

Homework 3: Neural Machine Translation (NMT)

***GPUs may be needed for speeding up the neural network training process in this homework.**

Description

In this homework you will practice how to implement neural machine translation (NMT) using Recurrent Neural Network (RNN) in Python with the TensorFlow framework. You can choose either Long Short-Term Memory (LSTM) or Gated Recurrent Units (GRU) to implement NMT. The goals of this homework are:

- To understand the steps to train/test the model for NMT.
- To understand and implement the RNN architecture using TensorFlow.

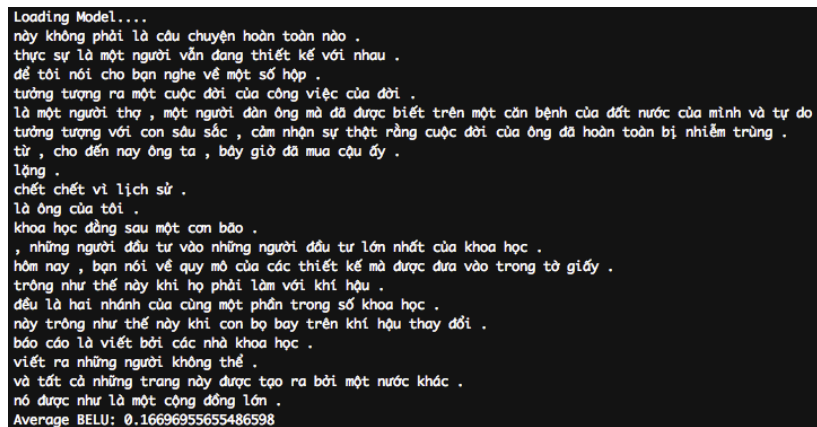
The English-Czech, English-German, and English-Vietnamese datasets can be found at: <https://nlp.stanford.edu/projects/nmt/> under Preprocessed Data.

Instruction

1. The neural machine translation file **NMT.py** should contain three functions, **train**, **test**, and **translate**. You can choose any one of the three datasets (English-Czech, English-German, and English-Vietnamese), based on your interests and the data size.
2. The **train** function would train the network with the command “**python NMT.py train**”. Display the training loss in **each iteration** of training function. Save the model in a folder named “**model**” after finishing the training process.

```
Processing Data
ENC_VOCAB: 41303
DEC_VOCAB: 18778
Bucket: [(19, 19), (28, 28), (33, 33), (40, 43), (50, 53), (60, 63)]
Number of samples in each bucket: [54053, 31229, 12026, 15393, 9186, 4990]
Bucket scale: [0.426026781843833, 0.6721628033449719, 0.7669475160982684, 0.8882697415607241, 0.9606705707102154, 1.0]
Loading Model....
Loss 0: 9.839266777038574
Loss 2000: 4.6640944488896
Loss 4000: 4.304739475250244
Loss 6000: 3.0162453651428223
Loss 8000: 2.8087968826293945
Loss 10000: 2.4592795372009277
Loss 12000: 2.1703057289123535
Loss 14000: 2.8176417350769043
Loss 16000: 3.131368637084961
Loss 18000: 2.736176013946533
Loss 20000: 1.8968796730041504
Loss 22000: 2.6051275730133057
Loss 24000: 2.3399462699890137
Loss 26000: 3.1411712169647217
Loss 28000: 2.223483085632324
Loss 30000: 1.698129653930664
Model saved in file: ./model/model.ckpt
```

Fig. 1 The screenshot of the train function (English-to-Vietnamese).



```

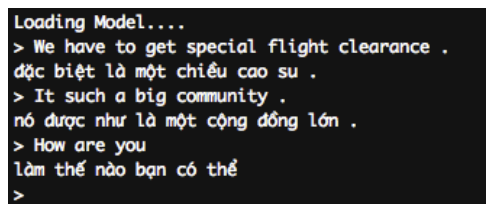
Loading Model....
này không phải là câu chuyện hoàn toàn nào .
thực sự là một người vẫn đang thiết kế với nhau .
để tôi nói cho bạn nghe về một số hộp .
tưởng tượng ra một cuộc đời của công việc của đời .
là một người thợ , một người dân ông mà đã được biết trên một căn bệnh của đất nước của mình và tự do .
tưởng tượng với con sâu sắc , cảm nhận sự thật rằng cuộc đời của ông đã hoàn toàn bị nhiễm trùng .
từ , cho đến nay ông ta , bây giờ đã mua cậu ấy .
lặng .
chết chết vì lịch sử .
là ông của tôi .
khoa học đằng sau một cơn bão .
, những người đầu tư vào những người đầu tư lớn nhất của khoa học .
hôm nay , bạn nói về quy mô của các thiết kế mà đã được đưa vào trong tờ giấy .
trông như thế này khi họ phải làm với khí hậu .
đều là hai nhánh của cùng một phần trong số khoa học .
này trông như thế này khi con bọ bay trên khí hậu thay đổi .
báo cáo là viết bởi các nhà khoa học .
viết ra những người không thể .
và tất cả những trang này được tạo ra bởi một nước khác .
nó được như là một cộng đồng lớn .
Average BLEU: 0.16696955655486598

```

Fig. 2 The screenshot of the test function (English-to-Vietnamese).

3. The **test** function would test the model with the command “**python NMT.py test**”, which will (1) load the testing data and translate the sentences; and (2) calculate the **BLEU** score (referring to https://www.nltk.org/_modules/nltk/translate/bleu_score.html) with the smoothing **method1**. Report the average BLEU score, which should be no less than **9% (0.09)**.

4. The **translation** function would accept the command “**python NMT.py translate**”. Given a sentence, the model should be able to translate it into the corresponding language.



```

Loading Model....
> We have to get special flight clearance .
đặc biệt là một chiều cao su .
> It such a big community .
nó được như là một cộng đồng lớn .
> How are you
làm thế nào bạn có thể
>

```

Fig. 3 The screenshot of the translation function (English-to-Vietnamese).

Submission

- You need to submit a **zip** file including:
 - a python file named “**NMT.py**”;
 - a generated model folder named “**model**”;
 - two screenshots of the **train** and **test** functions.
- The “**NMT.py**” file should be able to run with the following commands:


```
python NMT.py train
python NMT.py test
python NMT.py translate
```
- The **zip** file should be named using the following convention:


```
<Last-Name>_<First-Name>_HW3.zip
```

 Ex. Sun_Mengshu_HW3.zip

Note:

Don’t put any print function other than showing the results.

Comment your code.

Grading criteria:

- The two screenshots of the **train** and **test** functions will be checked, and the **translate** function will be tested by typing a sentence.
- The testing accuracy (BLEU score) should be greater than or equal to **9% (0.09)** in the end. There will be 1-point deduction for every 0.1% of accuracy degradation based on 9%.
- Upload the zip file to the Blackboard before 11:59PM (EST Time) 04/19/2018.

* Sample code can be found at: <https://github.com/chiphuyen/stanford-tensorflow-tutorials/tree/master/assignments/chatbot>