



PROJECT

Translation From One Language to Another Language

A part of the Deep Learning Nanodegree Foundation Program

PROJECT REVIEW

CODE REVIEW

NOTES

SHARE YOUR ACCOMPLISHMENT!  

Requires Changes

1 SPECIFICATION REQUIRES CHANGES

Good job !

I gave you some advice to improve your project even better.

Advanced tips:

- if you don't already know that, here how we use to represent words:
<https://www.tensorflow.org/tutorials/word2vec>
Another ressource: <https://www.oreilly.com/learning/capturing-semantic-meanings-using-deep-learning>
- A good link to optimize hyperparameters: http://neupy.com/2016/12/17/hyperparameter_optimization_for_neural_networks.html
- The article explains Neural Network for Machine Translation that Google created : <https://research.googleblog.com/2016/09/a-neural-network-for-machine.html>
- More advanced about recurrent network, Attention and Augmented Recurrent Neural Networks:
<http://distill.pub/2016/augmented-rnns/>

Good luck !

Required Files and Tests

The project submission contains the project notebook, called "dLnd_language_translation.ipynb".

Perfect all files are here.

All the unit tests in project have passed.

Good, you passed all tests!

Preprocessing

The function `text_to_ids` is implemented correctly.

Good !

Here another approach

```
target_ids = [[target_vocab_to_int[w] for w in s.split()] + [target_vocab_to_int['<EOS>']] for s in target_text.split('\n')]
source_ids = [[source_vocab_to_int[w] for w in s.split()] for s in source_text.split('\n')]
```

Neural Network

The function `model_inputs` is implemented correctly.

The function `process_decoding_input` is implemented correctly.

The function `encoding_layer` is implemented correctly.

The function `decoding_layer_train` is implemented correctly.

The function `decoding_layer_infer` is implemented correctly.

The function `decoding_layer` is implemented correctly.

Good!

The function `seq2seq_model` is implemented correctly.

Good !

For the `f.random_uniform`, I would suggest to add a range of value for the initialization, as -1 to 1

Neural Network Training

The parameters are set to reasonable numbers.

Good first choice !

A range of value that is good for this project:

- `epochs = [5-15]` : More you have layers, more you should increase the number of epochs, the key is to choose a number such that the loss on validation set stops decreasing further.
- `batch_size = [256, 1024]` : mainly depend on where you will run the code, and it's power
- `rnn_size = [128, 512]` : High mean, the model will learn more complex structure
- `num_layers = [2, 4]` : More you have more your complex your model will be
- `embedding_size = [128, 256]` : Can represent the number of unique words we can deal with.
- `learning_rate = [0.001, 0.01]` : low to learn form the large variability of the dataset
- `keep_probability = [0.6, 0.9]` : Depend where you put dropout layer, but shouldn't be too low due to the small size of our dataset

Don't forget: the values of hyper parameters should be power of 2. Tensorflow optimizes our computation if we do so.

So you should mainly increase the `embed_size`

The project should end with a validation and test accuracy that is at least 90.00%

Good !

It would be even better with a graph:

```
# Visualize the loss and accuracy
import matplotlib.pyplot as plt
f, (ax1, ax2) = plt.subplots(1, 2, figsize=(18, 6))
ax1.plot(loss_list, color='red')
ax1.set_title('Traning Loss')
```

```
ax1.set_ylabel('Loss value')
ax2.plot(valid_acc_list)
ax2.set_xlabel('Iterations')
ax2.set_ylabel('Accuracy')
ax2.set_title('Validation Accuracy')
```

```
plt.show()
```

Language Translation

The function `sentence_to_seq` is implemented correctly.

Good!

The project gets majority of the translation correctly. The translation doesn't have to be perfect.

Will be better with previous fix.
For the default example, you should reach something like: 'Il a vu un vieux camion jaune'.
Your example should be : La voiture camion jaune et le cheval bleu.

 RESUBMIT

 [DOWNLOAD PROJECT](#)

Learn the [best practices for revising and resubmitting your project](#).

[RETURN TO PATH](#)

[Student FAQ](#)