PA2 report

Ziao, LIU 20494087
zliuca@connect.ust.hk

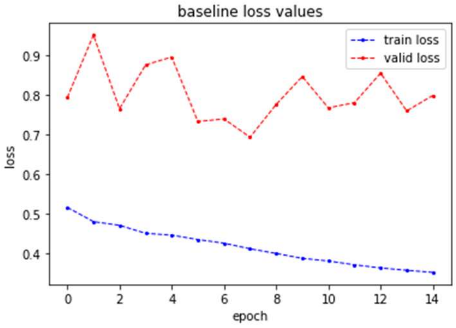
Q1:
Number of Out-of-Vocabulary words in training set is 8650.
The method via torchtext getting OOV value is that:

Firstly, build a vocab object of txt_field with min_freqs=0, i.e. all words would be in vocab. Then build a vocab with min_freqs=2, ignore those words appearing only one time. Finally, get the difference between the length of two vocabs.

Q2:

```
==========================================================
         Kernel Shape  Output Shape   Params Mult-Adds
Layer
0_emb       [50, 7953]  [64, 20, 50]  397.65k   397.65k
1_rnn          -          [64, 20, 64]   7.424k    7.296k
2_linear      [64, 1]   [64, 20, 1]     65.0      64.0
3_sigmoid        -       [64, 20, 1]        -         -
----------------------------------------------------------
                    Totals
Total params          405.139k
Trainable params       405.139k
Non-trainable params       0.0
Mult-Adds              405.01k
==========================================================
```

Q3:



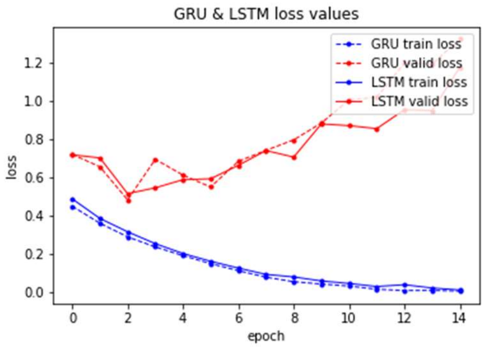
The highest validation accuracy is 0.5596.

Q4:
Advantages using LSTM:

1, Compared with general RNN, GRU and LSTM have the ‘memory’ cell that could ‘remember’ things for longer time, which lead to a shorter training period.

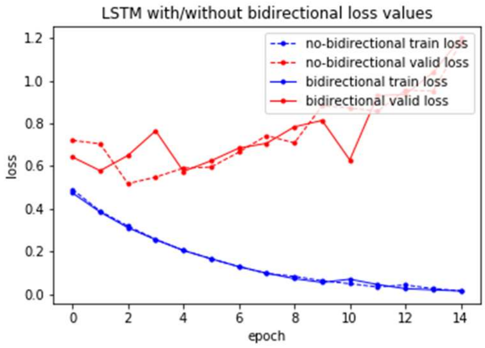
2, It is difficult to train standard RNN to solve problems with learning long-term temporal dependencies. Respectively, training GRU and LSTM would be easier and the accuracy would be higher.

Q5:



From figure, both GRU and LSTM training to the minimum validation loss using 3 epochs, while the minimum validation loss of standard RNN reached at 5th or 6th epoch. From the record of validation accuracy, the GRU one is 0.8110, the LSTM one is 0.7861, both of them are higher than the standard one.

Q6:



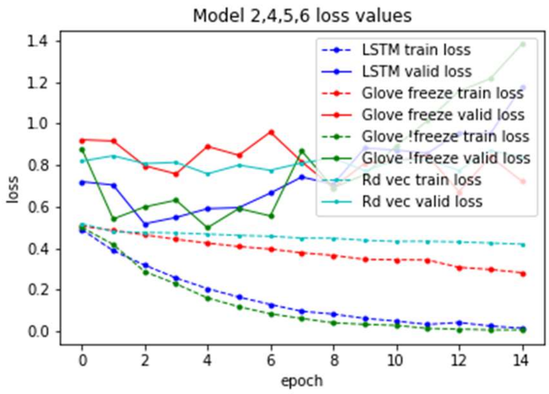
From the figure, when bidirectional parameter set to false, the validation loss, the accuracy and convergence rate of the validation loss performed better than the one with bidirectional set true. However, the training loss values of the two different models were nearly the same and both convergence with similar speed. We could also find a huge peak and a huge gap appeared on the curve of the bidirectional-on validation loss.

Q7a:
- \`happy' and \`good' : 2.7164
- \`france' and \`germany' : 3.9774
- \`france' and \`happy' : 6.3060

Q8a:
My expectation: GloVe without freeze would perform the best result. Then the standard LSTM. The worst one is the random vector and freeze set true. Cause the pretrained embedding model could help to initialize the relation of words at beginning, then the

embedding model would be trained and updated to meet the requirements of the task. If the model is frozen, it would hardly perform well. If the GloVe embedding model could help at beginning, then the random vector could not provide any assistance in training.

Q9a:

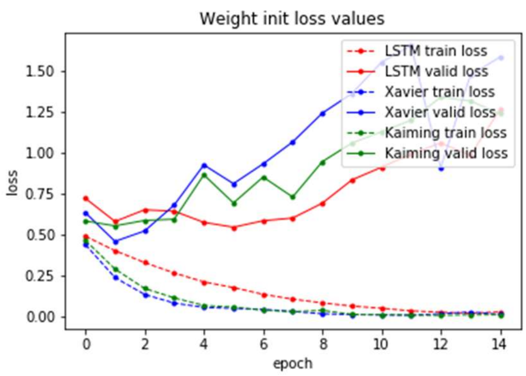


The result in the Figure is obvious that Glove unfrozen embedding model perform the fastest convergence of validation loss, the highest validation accuracy. When applying random vector, the model did not train.

The Highest accuracy:
Standard LSTM: 0.7861
Glove with freeze: 0.7061
Glove without freeze: 0.8179
Random vector: 0.5220

Q7b:
Weight initialization is a method with providing prepared information to shorten the training time consuming and obtain a better training result. If the task is complex, all zero weight or bad weight initialization would make a worse performance in training. Then when initializing weight, the method should be selected properly.

Q8b:



Standard LSTM: 0.7861
Xavier: 0.7998
Kaiming: 0.7974

Q9b:
With weight initialization, after second epoch, Xavier model and Kaiming model are over fitted and the validation loss are lower than that in standard LSTM, which indicated that the training efficiencies with weight initialization are higher than original standard LSTM. From the highest accuracy, Xavier model and Kaiming model performed better than standard one.

Prediction:
As I ran the code for several times, I select model7 as the standard model to predict the test set. In different running tern, the highest accuracy values of models are various. Hence I select model with better performance.