

Quiz 6: Relationships between Words

Instructions: This is a file upload quiz. The quiz is open notes/computer. You need to copy/paste your code and output into this document then upload it.

The quiz uses NLTK and the VADER sentiment packages. You should watch video 07nlp_Relationships-between-words and code along before taking the quiz.

1. Print a hypernym of 'reptile.n.01'

```
>>> print('Hyponym: ', wn.synset('reptile.n.01').hypernyms())
```

Hyponym: [Synset('vertebrate.n.01')]

2. Print a hyponym of 'reptile.n.01'

```
>>> print('Hyponym: ', wn.synset('reptile.n.01').hyponyms()[0])
```

Hyponym: Synset('anapsid.n.01')

3. Output the path_similarity of 'shoot.v.01' and 'gun_down.v.01'

```
>>> shoot = wn.synset('shoot.v.01')
>>> gun_down = wn.synset('gun_down.v.01')
>>> print('Path Similarity: ', shoot.path_similarity(gun_down))
```

Path Similarity: 0.5

4. Output the Wu-Palmer similarity of 'shoot.v.01' and 'gun_down.v.01'

```
>>> print('Wu-Palmer Similarity: ', wn.wup_similarity(shoot, gun_down))
```

Wu-Palmer Similarity: 0.9090909090909091

5. Find a holonym of 'kitchen.n.01' and print the definition of that holonym

```
>>> print('Kitchen Holonym Definition: ',  
wn.synset('kitchen.n.01').part_holonyms()[0].definition())
```

Kitchen Holonym Definition: housing that someone is living in

6. Find entailments of synset 'snore.v.01' using the ".entailments()" method

```
>>> print(wn.synset('snore.v.01').entailments())
```

[Synset('sleep.v.01')]

7. Use morphy to find the root form of 'snoring'

```
>>> print(wn.morphy('snoring', wn.VERB))
```

snore

8. Using the Lesk algorithm in NLTK, find the most likely synset of 'arm' in the sentence below, and print the definition of that synset.

'The soldier was convicted of selling arms in the war'

```
>>> sent = ['The', 'soldier', 'was', 'convicted', 'of', 'selling', 'arms', 'in', 'the', 'war']  
>>> print(lesk(sent, 'arms'))
```

Synset('sleeve.n.01')

9. Using VADER sentiment analysis, print the polarity of the statement above.

```
>>> sentence = 'The soldier was convicted of selling arms in the war'  
>>> vs = SentimentIntensityAnalyzer().polarity_scores(sentence)  
>>> print(sentence, '\n\t', str(vs))
```

**The soldier was convicted of selling arms in the war
{'neg': 0.302, 'neu': 0.698, 'pos': 0.0, 'compound': -0.5994}**

10. Using NLTK Text object text6 Monty Python, calculate the pmi of 'fire arrows'. Is this likely to be a collocation?

```
>>> text = ' '.join(text6.tokens)
>>> voc = len(set(text6))
>>> ab = text.count('fire arrows') / voc
>>> a = text.count('fire') / voc
>>> b = text.count('arrows') / voc
>>> pmi = math.log2(ab / (a * b))
>>> print('PMI: ', pmi)
```

PMI: 10.080817527608327