**Project 23: Semantic Similarity 2**

We aim in this project to study a new semantic similarity for two short text document

1. Consider two sentences S1 and S2 which are tokenized for instance, say S1=(w1, w2,..,wn) and S2= (p1, p2, …, pm). Consider the approach implemented in Lab2 for calculating the semantic similarity between sentences using WordNet semantic similarity. We want to test this strategy on publicly available sentence database. For this purpose, use STSS-131 dataset, available in “A new benchmark dataset with production methodology for short text semantic similarity algorithms” by O’Shea, Bandar and Crockett (ACM Trans. on Speech and Language Processing, 10, 2013). Use Pearson correlation coefficient to test the correlation of your result with the provided human judgment.
2. We want to introduce another modification on this semantic similarity. For this purpose, suggest a script that proceeds in the following. For sentence S1 and S2, first, use Spacy to identify part of speech tag of each token of the sentences and all named-entities in S1 and S2. We want to discard those named-entities from the calculus of sentence to sentence semantic similarity. Next, use WordNet derivationally related form to turn each verb, adjective /adverb token into its corresponding noun token. Then calculate the sentence-to-sentence semantic similarity as the average of Wu-Palmer semantic similarity calculated over all pairs (N1i, N2j) where N1i and N2j stand for ith and jth noun-token of S1 and S2, respectively. Report the results of this semantic similarity on STSS-131 dataset and the Pearson correlation with human judgment. Comment on the overall process whether it yields improvement and its limitations.
3. Now we want to implement a new semantic similarity-based measure. The idea is to use some hierarchical reasoning and explore the WordNet Hierarchy. For this purpose, proceed in the following. For each sentence, use the parser tree to identify various part-of-speech of individual token of the sentence. Generate the list of hypernyms H1 and hyponyms H2 of each noun of the sentence. Repeat this process for each verb. Compute the semantic similarity between the two sentences as

Implement the above similarity expression in your python code

1. Test the above similarity on STSS-131 database and report the Pearson correlation with human judgments.
2. Study another text similarity using both wordnet semantic similarity and string similarity provided in <https://github.com/pritishyuvraj/SOC-PMI-Short-Text-Similarity->. Check the behavior of program for some intuitive sentences (very similar sentences, ambiguous and very dissimilar ones)
3. Report the result of the above similarity on STSS-131 and report the corresponding Pearson correlation with human judgments
4. Suggest an interface of your choice that would allow the user to input a textual query in the form of a pair of sentences and output the similarity score according to the various methods described above.