Lab14. Word Sense Disambiguation with Improved Lesk Algorithm

DINESH KUMAR K_225229108

In [ ]:
```
#EXERCISE-1
```

In [1]:
```
import nltk
from nltk.wsd import lesk
from nltk.corpus import wordnet as wn
nltk.download('wordnet')
```

```
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\sweth\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

Out[1]:
```
True
```

In [5]:
```
import nltk
nltk.download('omw-1.4')
```

```
[nltk_data] Downloading package omw-1.4 to
[nltk_data]     C:\Users\sweth\AppData\Roaming\nltk_data...
```

Out[5]:
```
True
```

In [6]:
```
for ss in wn.synsets('bass'):
    print(ss,ss.definition())
```

```
Synset('bass.n.01') the lowest part of the musical range
Synset('bass.n.02') the lowest part in polyphonic music
Synset('bass.n.03') an adult male singer with the lowest voice
Synset('sea_bass.n.01') the lean flesh of a saltwater fish of the family Serranidae
Synset('freshwater_bass.n.01') any of various North American freshwater fish with lean flesh (especially of the genus Micr
opterus)
Synset('bass.n.06') the lowest adult male singing voice
Synset('bass.n.07') the member with the lowest range of a family of musical instruments
Synset('bass.n.08') nontechnical name for any of numerous edible marine and freshwater spiny-finned fishes
Synset('bass.s.01') having or denoting a low vocal or instrumental range
```

In [7]:
```
print(lesk('I went fishing for some sea bass'.split(),'bass','n'))
```

```
Synset('bass.n.08')
```

In [8]:
```
 print(lesk('The bass line of the song is too weak'.split(),'bass','s'))
```

```
Synset('bass.s.01')
```

In [14]:
```
print(lesk('Avishai cohen is an Israeli jazz musician, he plays double bass and is also a composer'.split(),'bass',pos='n')
```

```
Synset('sea_bass.n.01')
```

In [9]:
```
#EXERCISE-2: Print senses for 'chair'
```

In [10]:
```
for ss in wn.synsets('chair'):
     print(ss,ss.definition())
```

```
Synset('chair.n.01') a seat for one person, with a support for the back
Synset('professorship.n.01') the position of professor
Synset('president.n.04') the officer who presides at the meetings of an organization
Synset('electric_chair.n.01') an instrument of execution by electrocution; resembles an ordinary seat for one person
Synset('chair.n.05') a particular seat in an orchestra
Synset('chair.v.01') act or preside as chair, as of an academic department in a university
Synset('moderate.v.01') preside over
```

In [11]:
```
syn = wn.synsets('chair')[0]
print(syn)
```

```
Synset('chair.n.01')
```

In [12]:
```
print("Synset name : ",syn.name())
print("\nSynset abstract term : ",syn.hypernyms())
print("\nSynset specific term : ",
 syn.hypernyms()[0].hyponyms())
syn.root_hypernyms()
print("\nSynset root hypernerm : ",syn.root_hypernyms)
```

```
Synset name :  chair.n.01

Synset abstract term :  [Synset('seat.n.03')]

Synset specific term :  [Synset('bench.n.01'), Synset('bench.n.07'), Synset('box.n.08'), Synset('box_seat.n.01'), Synset
('chair.n.01'), Synset('ottoman.n.03'), Synset('sofa.n.01'), Synset('stool.n.01'), Synset('toilet_seat.n.01')]

Synset root hypernerm :  <bound method Synset.root_hypernyms of Synset('chair.n.01')>
```

In [13]:
```
#EXERCISE-3: Disambiguate the correct senses given the contextsentence
```

In [16]:
```
from nltk.corpus import wordnet as wn
from nltk.stem import PorterStemmer
from itertools import chain
bank_sents= ['I went to the bank to deposit my money','The river bank was full of dead fishes']
plant_sents = ['The workers at the industrial plant were overworked','The plant was no longer bearing flowers']
ps =PorterStemmer()
```

In [19]:
```
def my_lesk(context_sentence ,ambiguous_word ,pos=None,stem=True,hyperhypo=True):
    max_overlaps=0
    lesk_sense=None
    context_sentence=context_sentence.split()
    for ss in wn.synsets(ambiguous_word):
            if pos and ss.pos is not pos:
                 continue
            lesk_dictionary=[]
            defns=ss.definition().split()
            lesk_dictionary+=defns
            lesk_dictionary+=ss.lemma_names()
            if hyperhypo==True:
                hhwords =ss.hypernyms()+ss.hyponyms()
                lesk_dictionary+=list(chain(*[w.lemma_names() for w in hhwords] ))
            if stem ==True:
                lesk_dictionary=[ps.stem(w) for w in lesk_dictionary]
                context_sentence= [ps.stem(w) for w in context_sentence]
                overlaps= set(lesk_dictionary).intersection(context_sentence)
            if len(overlaps)>max_overlaps:
                lesk_sense= ss
                max_overlaps=len(overlaps)
                return lesk_sense
```

In [21]:
```
print("Context:",bank_sents[0])
answer =my_lesk(bank_sents[0],'bank')
print("Sense:",answer)
print("Definition:",answer.definition)
```

```
Context: I went to the bank to deposit my money
Sense: Synset('bank.n.01')
Definition: <bound method Synset.definition of Synset('bank.n.01')>
```

In [22]:
```
print("Context:",bank_sents[1])
answer=my_lesk(bank_sents[1],'bank')
print("Sense:",answer)
print("Definition:",answer.definition)
```

```
Context: The river bank was full of dead fishes
Sense: Synset('bank.n.01')
Definition: <bound method Synset.definition of Synset('bank.n.01')>
```

In [23]:
```
print("Context:",plant_sents[0])
answer= my_lesk(plant_sents[0],'plant')
print("Sense:",answer)
print("Definition:",answer.definition)
```

```
Context: The workers at the industrial plant were overworked
Sense: Synset('plant.n.01')
Definition: <bound method Synset.definition of Synset('plant.n.01')>
```

In [ ]:
```

```