Aaron Goyzueta
Language Technology
MP1

Part 1

Results:

Method	Correlation	Coverage (out of 203)
Path	0.5734	203
Resnik	0.5713	192
WUP	0.6152	203
LCH	0.5914	203
JCN	0.5512	202
LIN	0.5389	192

The code I used to get these results are in the file ‘correlation_and_coverage.py’. I created a dictionary that would hold the similarity scores for every method for each pair, as well as the human judgement scores, and then turned that dictionary into a pandas dataframe. Using the data frame, I found the correlation between the column that held the human judgements (which I called the ‘gold’) and the scores from every method. I used a dictionary to avoid the problem of some methods having NaN and being skipped.

Part 2

	Correlation	Coverage
PPMI	0.1204	85

The code I used is also in ‘correlation_and_coverage.py’. After running the provided PPMI script on the text, I created a dictionary that held every word pair as the key and resulting similarity as the value, then looped through every word pair in the human judgements data and if the pair existed in the ppmi data, then added a entry into a new dictionary where the word pain was matched with a tuple containing the judgements and ppm similarity. If there was not, then it matched the gold and None together.

Part 3

	Correlation	Coverage
Word2Vec	0.6599	203

My approach to this part was basically the same, and in fact I used the same exact code that I can use for both files depending on the command line argument passed in.

Part 4

Overall Word2Vec seems to have the best performance with the highest correlation at 0.6599, as well as having perfect coverage (203/203), beating out the first 6 methods in part 1. The part 1 methods all were fairly similar in correlation and varied slightly with coverage, but never dropping below 90% word coverage. PPMI seems to perform the worst at 0.1204 correlation and a coverage of 85/203, which is not even 50%.