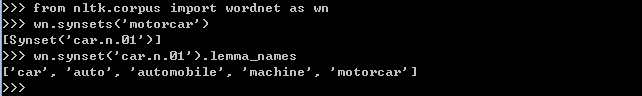
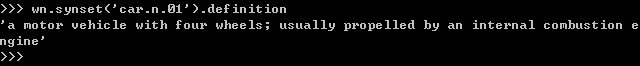
WordNet 是面向语义的英语词典，与传统辞典类似，但结构更丰富，Nltk中包括英语WordNet，共有155287个单词和117659个同义词。

意义与同义词

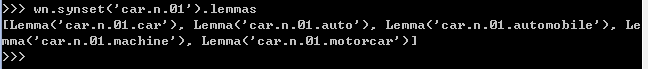


motorcar只有一个可能的含义，它被定义为car.n.01, 即car的第一个名词含义。

car.n.01被称为synset或“同义词集”，即意义相同的词（或“词条”）的集合。

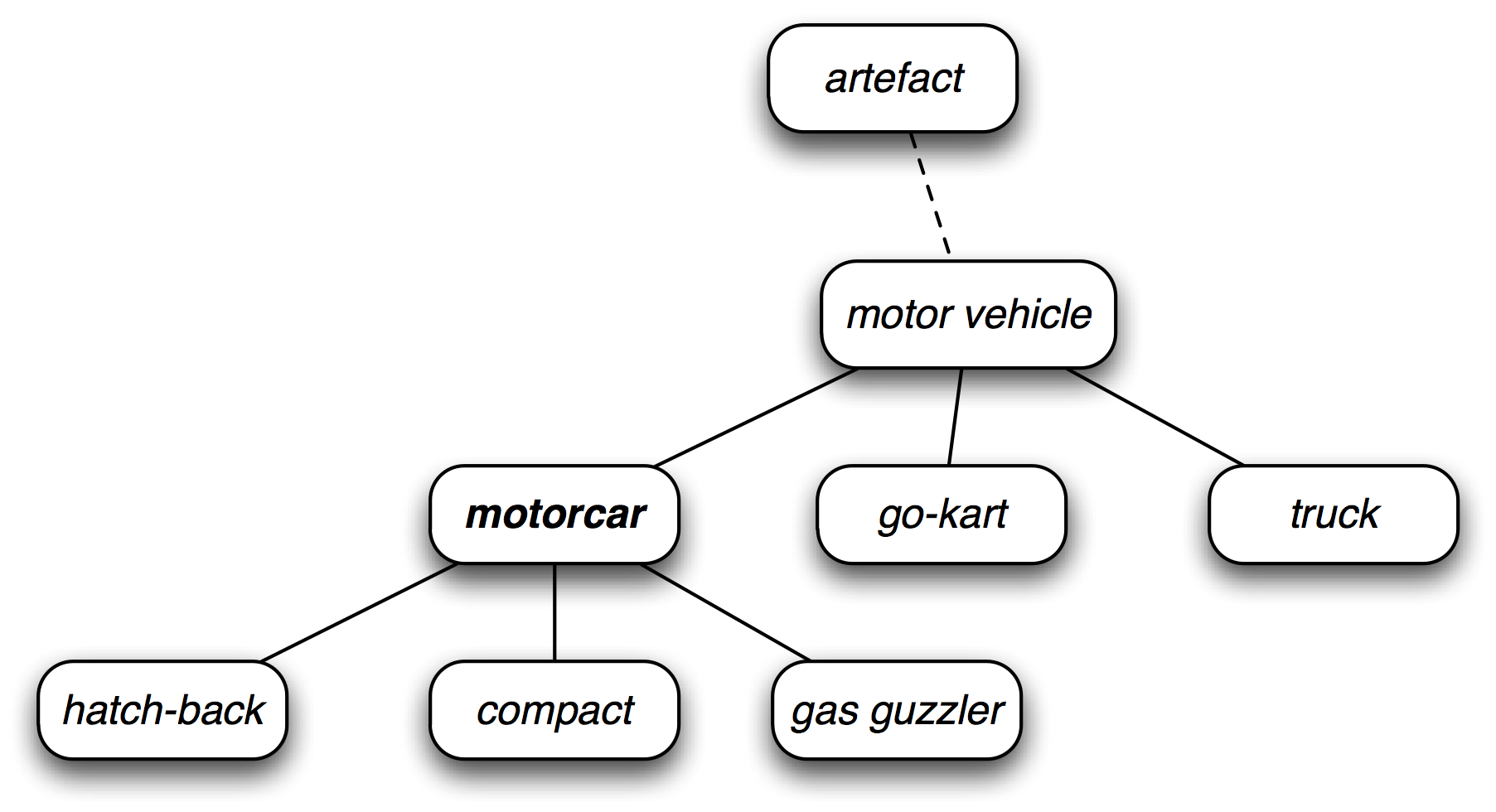


我们可以查看同义词集的定义。



我们也能查看同义词集中的单词

WordNet的层次结构



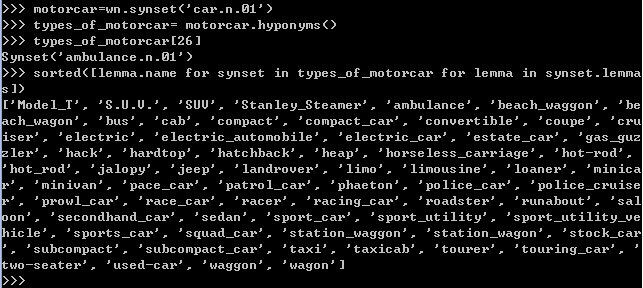
每一个节点对应一个同义词集

motorcar=wn.synset('car.n.01')

types\_of\_motorcar= motorcar.hyponyms()

types\_of\_motorcar[26]

sorted([lemma.name for synset in types\_of\_motorcar for lemma in synset.lemmas])



我们可以查看car的下位词

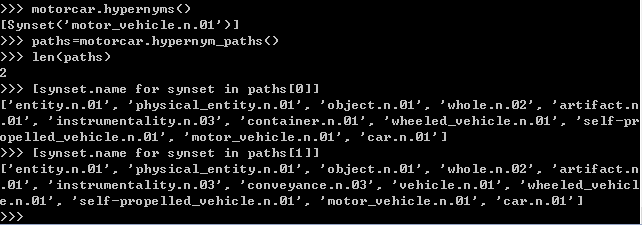
motorcar.hypernyms()

paths=motorcar.hypernym\_paths()

len(paths)

[synset.name for synset in paths[0]]

[synset.name for synset in paths[1]]



这个例子显示了car.n.01和entity.n.01之间的两条路径，路径显然不是唯一的。

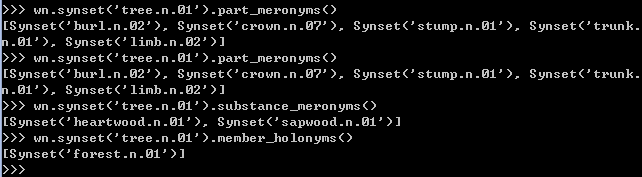
更多的词汇关系

上位词和下位词被称为词汇关系，因为他们是同义词集之间的关系。

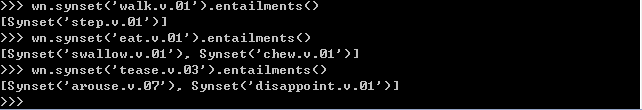
wn.synset('tree.n.01').part\_meronyms()

wn.synset('tree.n.01').substance\_meronyms()

wn.synset('tree.n.01').member\_holonyms()



一棵树可以分为树干，树冠等部分part\_meronms。一棵树有心材和边材组成，substance\_meronyms。树干的集合形成了一个森林，即member\_holonyms。



动词同样也有关系，例如吃就包含咀嚼和吞咽两个动作。

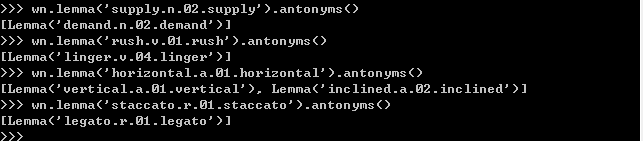
词条之间还存在一些词汇关系，如反义词。

wn.lemma('supply.n.02.supply').antonyms()

wn.lemma('rush.v.01.rush').antonyms()

wn.lemma('horizontal.a.01.horizontal').antonyms()

wn.lemma('staccato.r.01.staccato').antonyms()



语义相似度

同义词集是由复杂的词汇关系网络所连接起来的。给定一个同义词集，可以遍历WordNet网络来擦好看相关含义的同义词集。

right = wn.synset('right\_whale.n.01')

orca = wn.synset('orca.n.01')

minke = wn.synset('minke\_whale.n.01')

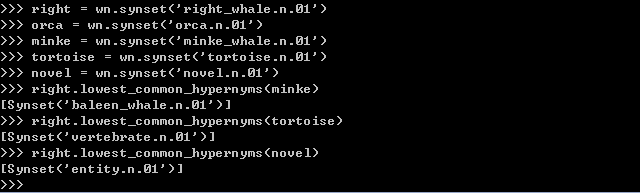
tortoise = wn.synset('tortoise.n.01')

novel = wn.synset('novel.n.01')

right.lowest\_common\_hypernyms(minke)

right.lowest\_common\_hypernyms(tortoise)

right.lowest\_common\_hypernyms(novel)



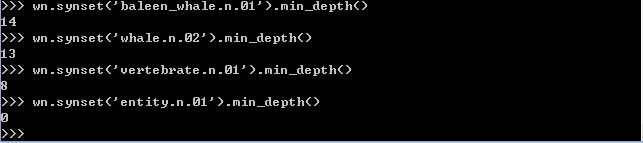
如果两个同义词集用一个特定的上位词，且此上位词在网络中处于较低层级，则他们必然有密切关系。

wn.synset('baleen\_whale.n.01').min\_depth()

wn.synset('whale.n.02').min\_depth()

wn.synset('vertebrate.n.01').min\_depth()

wn.synset('entity.n.01').min\_depth()

我们可以查看每个词在的层次位置。

通过观察词的层次，我们可以推测词与词之间的相似度。

**>>> right.path\_similarity(minke)  
0.25  
>>> right.path\_similarity(orca)  
0.16666666666666666  
>>> right.path\_similarity(tortoise)  
0.07692307692307693  
>>> right.path\_similarity(novel)  
0.043478260869565216**