Implementation of a Chatbot using Jupyter Python

By Thawatchai Sangthep
P2681054
IMAT5118 - NLP

Overview

The project has a mission to improve the chatbot tutorial code by tuning and using other techniques to modify the result to show the accuracy in PyTorch. The chatbot code is set with a source that is cornell movie dataset with RNN parameter to tune and is trained with 4000 words that are generated.

Firstly, used the dataset and tune with RNN.
Secondly, split out 2 code files and create the specification.
Lastly, show the output of the product.

Word Embedding

Embedding is a table of the words that use the matrix to present the values of word to word. This is a tool to manage and analyze the words.

For example, if the words are closer in the vector space, That could be similar in meaning.

Hey & Hi & Hello || Subway & Underground, they are same meaning so the robot knew by technic Embedding.

Dataset

Cornell Movie Dialogue Dataset

It has a content of 4,000 words. it is limited, if we ask questions beyond 4,000 words. The robot shows the “unknown word encountered” answer. So, the dataset must be upgraded to increase intelligence.

I have trained the chatbot to iterate over 6000 times.

```
# Configure training/optimization
clip = 50.0
teacher_forcing_ratio = 1.0
learning_rate = 0.0001
decoder_learning_ratio = 5.0
n_iteration = 6000
print_every = 1
save_every = 500
```

Data Preparation

The data preparation is a step to prepare the dataset to tune.
We did follow the step.

1. Spitting each line from the file and adding dictionary to fields
2. Group them from Load-lines and add conversations base into the data “Movie_conversations.txt”
3. Extracting pairs of sentences from conversations

Result 1

The chatbot was trained with a lot of words.
The iteration spend time too long and got loss 2.1285

```
Iteration: 6000; Percent complete: 100.0%; Average loss: 2.1285
```

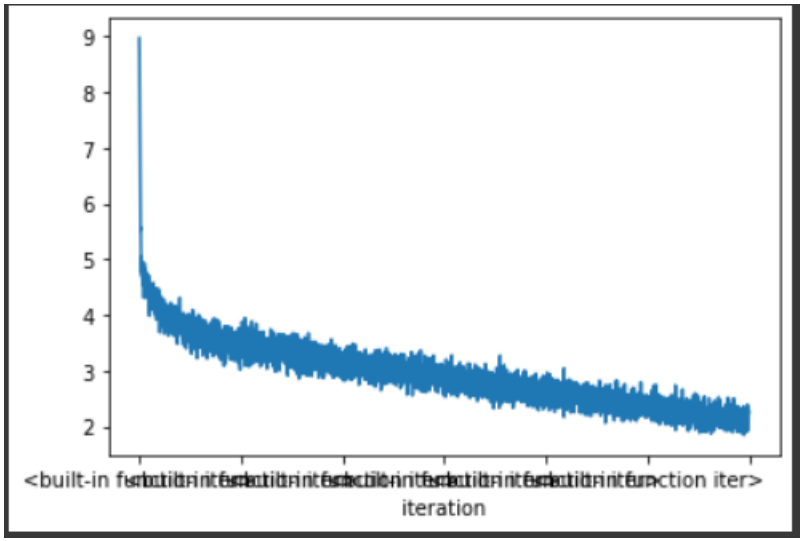
Result 2

The dataset was used, the chatbot has increased intelligence. The chatbot is responded.

```
> Hey
Bot: what ? ? ? ? ?
> How are you
Bot: fine . how i m doing
> What are you doing
Bot: i m sorry i m just dressed .
> Where are you going
Bot: i m going home . !
```

Loss graph

Loss graph has been plotted and shows the value that it has decreased average loss with every iteration. That means the chatbot has learned and is smarter than past. The graph shows the decreased average loss.



BLEU score

BLEU score is a tool to use for evaluating the machine-translated text.
The score is calculated from 0 to 1 based on the complexity of the text, We got a score of 0.02
If the score is nearly 1.0 that means a nearly perfect

```
/usr/local/lib/python3.7/dist-packages/nltk/translate/bleu_score.py:490: UserWarning:
Corpus/Sentence contains 0 counts of 2-gram overlaps.
BLEU scores might be undesirable; use SmoothingFunction().
  warnings.warn( msg)
0.0297941495124593 76
```

Thank you