HW #8

Implementation

- Implement a simplified version of Resnik's "Associating Word Senses with Noun Groupings"
- Select a sense for the probe word, given group
  - Rather than all words as in the algorithm in the paper
- For each pair (probe, $noun_i$)
  - Loop over sense pairs to find MIS (Most informative sense), similarity value *v*
  - Update each sense of probe descended from MIS, with *v*
- Select highest scoring sense of probe
- Repeat noun-pair correlation with Resnik similarity

Algorithm

for i=1 to n Given W={w_1,…,w_n}, a set of nouns, ***and input word w_0***

$v_{0,i}$=wsim(w_0,w_i)

$c_{0,i}$=the most informative subsumer for w_0 and w_i

for k'=1 to **num_senses**(w_0)

if $c_{0,i}$ is an ancestor of $sense_{k'}$

increment_support[0,k'] by $v_{0,i}$

Return the sense_k with highest support

Components

- Similarity measure:
  - IC:
    - `/corpora/nltk/nltk-data/corpora/wordnet_ic/ic-brown-resnik-add1.dat`
  - NLTK accessor:
    - `wnic = nltk.corpus.wordnet_ic.ic('ic-brown-resnik-add1.dat')`
  - Note: Uses WordNet 3.0

Components

```
>>> from nltk.corpus import word net, wordnet_ic, information_content
>>> brown_ic = wordnet_ic.ic('ic-brown-resnik-add1.dat')
>>> wordnet.synsets('artifact')
[Synset('artifact.n.01')]

>>> wordnet.synsets(‘artifact’)[0].name
 ‘artifact.n.01’

>>> artifact = wordnet.synset('artifact.n.01’)
from nltk.corpus.reader.wordnet import information_content

>>> information_content(artifact, brown_ic)
2.4369607933293391
```

Components

- Hypernyms:

```
>>>wn.synsets('artifact')[0].hypernyms()
[Synset('whole.n.02')]
```

- Common hypernyms:

```
>>> hat = wn.synsets('hat')[0]
>>> glove = wn.synsets('glove')[0]
>>> hat.common_hypernyms(glove)
[Synset('object.n.01'), Synset('artifact.n.01'),
Synset('whole.n.02'), Synset('physical_entity.n.01'),
Synset('entity.n.01')]
```

Components

- WordNet API in NLTK:
- http://www.nltk.org/howto/wordnet.html
- http://www.nltk.org/api/nltk.corpus.reader.html#module-nltk.corpus.reader.wordnet

Note

- You can use supporting functionality, e.g.
  - `common_hypernyms`, `full_hypernyms`, etc
- You can NOT just use the built-in
  - `resnik_similarity`
  - `least_common_hypernym`, etc
- If unsure about acceptability, just ask!