DL4H Project Code Instructions

Aaron Shipway

UIUC CS 598 DL4H

Spring 2022

Environment

Windows 10

Python 3.10

Coding was done in Jupyter Notebooks and the code is provided in the original notebooks. There is one notebook for each dataset run through the model. The notebook includes all code necessary to load, clean, process and model each raw dataset. I used the latest version of Jupyter Notebook. Download the latest version CoreNLP from https://stanfordnlp.github.io/CoreNLP/ and unzip the folder in a convenient location. The folder will be accessed each time CoreNLP is used for sentence parsing.

I recommend setting up a virtual environment to run the project code in. Once the virtual environment is active, install the latest versions of Numpy, NLTK, Gensim Word2Vec, Pytorch, and Sklearn in the environment. You are now ready to run the project code notebooks.

Code and Raw Data Files

15 datasets were run through the CNN model. There is a Jupyter Notebook for each dataset. The folder containing the notebooks also contains the raw data files. Each notebook contains the code required to load the raw data from the folder. No rearranging of the locations of data files is required. The following table lists names of the datasets used in the paper and the corresponding notebook file names and raw data file names.

Data Set	Notebook File	Raw Data Files
MSRP SPO	MSRP_Train_Test_SPO	msr_paraphrase_train.txt, msr_paraphrase_test.txt
MSRP RW	MSRP_Train_Test_RW	msr_paraphrase_train.txt, msr_paraphrase_test.txt
MSRP RSW	MSRP_Train_Test_RSW	msr_paraphrase_train.txt, msr_paraphrase_test.txt
MSRP RAD	MSRP_Train_Test_Rad	msr_paraphrase_train.txt, msr_paraphrase_test.txt
2012 MSRvid	STS-2012_MSRvid	STS.input.MSRvid.txt, STS.gs.MSRvid.txt

Data Set	Notebook File	Raw Data Files
2012 OnWN	STS-2012_OnWN	STS.input.surprise.OnWN, STS.gs.surprise.OnWN.txt
2012 SMTeuroparl	STS-2012_SMTeuroparl	STS.input.SMTeuroparl.txt, STS.gs.SMTeuroparl.txt
2013 OnWN	STS-2013_OnWN	STS.input.OnWN.tx, STS.gs.OnWN.txt
2013 headlines	STS-2013_headlines	STS.input.headlines.txt, STS.gs.headlines.txt
2014 deft-news	STS-2014_Deft_News	STS.input.deft-news.txt, STS.gs.deft-news.txt
2014 images	STS-2014_Images	STS.input.images.txt, STS.gs.images.txt
2014 OnWN	STS-2014_OnWN	STS.input.OnWN.txt, STS.gs.OnWN.txt
2015 answer-forums	STS-2015_Answer_Forums	STS.input.answers-forums.txt, STS.gs.answers-forums.txt
2015 answer-students	STS-2015_Answer_Students	STS.input.answers-students.txt, STS.gs.answers-students.txt
2015 images	STS-2015_Images	STS.input.images.txt, STS.gs.images.txt

Running the Code

Running the code is simply a matter of opening the Jupyter Notebook of the desired dataset and then running the notebook cells in order. Once you get to the step requiring sentence parsing with CoreNLP, copy the saved sentence textfile pair to the CoreNLP folder, open a command prompt and navigate to the CoreNLP folder then run the command:

```
java -cp "*" -Xmx8g edu.stanford.nlp.pipeline.StanfordCoreNLP -annotators
"tokenize,ssplit,pos,lemma,ner" -tokenize.whitespace=true -ssplit.eolonly -file
<input_filename.txt> -outputFormat json
```

After CoreNLP has completed the process the output JSON files will be in the CoreNLP folder. Copy the output files to the code notebook folder, change the file extension to .json, then proceed executing the remaining code cells in the notebook. Once the final results are generated, compare the values to what was reported in the project report for verification.