VISUAL RECOGNITION MINIPROJECT IMAGE CAPTIONING

Group: JKSS

Jishnu V K
IMT2018033
International Institute of Information Technology Bangalore
Vinodkumar.Jishnu@iiitb.org

Karthik Hegde
IMT2018509
International Institute of Information Technology Bangalore
Karthik.Hegde@iiitb.org

Saad Patel
IMT2018514
International Institute of Information Technology Bangalore
Mohammad.Saad@iiitb.org

Saravan sriram
IMT2018521
International Institute of Information Technology Bangalore
Saravan.Sriram@iiitb.org

I. INTRODUCTION

Automatic Image Captioning is a very well-known problem in the area of deep learning, NLP, and Computer Vision. Given an image, the model has to generate a text description of the image. In this project, we try to build the image captioning deep learning model with plain vanilla CNN-LSTM encoder-decoder and with CNN-LSTM using soft-attention mechanism.

II. VANILLA CNN-LSTM ENCODER DECODER

A. Model Architecture

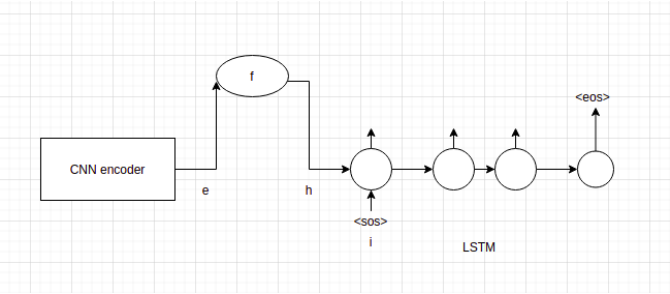

Fig. 1. Vanilla encoder-decoder

The plain vanilla encoder-decoder is the first step that we took to solve the task of image captioning. This architecture, as shown in the diagram 1, consists of CNN encoder and one-layer LSTM decoder. In addition, we added another feed-forward layer (f) to the output of the CNN making the LSTM hidden vector size flexible to choose and to improve training. At the first time step, the input is the embedding of the start token ('$< sos >$'), and encoder output (after passing to feed-forward) acts as the initial hidden vector to the LSTM. For the CNN part, we mainly used pre-trained models like VGG, Resnet. We extract only the first sequential layers (before Adaptive pooling) from the pre-trained models for the CNN encodings.

B. Training Methodology

For training, we took only the tuple $(img, caption\#1)$ as training instance. In essence, only the first image caption of the corresponding image was taken to be the ground truth while training. For the validation, we consider all the 5 image captions, and the lowest loss score is treated as the judging factor.

For the initial experiments, we implemented only the teacher forcing method during training, but in the later experiments, we moved towards a mixed type of teacher forcing and non-teacher forcing methods. In the mixed type technique, teacher-forcing is used only for the first few time steps (T_i), and later on the training shifts to the non-teacher forcing technique. In essence, at the initial steps, teacher-forcing guides the model architecture, and as the training progresses model starts to train on the previous outputs. We mainly experimented with $T_i = 10$ out of total of 163 time steps.

C. Experiments

We used Vggnet and Resnet as the pretrained CNN models for encoder part. We tried experimenting with different values of embedding size, hidden vector size, encoder input size. For both of them the best configurations were embedding size of 256, encoder size of 512, hidden vector size of 512, LSTM dropout of 0.5. We set learning rate at $5 * 10^{-4}$, batch size of 32, and we mainly used Adam optimizer.

D. Results

The table I shows the BLEU scores for the mentioned configurations of the model. Table II shows the sentences generated for the corresponding subjective image sample. The sentences are in the order of image number starting from 1 to 5. In this case the mix-technique doesn't give a boost to the scores. Both Vggnet and Resnet scores are similar.

TABLE I
BEST RESULTS WITH VANILLA ENCODER-DECODER

Model	Size	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Vggnet (Teacher-forcing)	22,745,181 (90.98 MB)	val score = 0.54 test score = 0.54	val score = 0.317 test score = 0.32	val score = 0.20 test score = 0.20	val score = 0.129 test score = 0.124
Vggnet (mix-technique)	22,745,181 (90.98 MB)	val score = 0.52 test score = 0.53	val score = 0.306 test score = 0.314	val score = 0.194 test score = 0.196	val score = 0.12 test score = 0.12
Resnet (Teacher-forcing)	19,147,023 (76.59 MB)	val score = 0.545 test score = 0.549	val score = 0.315 test score = 0.32	val score = 0.20 test score = 0.20	val score = 0.125 test score = 0.126
Resnet (mix-technique)	19,147,023 (76.59 MB)	val score = 0.499 test score = 0.50	val score = 0.285 test score = 0.287	val score = 0.18 test score = 0.179	val score = 0.11 test score = 0.11

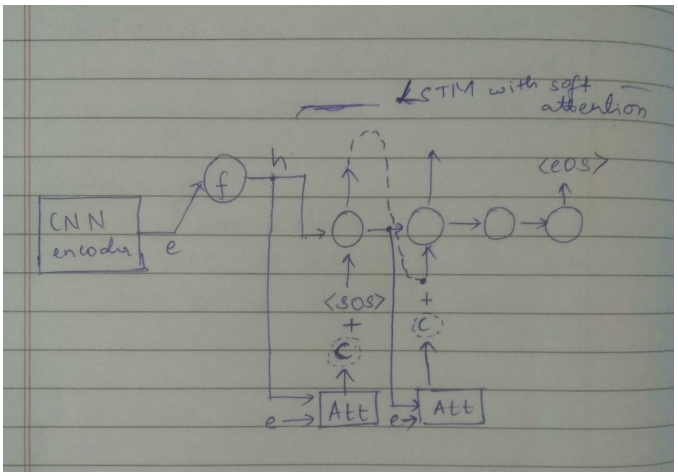

Fig. 2. CNN-LSTM encoder decoder with soft attention

III. CNN-LSTM WITH SOFT ATTENTION

A. Model Architecture

During our experiments from the vanilla encoder-decoder model, we observed a very well-known fact that with increased string length size (larger time steps), the model was not able to give good meaningful sentence outputs. Theoretically, the model was experiencing a bottleneck in encoding the entire image. It required some kind of attention mechanism. Referring to the paper Show, Attend and Tell, we succeeded in implementing the soft-attention mechanism in our encoder-decoder architecture.

TABLE II
SUBJECTIVE RESULTS WITH VANILLA ENCODER-DECODER

Model	Subjective sentences
Vggnet (Teacher-forcing)	boy in black shirt is doing skateboard trick group of people ride bicycles down street crowd of people are standing in front of large building group of people are walking over large white and white dog dog jumps into air to catch tennis ball
Vggnet (mix-technique)	boy in black shirt is doing skateboard trick group of people ride bikes on street crowd of people are standing in front of building group of people are walking over large white and white dog dog jumps into air to catch tennis ball
Resnet (Teacher-forcing)	boy is doing skateboard trick in air while another man is standing behind him group of people are riding bicycles down street group of people are standing in front of some stores two dogs are playing together in field brown dog is running on beach
Resnet (mix-technique)	man in black shirt is sitting on bench in front of wooden barn group of people are standing on street with their arms outstretched group of people are standing in front of some stores black dog is running in field of plants and houses brown dog is jumping over hurdle

The diagram 2 shows the overview of the model architecture. The implementation uses Bahdanau (local) attention, using hidden vector and encoder output as inputs. The output will be concatenated with the embedding input before passing to the LSTM (single layer) layer. In this way, the decoder will be able to attend to the required part of the image for outputting the current caption element.

The following part describes the attention mechanism and LSTM input mathematically.

$$F_{att} = V_{att}^{T} * tanh(U_{att} * enc_j + W_{att} * h_{t-1}) \quad (1)$$

enc_j refers to encoder output at jth pixel (cuboid along the image depth), h_{t-1} refers to hidden vector of the previous time step. U_{att}, W_{att}, V_{att} are feed forward networks.

$$e_{jt} = F_{att} \quad (2)$$

$$\alpha = softmax(F_{att}) \tag{3}$$

The α term is later used in the regularization term.

$$c_t = \sum_{j=1}^{T} \alpha_{jt} * enc_j \tag{4}$$

c_t is the context vector for the time-step t.

$$\beta = sigmoid(f_{beta} * h_{t-1}) \tag{5}$$

f_{beta} is the feed-forward network.

$$c_t = \beta * c_t \tag{6}$$

β acts as the gating term. This is given in the paper.

$$h_t = LSTM([embedding(y_{t-1}), c_t]) \tag{7}$$

y_{t-1} is the output at the previous time step.

B. Training Methodology

Training methods are very much similar to the methods followed in vanilla encoder-decoder methods. We tried with both teacher forcing and mixed-type technique ($T_i = 10$ and $T_i = 15$). And it turns out that limiting teacher-forcing technique to 15 time steps or so will help boost the scores. Similar to vanilla encoder-decoder methods, here also we take tuple $(img, caption\#1)$ as training instance (only first caption per image) while training. During validation, we take the lowest score of the output among the five captions as the judging factor.

However, one major difference between training via attention mechanism is that we added a regularization term given in the paper known as doubly stochastic regularization.

C. Experiments

We tried many different model hyperparameter configurations, such as changing the hidden vector size, encoder input size, embedding size, attention size (F_{att} size in attention mechanism), LSTM dropout, regularization constant in the regularization term. We also tried changing the optimizer from Adam to RMSprop, and both of them work well. We tested with changing the number of epochs, learning rate, batch size, as well as techniques from teacher-forcing to mixed-type technique.

The tables III and IV lists the hyperparameter configurations that we experimented with for Googlenet and Vggnet & Resnet.

D. Results

The tables V and VI show the results of the five best models and their corresponding sentences on the subjective images. The subjective sentences are in the order of the image number starting from 1 to 5. We used corpus_bleu function from the NLTK library to compute the BLEU score which is used as an eval metric.

TABLE III
TESTED CONFIGURATIONS FOR GOOGLENET

Encoder size	1024
Embedding size	512
Hidden size	512
Attention size	512
LSTM dropout	0.4, 0.5
regularization constant	1, 0.3
number of epochs	20, 40
learning rate	10^{-4}, 10^{-3}
batch size	32, 64

TABLE IV
TESTED CONFIGURATIONS FOR RESNET, VGGNET

Encoder size	512
Embedding size	256
Hidden size	512
Attention size	256
LSTM dropout	0.4, 0.5, 0.6
regularization constant	1, 0.3, 1.4
number of epochs	20, 30, 40
learning rate	10^{-4}, 10^{-3}
batch size	32, 64

A Comparison between the BLEU scores and the subjective sentences indicates that though Googlenet + LSTM scores are less than the Resnet and Vggnet counterparts, it outputs more reasonable sentences on the subjective images. In fact, the first output sentence that 'boy in black shirt is sitting on bench in front of building' is almost perfect. They all give reasonable sentences on the second and third images and mess up in the last two images. In the fourth subjective image, the model doesn't recognize the groom and bride from behind. In the fifth image, the word dinosaur is not available in the training vocabulary and hence the closest answer would be a dog.

IV. PREPROCESSING DETAILS

On using lemmatized text, the BLEU scores would have been much better (nearing 0.64 on test). But since the subjective results would have made no sense, we resorted to the normal text. As a pre-processing step, we normalized the text sentences - remove points and punctuation, article (a/an/the), and lower case the sentences - and then build the vocabulary from the training captions alone. Since all of the training image captions are within the limit of 163 words, the LSTM uses max of 163 time steps. The vocab size consists of 7631 unique tokens.

V. CONCLUSION

In this work, we have learnt to implement and experiment with the CNN-LSTM encoder-decoder model with and without

TABLE V
BEST RESULTS WITH ATTENTION MECHANISM

Model	Size	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Googlenet (Teacher-forcing)	19,982,960 (79.93 MB)	val score = 0.561 test score = 0.549	val score = 0.33 test score = 0.326	val score = 0.212 test score = 0.206	val score = 0.136 test score = 0.127
Vggnet (Teacher-forcing)	24,259,344 (97.04 MB)	val score = 0.579 test score = 0.579	val score = 0.349 test score = 0.354	val score = 0.226 test score = 0.226	val score = 0.1437 test score = 0.142
Vggnet (mix-technique)	24,259,344 (97.04 MB)	val score = 0.588 test score = 0.588	val score = 0.349 test score = 0.357	val score = 0.222 test score = 0.228	val score = 0.142 test score = 0.145
Resnet (Teacher-forcing)	20,721,168 (82.88 MB)	val score = 0.588 test score = 0.579	val score = 0.354 test score = 0.344	val score = 0.227 test score = 0.212	val score = 0.145 test score = 0.128
Resnet (mix-technique)	20,721,168 (82.88 MB)	val score = 0.591 test score = 0.591	val score = 0.353 test score = 0.352	val score = 0.2245 test score = 0.222	val score = 0.143 test score = 0.139

attention. After a thorough process of experimentation, we conclude that although soft-attention is one step ahead of the plain vanilla encoder-decoder models in generating captions and sentences, they are far from perfection, which is evident from the metric scores. The other challenge ahead is to implement the mechanisms and techniques to generalize on unseen data.

VI. REFERENCES

- Show, Attend and Tell
- Bahdanau, local attention mathematical formalism

TABLE VI
SUBJECTIVE RESULTS WITH ATTENTION MECHANISM

Model	Subjective sentences
Googlenet (Teacher-forcing)	boy in black shirt is sitting on bench in front of building group of people are riding on street group of people are standing on rock with their arms out-stretched group of people are in white shirts in front of large structure brown dog is jumping in air on sand
Vggnet (Teacher-forcing)	boy in black shirt is standing on sidewalk man in red shirt is riding bike on dirt bike group of people are standing in front of people group of people are standing on beach dog is jumping in air in air
Vggnet (mix-technique)	boy in red shirt is sitting on bench man in red shirt is riding bike on bike group of people are sitting on bench two dogs are playing in front of water dog is jumping in air
Resnet (Teacher-forcing)	boy is sitting on skateboard on skateboard group of people are riding on street group of people are standing in front of water group of people are standing on beach brown dog is jumping in air
Resnet (mix-technique)	boy is standing on skateboard group of people are riding on street group of people are standing in front of building two dogs are playing in front of water brown dog is running through water