CSCI544 Homework4

Name Entities Recognition (NER) by BiLSTM model

Requirement:

Task1: BiLSTM model for NER, some required hyperparameters in model, SGD optimizer
Task2: pretrained glove embedding, BiLSTM model for NER, some required hyperparameters in model, SGD optimizer

Step1 clean and preprocess data (implement in prepare_data.ipynb)
Since I need to use SequenceTagged.split() in torchtext, which is a built-in function to split data in format:
Word tag
W1 t1
W2 t2

W1. T1
W2. T2
W3. T3
To sentence which split by blank row. However, there are some word tag pairs that word in nan, and tag -<DOCSTART>- which is not a sentence.
I filled nan value word space and remove docstart tag, output as csv files.
Then remove sentence index column which is not used, output as tsv files.

Step2 Use Torchtext to build word to index map, and tag to index map and corpus for task1, using pre-trained glove model to build word to index map, and tag to index map. (encode) same batch size=16 for both tasks.

Step3 built train and validation(dev) iterators

Step4 Build biLSTM model
Embedding layer + dropout + bilstm + dropout + dropout + regression + elu activation (same structure for both tasks)
And also save model

Step5 Set optimizer and loss function
SGD and CrossEntropyLoss , same lr=0.01 for both tasks

Step6 Train, evaluation, predict
epoch=10 for both tasks.
Set epochs and train model in 10 epoches, and write function calculate loss(by loss function) and accuracy for train data and validation data.
Make prediction (decode from index to tag) for validation data and test data.
There are two kind of output file.
Example1: dev.out is a output file in format:
Sentence_idx. Word. Pred_tag
Example2: dev_eval.txt is use for perl conll03eval to calculate dev precision, recall and f1 score. Is a output txt file in format:
Sentence_idx word gold(true_tag). Pred_tag

Step7. Calculate dev precision, recall and f1 for both tasks.
Since I write code on google colab notebook, some drive path maybe cannot run on others computer. Output model and output file can used to test and evaluate.

dev precision, recall and f1 for task1 and task2

```
(base) MaggieTangdeMacBook-Pro:~ maggietang$ cd desktop/ColabNotebooks
(base) MaggieTangdeMacBook-Pro:ColabNotebooks maggietang$ perl conll03eval.pl < dev1_eval.txt
processed 51578 tokens with 5942 phrases; found: 3596 phrases; correct: 2117.
accuracy:  89.19%; precision:  58.87%; recall:  35.63%; FB1:  44.39
              LOC: precision:  81.80%; recall:  43.06%; FB1:  56.42  967
             MISC: precision:  60.05%; recall:  25.92%; FB1:  36.21  398
              ORG: precision:  44.97%; recall:  26.32%; FB1:  33.21  785
              PER: precision:  50.76%; recall:  39.85%; FB1:  44.65  1446
(base) MaggieTangdeMacBook-Pro:ColabNotebooks maggietang$ perl conll03eval.pl < dev2_eval.txt
processed 51578 tokens with 5942 phrases; found: 734 phrases; correct: 368.
accuracy:  84.32%; precision:  50.14%; recall:   6.19%; FB1:  11.02
              LOC: precision:  34.00%; recall:   0.93%; FB1:   1.80  50
             MISC: precision:   9.09%; recall:   0.33%; FB1:   0.63  33
              ORG: precision:  59.55%; recall:  17.67%; FB1:  27.26  398
              PER: precision:  43.87%; recall:   6.03%; FB1:  10.60  253
(base) MaggieTangdeMacBook-Pro:ColabNotebooks maggietang$
```

For dev in task1:
F1 score = 0.8919
precision score: 0.5887
recall score: 0.3563

For dev in task2:
F1 score = 0.8432
precision score: 0.5014
recall score: 0.06