DNN Speech Recognizer

REVIEW | CODE REVIEW | HISTORY

Meets Specifications

Impressive work!

You meet all the specifications!

I especially enjoyed seeing your predicted transcription which your Notebook printed in the last step "3: Obtain predictions"!

Keep up the good work and good luck with your Nanodegree!

STEP 2: Model 0: RNN

The submission trained the model for at least 20 epochs, and none of the loss values in `model_0.pickle` are undefined. The trained weights for the model specified in `simple_rnn_model` are stored in `model_0.h5`.

Your VUI Notebook shows that you trained the model for 20 epochs. All the loss values are numeric and not undefined. Validation loss ends at 727.2913

STEP 2: Model 1: RNN + TimeDistributed Dense

The submission includes a `sample_models.py` file with a completed `rnn_model` module containing the correct architecture.

The submission trained the model for at least 20 epochs, and none of the loss values in `model_1.pickle` are undefined. The trained weights for the model specified in `rnn_model` are stored in `model_1.h5`.

Good job! Your VUI Notebook shows that you trained your `rnn_model` model for 20 epochs. All the loss values are numeric and not undefined. Your validation loss ends at 210.2870

STEP 2: Model 2: CNN + RNN + TimeDistributed Dense

The submission includes a `sample_models.py` file with a completed `cnn_rnn_model` module containing the correct architecture.

The submission trained the model for at least 20 epochs, and none of the loss values in `model_2.pickle` are undefined. The trained weights for the model specified in `cnn_rnn_model` are stored in `model_2.h5`.

Well done! Your VUI Notebook shows that you trained `cnn_rnn_model` model for 20 epochs. All the loss values are numeric and not undefined. Your validation loss ends at 147.4331

STEP 2: Model 3: Deeper RNN + TimeDistributed Dense

The submission includes a `sample_models.py` file with a completed `deep_rnn_model` module containing the correct architecture.

The submission trained the model for at least 20 epochs, and none of the loss values in `model_3.pickle` are undefined. The trained weights for the model specified in `deep_rnn_model` are stored in `model_3.h5`.

Great job! Your VUI Notebook shows that you trained `deep_rnn_model` model for 20 epochs. All the loss values are numeric and not undefined. Your validation loss ends at 161.4216

STEP 2: Model 4: Bidirectional RNN + TimeDistributed Dense

The submission includes a `sample_models.py` file with a completed `bidirectional_rnn_model` module containing the correct architecture.

The submission trained the model for at least 20 epochs, and none of the loss values in `model_4.pickle` are undefined. The trained weights for the model specified in `bidirectional_rnn_model` are stored in `model_4.h5`.

Well done! Your VUI Notebook shows that you trained `bidirectional_rnn_model` model for 20 epochs. All the loss values are numeric and not undefined. Your validation loss ends at 210.8030

STEP 2: Compare the Models

The submission includes a detailed analysis of why different models might perform better than others.

In your answer to question 1 you analyzed different models and described why each of them might perform better than others. Excellent job here!

STEP 2: Final Model

The submission trained the model for at least 20 epochs, and none of the loss values in `model_end.pickle` are undefined. The trained weights for the model specified in `final_model` are stored in `model_end.h5`.

Your VUI Notebook shows that you trained your final model for 20 epochs. All the loss values are numeric and not undefined. Your final validation loss ends at 73.9723

The submission includes a `sample_models.py` file with a completed `final_model` module containing a final architecture that is not identical to any of the previous architectures.

Your submission includes a sample_models.py file with a completed `final_model` module containing a final architecture. Good job!

The submission includes a detailed description of how the final model architecture was designed.

In your answer to question 2 you described your final model layers and gave your reasoning why you chose such an architecture.

DOWNLOAD PROJECT