Machine Learning Homework 4 Report

R06221012 數學所 李岳洲

April 31, 2020

1. 請說明你實作的 ***RNN*** 的模型架構、***word embedding*** 方法、訓練過程 ***(learning curve)*** 和準確率為何？

   (a) Model structure:

   LSTM(

   (embedding): Embedding(31889, 50)

   (lstm): LSTM(embedding_dim=50, hidden_layers=256, num_layers=4, batch_first=True)

   (classifier): Sequential(

   (0): Dropout(p=0.2, inplace=False)

   (1): Linear(in_features=256, out_features=1, bias=True)

   ))

   (b) Word embedding:

   Use the package function ***gensim.models.Word2Vec*** to transform all words to vectors, then make the embedding matrix by using those all vectors.

(c) Learning curve figure:

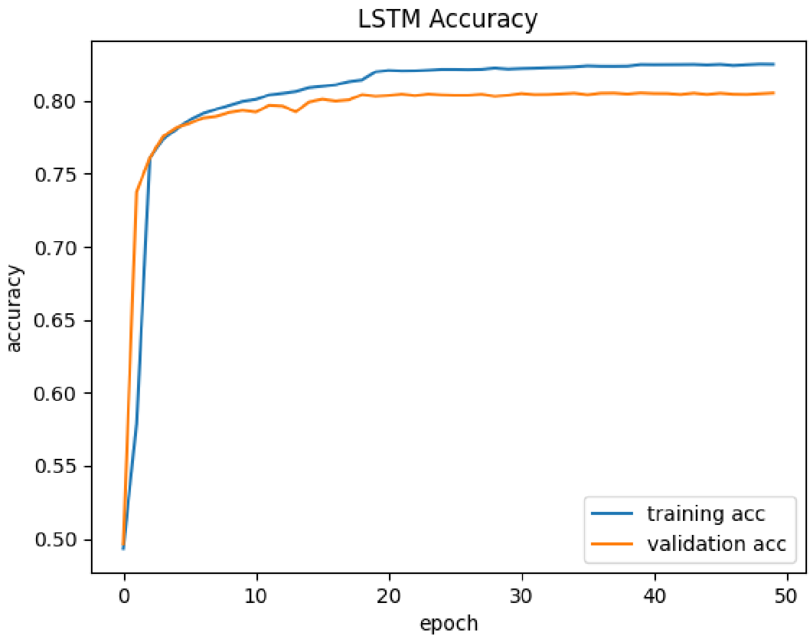

Figure 1: Accuracy curve

2. 請比較 ***BOW + DNN*** 與 ***RNN*** 兩種不同 ***model*** 對於 ***”today is a good day, but it is hot”*** 與 ***”today is hot, but it is a good day”*** 這兩句的分數 ***(***過 ***softmax*** 後的數值***)***，並討論造成差異的原因。

(a)

	today is a good day, but it is hot	today is hot, but it is a good day
GRU	0.607247	0.976884
BOW + DNN	0.766139	0.766139

Table 1: Results

(b) Discussion:

The probabilities of two sentences are different in GRU model and are the same in BOW + DNN model. Because BOW does only collect all words into the bag but disregarding grammar and even word order.

3. 請敘述你如何 ***improve performance***（***preprocess***、***embedding***、架構等等），並解釋為何這些做法可以使模型進步，並列出準確率與 ***improve*** 前的差異。

(a) Model structure:

GRUSentiment(

(embedding): Embedding(31889, 50)

(gru): GRU(embedding_dim=50, hidden_layers=512, num_layers=6, batch_first=True, dropout=0.2)

(classifier): Sequential(

(0): Linear(in_features=512, out_features=1, bias=True)

))

(b) Word embedding:

Use the package function ***gensim.models.Word2Vec*** to transform all words to vectors, then make the embedding matrix by using those all vectors.

(c) Results:

	Training accuracy	Training loss	Validation accuracy	Validation loss	Test accuracy
LSTM	0.825083	–	0.800862	–	–
GRU	0.842060	0.363514	0.821499	0.406715	0.82528

Table 2: Comparison table

(d) Discussion:

GRU forgets as well as input gates. GRU uses less training parameters and therefore use less memory, execute faster and train faster than LSTM, however, LSTM is more accurate on the dataset using longer sequence. For our dataset, there are collected many short sentences from Twitter, therefore, using GRU will be easier to adjust parameters to get better performance. In short, if the sequence is large or accuracy is very critical, please go for LSTM whereas for less memory consumption and the faster operation go for GRU.

4. 請描述你的 ***semi-supervised*** 方法是如何標記 ***label***，並比較有無 ***semi-supervised training*** 對準確率的影響並試著探討原因。

   (a) Methodology:

      i. We train three different GRU models and use them to predict the non-label training dataset.

      ii. Ensemble those three results to label the non-label training dataset, if there is a sentence marked as 1 more than twice, then this sentence will be labeled 1.

      iii. Combine this new labeled dataset with the original training dataset, then train the GRU model again.

   (b) Results:

	Training acc	Training loss	Validation acc	Validation loss	Test acc
LSTM	0.825083	–	0.800862	–	–
GRU	0.842060	0.363514	0.821499	0.406715	0.82327
GRU (semi-supervised)	0.897974	–	0.835606	–	0.82815

Table 3: Comparison table

   (c) Discussion: