UDACITY

‹ Return to "Artificial Intelligence Nanodegree and Specializations" in the classroom

DISCUSS ON STUDENT HUB

Machine Translation

REVIEW

CODE REVIEW

HISTORY

Meets Specifications

Awesome job working on the project.Shows you have good grasp of concepts and ability to translate into code.
Keep it up
It was a pleasure reviewing your project.
All the best for your future projects

Submitted Files

The following files have been submitted: `helper.py`, `machine_translation.ipynb`, `machine_translation.html`

Preprocess

The function `tokenize` returns tokenized input and the tokenized class.

Neat and clean implemenetation of tokenize function

The function `pad` returns padded input to the correct length.

The function `pad` returns padded input to the correct length.

Padding is correctly implemented
One tip, if you read the pad_sequences description
https://keras.io/preprocessing/sequence/#pad_sequences, you will see that the function correctly manages the situation where length = None

Models

The function `simple_model` builds a basic RNN model.

Correctly built a simple model.Well done.
Tip: Try experimenting with differenct architectures to understand their behaviour.You can try using LSTM cell instead of GRU, try a wide network by increasing the number of units to 256 or 512 or by changing various learning rate and see how training proceeds.This will provide you with some idea and intution who performs better and can be used further in complex models

You might find these links useful!

1) When to use GRU over LSTM https://datascience.stackexchange.com/questions/14581/when-to-use-gru-over-lstm
2) The Difference between Dense and TimeDistributedDense of Keras
https://datascience.stackexchange.com/questions/10836/the-difference-between-dense-and-timedistributeddense-of-keras

The function `embed_model` builds a RNN model using word embedding.

Great job building an embedded model.Great validation accuracy.

Tip: Embeddings are a great way to improve your models.If you want to learn more about it, you can refer to these links
https://developers.google.com/machine-learning/crash-course/embeddings/video-lecture
https://towardsdatascience.com/neural-network-embeddings-explained-4d028e6f0526
https://stats.stackexchange.com/questions/182775/what-is-an-embedding-layer-in-a-neural-network

The Embedding RNN is trained on the dataset. A prediction using the model on the training dataset is printed in the notebook.

```
Epoch 10/10
110288/110288 [==============================] - 28s 250us/step - loss: 0.2308 - acc: 0.9237 - val_loss: nan - val_acc: 0.9245
new jersey est parfois calme au l' automne et il est neigeux en avril <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
```

The function `bd_model` builds a bidirectional RNN model.

Correctly build a bidirectional model.Good job
Additional Reading:
1) Bidirectional LSTMs from scratch https://machinelearningmastery.com/develop-bidirectional-lstm-sequence-classification-python-keras/
2) A small chapter on Bidirectional RNNs from Dive into Deep Learning book.
https://d2l.ai/chapter_recurrent-modern/bi-rnn.html

The Bidirectional RNN is trained on the dataset. A prediction using the model on the training dataset is printed in the notebook.

```
Epoch 10/10
110288/110288 [==============================] - 38s 347us/step - loss: 0.7595 - acc: 0.7407 - val_loss: nan - val_acc: 0.7414
new jersey est parfois calme en mars et il est est en en <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
```

The function `model_final` builds and trains a model that incorporates embedding, and bidirectional RNN using the dataset.

Correctly built the final model.Well done

Prediction

The final model correctly predicts both sentences.

Great job implemeting the corrections suggested by the reviewer.
Your model predicts both the sentences correctly.Awesome

To improve the model you can further look at these steps
Tweak your learning rate.Start with a higher value and reduce down as training progresses

For more information check out the link
https://machinelearningmastery.com/using-learning-rate-schedules-deep-learning-models-python-keras/

DOWNLOAD PROJECT

RETURN TO PATH

Rate this review