WordNet is a lexical database of semantic relations between words in more than 200 languages. WordNet links words into semantic relations including synonyms, hyponyms, and meronyms. The synonyms are grouped into synsets with short definitions and usage examples

In [ ]:

```
import nltk
from nltk.corpus import wordnet as wn
  #output all synsets for noun: exercise
syn = wn.synsets('exercise')
print(syn)
```

```
[Synset('exercise.n.01'), Synset('use.n.01'), Synset('exercise.n.03'), Synset('exerci
se.n.04'), Synset('exercise.n.05'), Synset('exert.v.01'), Synset('practice.v.01'), Sy
nset('exercise.v.03'), Synset('exercise.v.04'), Synset('drill.v.03')]
```

Answers

Nouns seem to all be rooted in entity.n.01. It seems that you start at the top level of *thing* and then go from the mind to the body in terms of hierarchy.

In [ ]:

```
import nltk
from nltk.corpus import wordnet as wn
syn = wn.synsets('exercise')

workingSyn = syn[0]

print(workingSyn.definition())
print(workingSyn.examples())
print(workingSyn.lemmas())

print(workingSyn.hypernym_paths())
```

```
the activity of exerting your muscles in various ways to keep fit
['the doctor recommended regular exercise', 'he did some exercising', 'the physical e
xertion required by his work kept him fit']
[Lemma('exercise.n.01.exercise'), Lemma('exercise.n.01.exercising'), Lemma('exercise.
n.01.physical_exercise'), Lemma('exercise.n.01.physical_exertion'), Lemma('exercise.
n.01.workout')]
[[Synset('entity.n.01'), Synset('abstraction.n.06'), Synset('psychological_feature.n.
01'), Synset('event.n.01'), Synset('act.n.02'), Synset('activity.n.01'), Synset('wor
k.n.01'), Synset('labor.n.02'), Synset('effort.n.02'), Synset('exercise.n.01')]]
```

In [ ]:

```
import nltk
from nltk.corpus import wordnet as wn
syn = wn.synsets('exercise')

workingSyn = syn[0]

print("Hypernyms: ", workingSyn.hypernyms())
print("Hyponyms: ", workingSyn.hyponyms())
print("Meronyms: ", workingSyn.member_meronyms())
print("Holonyms: ", workingSyn.member_holonyms())
print("Antonyms: ", workingSyn.lemmas()[0].antonyms())
```

```
Hypernyms:  [Synset('effort.n.02')]
Hyponyms:  [Synset('arm_exercise.n.01'), Synset('back_exercise.n.01'), Synset('bodybu
ilding.n.01'), Synset('calisthenics.n.02'), Synset('cardiopulmonary_exercise.n.01'),
Synset('gymnastic_exercise.n.01'), Synset('isometrics.n.01'), Synset('isotonic_exerci
se.n.01'), Synset('kegel_exercises.n.01'), Synset('kick_up.n.01'), Synset('leg_exerci
se.n.01'), Synset('neck_exercise.n.01'), Synset('set.n.03'), Synset('stomach_exercis
e.n.01'), Synset('stretch.n.04'), Synset('yoga.n.02')]
Meronyms:  []
Holonyms:  []
Antonyms:  []
```

In [ ]:

```
import nltk
from nltk.corpus import wordnet as wn
  #output all synsets for noun: exercise
syn = wn.synsets('working')
print(syn)
```

```
[Synset('working.n.01'), Synset('work.v.01'), Synset('work.v.02'), Synset('work.v.0
3'), Synset('function.v.01'), Synset('work.v.05'), Synset('exercise.v.03'), Synset('m
ake.v.36'), Synset('work.v.08'), Synset('work.v.09'), Synset('work.v.10'), Synset('br
ing.v.03'), Synset('work.v.12'), Synset('cultivate.v.02'), Synset('work.v.14'), Synse
t('influence.v.01'), Synset('work.v.16'), Synset('work.v.17'), Synset('work.v.18'), S
ynset('work.v.19'), Synset('shape.v.02'), Synset('work.v.21'), Synset('knead.v.01'),
Synset('exploit.v.01'), Synset('solve.v.01'), Synset('ferment.v.03'), Synset('sour.v.
01'), Synset('work.v.27'), Synset('working.s.01'), Synset('working.s.02'), Synset('wo
rking.s.03'), Synset('running.s.06'), Synset('working.s.05')]
```

Answers

For Nouns every noun ended at the root of entity.n.01. However this is not the case for verbs.

In [ ]:

```
workingSyn = syn[1]

print(workingSyn.definition())
print(workingSyn.examples())
print(workingSyn.lemmas())

print(workingSyn.hypernym_paths())
```

```
exert oneself by doing mental or physical work for a purpose or out of necessity
['I will work hard to improve my grades', 'she worked hard for better living conditio
ns for the poor']
[Lemma('work.v.01.work')]
[[Synset('work.v.01')]]
```

In [ ]:

```
wn.morphy('working', wn.VERB)
```

Out[ ]:

```
'work'
```

Answers

It would seem potato and tomato are particularly related. Wu-Palmer returns a .8 which is quite high considering identity is only .2 away.

In [ ]:

```
from nltk.wsd import lesk
potato = wn.synset('potato.n.02')
print(potato.definition().split())
tomato = wn.synset('tomato.n.02')
print(tomato)

print("Wu-Palmer: ", wn.wup_similarity(potato, tomato))
print("Lesk: ", lesk(potato.definition().split(), 'tomato'))
```

```
['annual', 'native', 'to', 'South', 'America', 'having', 'underground', 'stolons', 'b
earing', 'edible', 'starchy', 'tubers;', 'widely', 'cultivated', 'as', 'a', 'garden',
'vegetable;', 'vines', 'are', 'poisonous']
Synset('tomato.n.02')
Wu-Palmer:  0.8
Lesk:  Synset('tomato.n.02')
```

Answers

These scores help us understand the general mood of the word that we're looking at. This is very helpfull in eventually getting the mood a corpus.

In [ ]:

```
from nltk.corpus import sentiwordnet as swn

angSyn = wn.synsets('anguish')[0]
print(angSyn)
anguish = swn.senti_synset('anguish.n.01')
print(anguish)
print("Positive score = ", anguish.pos_score())
print("Negative score = ", anguish.neg_score())
print("Objective score = ", anguish.obj_score())
```

```
Synset('anguish.n.01')
<anguish.n.01: PosScore=0.0 NegScore=0.625>
Positive score =  0.0
Negative score =  0.625
Objective score =  0.375
```

Answers

Collocations are common groups of words that create the phrases we use in everyday conversation. These usually hold more meaning than is present at first glance.

With the mutual information we can see that United States has a decently high mutual information, this means that most of the time when you see United it will be followed by States.

In [ ]:

```
from nltk.book import text4
import math

text4.collocations()
```

```
vocab = len(set(text4))
text = ' '.join(text4.tokens)

hg = text.count('United States')/vocab
print("p(United States) = ",hg )
h = text.count('United')/vocab
print("p(United) = ", h)
g = text.count('States')/vocab
print('p(States) = ', g)
pmi = math.log2(hg / (h * g))
print('pmi = ', pmi)
```

```
United States; fellow citizens; years ago; four years; Federal
Government; General Government; American people; Vice President; God
bless; Chief Justice; one another; fellow Americans; Old World;
Almighty God; Fellow citizens; Chief Magistrate; every citizen; Indian
tribes; public debt; foreign nations
p(United States) =  0.015860349127182045
p(United) =  0.0170573566084788
p(States) =  0.03301745635910224
pmi =  4.815657649820885
```