Assignment 5 - Recurrent Neural Networks, Attention and Reading Comprehension

Manasa Bollavaram

March 19, 2017

1 CBOW Model

1.1 Performance

CBOW model has poor performance on the given dataset. One of the main reasons behind this, is that every word in the question is given equal priority for finding the answer. This is confusing the model because if the question contains a proper noun and the proper noun occurs many times throughout the context, the model is not able to learn to downweight this similarity. Sometimes, answers are spread across the context.

To improve the model, long-term dependencies need to be included in learning. This can be done by using LSTMs or GRUs. Also, a weighted-average of question vector embeddings will work better than uniform averaging.

2 Gated Recurrent Unit

2.1 Performance

For this part of the assignment, GRU Cell with hidden layer dimension = 200 is implemented. This solves the problem of long-term dependency to a great extent. However, this does not solve the problem of uniform averaging. Hence, we can not expect great performance from this model either. The model is trained for 2500 steps with batch size of 128. Results are listed in Table 2.

Without dropout, F-measure of the model is 27.45. Dropout is expected to regularize the model and improve its performance in the long run. However, given the short training

Corpus	Hidden Size	Batch Size	Acc1%	Acc2%	EM	F1
Validation	50 (Default)	64 (Default)	6.23	8.34	5.05	11.01
Test	50 (Default)	64 (Default)	5.04	5.46	2.95	9.79

Table 1: CBOW Performance on Validation and Test Set

Corpus	Dropout Scheme	QVec	Loss	Acc1%	Acc2%	EM	F1
Val	None	Avg	5.91	22.46	23.39	17.77	27.45
Test	None	Avg	6.66	16.60	18.68	14.15	24.54
Val	Output Keep Prob (x, q) = 0.9	Avg	6.05	19.75	22.69	16.02	26.44
Test	Output Keep Prob (x, q) = 0.9	Avg	6.78	15.26	17.67	12.42	23.07
Val	Output Keep Prob (x, q) = 1.0, 0.7	Avg	6.26	18.69	21.06	14.69	24.24
Test	Output Keep Prob (x, q) = 1.0, 0.7	Avg	6.98	14.41	17.05	11.30	21.80
Val	Output Keep Prob (x, q) = 1.0, 0.9	Avg	6.09	19.92	22.55	16.24	25.80
Test	Output Keep Prob (x, q) = 1.0, 0.9	Avg	6.79	15.78	18.31	12.93	23.53

Table 2: GRU Performance with Dropout Scheme; Model is trained for 2500 steps

Corpus	Loss	Acc1%	Acc2%	EM	F1
Val	5.89	22.07	23.69	18.42	29.06
Test	7.93	7.35	7.19	2.89	12.09

Table 3: Attention Model Performance

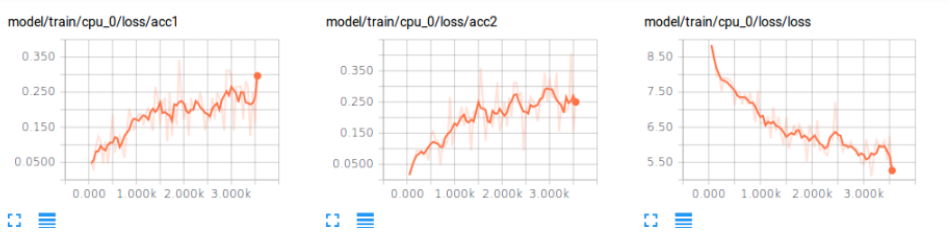

Figure 1: Smoothed graphs of Accuracy1, Accuracy2 and Loss for Attention Model a seen in TensorBoard

time of this model, it is difficult to comment on the effectiveness of various dropout schemes experimented here. The performance drops marginally for small values of dropout (at input as well as output) and by up to 3 points for dropout of 0.3.

3 Attention

For this part of the assignment, the softmax Attention model as mentioned in homework has been implemented along with the GRU Cell used in previous part of the homework. The learning curve starts with an encouraging steep, but flattens out after approximately 4000 steps. The learning rate is dropped when this happens but the curve flattens again after a slight steep. This shows that the Attention model implemented here is lacking sufficient representation. Performance on test set also shows signs of overfitting. Improvements can be made to the module by way of increasing number of layers within the GRU.