Documentation

prepare\_dataset.py:

*Prepares the data set for the LSTM model***:param** *seq\_length: sequence length (size of sentence)***:param** *txt: text file***:param** *n\_chars: numbe of chars in text file***:param** *char\_to\_int: chars to int map***:param** *n\_vocab: number of unique chars in text file***:return***:*

prepare\_dataset\_test.py:

*tests the prepare\_dataset model***:return***:*

text\_generator.py: argv[1] – text\_path , argv[2] – weight\_file\_path

The main file which will run the LSTM prediction after the weights have been generated.

*Receives 2 parameters - the text to use as a starting point, and the weights file***:return***: prints out 1000 characters based on the training and the chosen seed*

text\_parser.py :

*Parses the given text and provides a dictionary to convert char to int and int to char***:param** *txt: text file***:return***: dictionary containing arrays to convert between char to int and int to char, as well as the number of total characters (n\_chars) and number of different characters (n\_vocab)*

text\_parser\_test.py: argv[1] – text\_path

*Verifies that the text parser creates the conversion dictionary, receives the script and the text path as parameters***:return***:*

weights\_calculator.py : argv[1] – text\_path

*Generates the weights file based on the given text***:return***:* saves weight file to disk