HW6

Mike Roylance
roylance@uw.edu

Describe and discuss your work in a write-up file

I completed this assignment using Python with nltk. My solution is organized into the following files and directories:

Location Description	Purpose
docs/	folder that contains all the documents needed for this assignment, such as the brown corpus (under docs/brown/), the ic-brown-resnik-add1.dat file (although this is under the patas environment as well), and wsd_contexts.txt.
docs/wsd_contexts.txt	file that contains all the words and their associated contexts.
docs/brown	the brown corpus
docs/ic-brown-resnik-add1.dat	resnik information content file
source/	folder that contains all the source code
source/main.py	entry point script that reads in the files from the user and prints out the result to the console. also processes extra credit ic file and result
source/tests.py	a few tests around functionality, I didn't need to write as many this time for this assignment
source/resnik.py	class that takes in an information content structure and processes each line from the wsd_contexts.txt file. this will find the resnik similarity for each probe/context pair and choose the sense after all the comparisons are made. the reznik method contains the logic to handle the similarity between two words.
source/utils.py	wrote my own function to build the ic function here. Modified original source code from here: http://nltk.googlecode.com/svn-hist/trunk/doc/api/nltk.corpus.reader.wordnet-pysrc.html#
source/icGenerator.py	class to generate the icfile from a corpus given the words that will be seen from the wsd_context.txt file. this is looking for par of speech tags specific to the brown corpus.

source/relationalWordBuilder.py	takes in the wsd_contexts.txt file and returns a list of all the word senses (and their lemmas) that could appear. this is used in the icGenerator
hw6.cmd	command file used by condor. this calls hw6.sh with the parameters of ic-brown-resnik-add1.dat docs/wsd_contexts.txt docs/brown extraCredit-ic-brown.dat extraCreditResults results
hw6.sh	file to handle calling the Python file main.py with specific parameters

Include problems you came across and how (or if) you were able to solve them, any insights, special features, and what you learned. Give examples if possible.

I read through Resnik's paper a few times, specifically section 5.1 and the references he made to the similarity function (WSIM) between two words. Once I got my head wrapped around how the senses referred to the wordnet synsets and the hypernyms were a way of finding the most informative subsumer, I felt it was straight forward.

Here are my results for main requirements:

Result	**Gold**	**IsMatch (Yes,No)**
necktie.n.01	necktie.n.01	Yes
suit.n.01	suit.n.01	Yes
head.n.11	head.n.01	Yes (GoPost)
hand.n.09	hand.n.01	Yes (GoPost)
buttocks.n.01	buttocks.n.01	Yes
doctor.n.01	doctor.n.01	Yes
lawyer.n.01	lawyer.n.01	Yes
lookout.n.01	lookout.n.01	Yes
line.n.05	line.n.05	Yes
plot.n.03	plot.n.03	Yes
wrinkle.n.01	line.n.05	No

line.n.05	line.n.05	Yes
line.n.23	line.n.22	No
wrinkle.n.01	line.n.18	No
line.n.05	line.n.05	Yes
cable.n.02	telephone_line.n.02	No
pipeline.n.02	telephone_line.n.02	No
line.n.22	line.n.22	Yes
		13 / 18 - **72.22%**

I also implemented extra credit for this assignment. I created my custom ic file from the brown corpus, also included in the tar submission. I printed out the custom ic file with the name of “new-ic-brown.dat”. I implemented my own version of the ic method to generate the data structure associated with the file. I picked the words I would use based on the wsd_contexts.txt file (and all the senses/lemmas with the probes and words). I picked the probabilities from the brown corpus by cycling through each file and examining each word/pos pair. If the pair was a noun of some sort, and the word was included in the list of words I had chosen, I would keep count. I would also keep count of all the total words I found.

Using the custom ic file with the same resnik class, I achieved almost similar results for the first 8 words, but significantly worse results for the remaining.

Result	**Gold**	**IsMatch (Yes,No)**
necktie.n.01	necktie.n.01	Yes
suit.n.01	suit.n.01	Yes
head.n.01	head.n.01	Yes
hand.n.01	hand.n.01	Yes
seat.n.01	buttocks.n.01	No
doctor.n.01	doctor.n.01	Yes
lawyer.n.01	lawyer.n.01	Yes
lookout.n.01	lookout.n.01	Yes

line.n.01	line.n.05	No
plot.n.01	plot.n.03	No
line.n.01	line.n.05	No
line.n.01	line.n.05	No
line.n.01	line.n.22	No
line.n.01	line.n.18	No
line.n.01	line.n.05	No
ine.n.06	telephone_line.n.02	No
line.n.06	telephone_line.n.02	No
line.n.01	line.n.22	No
		7 / 18 - **38.89%**

I think my problem with this custom ic file is that I’m not accounting for the different senses correctly in the corpus. I need to determine a better techniques to find out when line is a different sense, so I can assign it the proper count. However, this is the same problem that I am attempting to solve with this homework assignment, so I need to come up with a better strategy around it for the future.

Overall I enjoyed this assignment. I look forward to using this with some personal projects I have.