CSI5386, Winter 2020

Assignment 2
Due Mar 20, 10pm

**Text Entailment and Semantic Relatedness [100 points]**

**Note**: **You will work in groups of two students.**

In this assignment, you will classify them as the first sentence (the premise) entailing the second sentence (the hypothesis), contradicting it, or bearing no relation. Therefore the three classes will be: ENTAILEMNT, CONTRADICTION, and NEUTRAL. An entailment relation exists if the premise sentence can cover the meaning of the hypothesis sentence. Read more about the two tasks at http://alt.qcri.org/semeval2014/task1/
You will also calculate the semantic relatedness of pairs of sentences.

You will use the SICK dataset, which consists of 10,000 pairs of sentences annotated for semantic relatedness and entailment. The SICK data set was built starting from two existing paraphrase sets: the 8,000 ImageFlickr data set (http://nlp.cs.illinois.edu/HockenmaierGroup/data.html)
and the SEMEVAL-2012 Semantic Textual Similarity Video Descriptions data set
(http://www.cs.york.ac.uk/semeval-2012/task6/index.php?id=data).

Each sentence pair is annotated
for relatedness in meaning and for the entailment relation between the two elements.
The files are tab-separated text file with the following fields:
- sentence pair ID
- sentence A (premise)
- sentence B (hypothesis)
- semantic relatedness gold label (on a 1-5 continuous scale)
- textual entailment gold label (NEUTRAL, ENTAILMENT, or CONTRADICTION)

For example:

```
77    People wearing costumes are gathering in a forest and are looking in the
same direction Masked people are looking in the same direction in a forest
4.4   ENTAILMENT
85    People wearing costumes are gathering in a forest and are looking in the
same direction A little girl in costume looks like a woman      2     NEUTRAL
88    There is no biker jumping in the air   A lone biker is jumping in the air
4.2   CONTRADICTION
```

You will use Machine Learning (ML) algorithms in order to train classifiers for both tasks. I recommend using deep learning classifiers from packages such as TensorFlow, Theano, Keras, Torch, or others. In addition you can use any other ML method, for comparison. Feel free to use any tools and libraries, but you should cite them and explain how you used them.

The training data is available here. Use it for training classifiers.
Some trial data is available here. Use as development data to tune the parameters for deep learning. The test data is available here. Use it to produce the Results.txt file. Here is a version of the test data with expected solution. You can use it to compute evaluation measures for your report. You can write your own evaluation script or use the provided evaluation script (in R).

Perform the following experiments:

**1. [30 marks]** Task 1: Text entailment.
Train a deep learning (DL) classifier to solve this task. Here is a starter code in Theano that you can use, unless you prefer to write your own in another DL platform. Here is a diagram of the starter code. Here is another

possible starter code, in TensorFlow, with good explanations. It uses a different set of training data, so please train on the SICK dataset (or on any training data), but submit results on the provided SICK test data.
For comparison to a simple baseline, a very simple baseline classifier computed with the script compute_overlap_baseline.py from the SemEval webpage achieves an accuracy of 56.15%, as computed with the R evaluation script. (This baseline method choses the class ENTAILMENT if the overlap score between the two sentences is higher than a threshold, the class CONTRADICTION if it is lower than another threshold, and the class NEUTRAL if it is in between the two thresholds).
The main evaluation measure will be the classification accuracy on the test data. Other measures you can look at are the confusion matrices, as well as the Precision, Recall, and F-measure for each class.

**2. [30 marks]** Task 2: Semantic relatedness.
Train a deep learning classifier to solve this task. You can modify your code for task 1.
For comparison to a simple baseline, a very simple baseline classifier that computes the overlap between the two sentences leads to a Pearson correlation of 0.62. These baseline overlap scores were produced with the script compute_overlap_baseline.py and the correlation was computed with the R evaluation script.
The main evaluation measure will be the Pearson correlation between your scores for the test data and the gold standard ratings. Additional evaluation can be the mean squared error (computed on standardized scores) and the Spearman correlation.

**[20 marks] Write a report in a file Report (.pdf, .doc, or .txt)**
Explain what you did for task 1 and for task 2.
For task 1, report the accuracy of the classification on the test set for all the experiments that you ran.
For task 2, report Pearson correlation scores.
Discuss what classifier and what features led to your best results.

**[20 marks] Resulst.txt**
Submit the predictions of your best classifier on the test data in a file named Results.txt
Your file must contain the following 3 tab-delimited columns:
-pair_ID (the ids from the test data),
- entailment_judgment (predictions of your system for task 1; possible values: ENTAILMENT, CONTRADICTION, NEUTRAL) and
- relatedness_score (numerical predictions of your system for task 2).

**Submission instructions**:
- Submit your report and your best results for each sentence in a file Results.txt:

In the report include:
- * the names and student numbers of the students in the group, and specify how the tasks were divided,
- * explain what methods and ML algorithms you tried and what data representations (features) you used
- * discuss what methods led to the best results
- * a detailed note about the functionality of your programs
- * complete instructions on how to run them

- Submit your assignment as a zip file, including programs, Report file, and the Result.txt file through the Virtual Campus or by email. Only one partner in a team needs to submit.

**Have fun!!!**