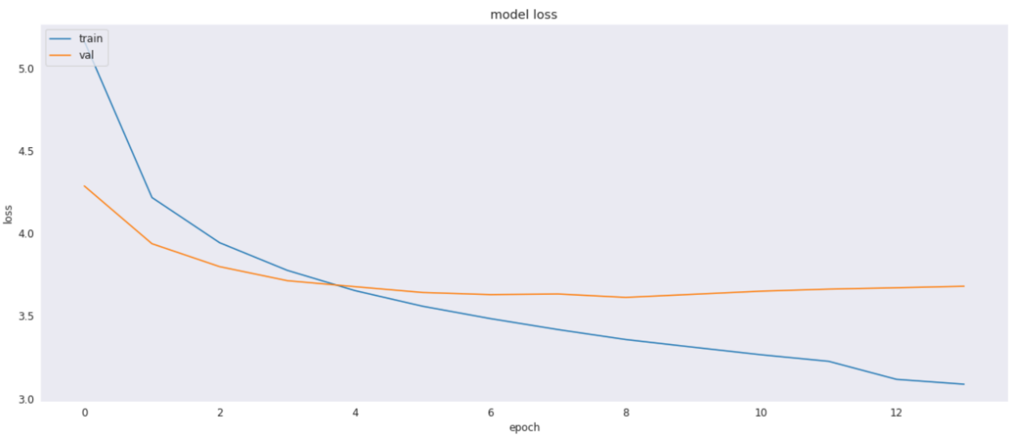
startseq baby in
blue shirt is
sitting on the
camera endseq



3. There are some other minor wrong prediction errors which is not important as the above mentioned cases.Overall this is pretty good at generating images from the given image.

Other results:

1. Learning Curve


model loss
train
val
loss
epoch