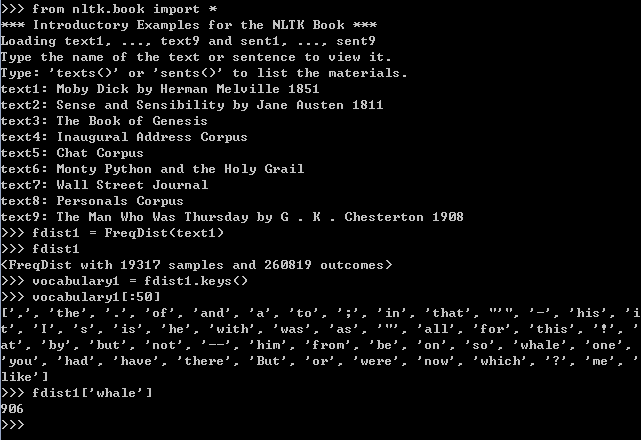
频率分布

使用FreqDist()函数得到了《白鲸记》中的单词总数，260819个单词

key()函数返回了所有不同类型的链表，然后我们查看前50项。

我们可以看到大部分都是一些无法反应文本主题的词，一个除外，就是whale，出现了906次。

FreqDist(text)

fdist1 = FreqDist(text1)

实例化得到的fdist1是一种table结构的数据，即类似dict类型，如：{“whale” : 906}.

是否能够实现如下循环：

for k,v in fdist1:

print k,v

or

for k,v in fdist1.items():

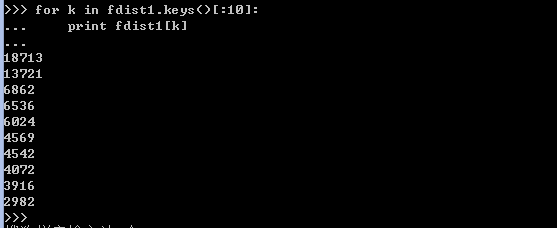
print k,v

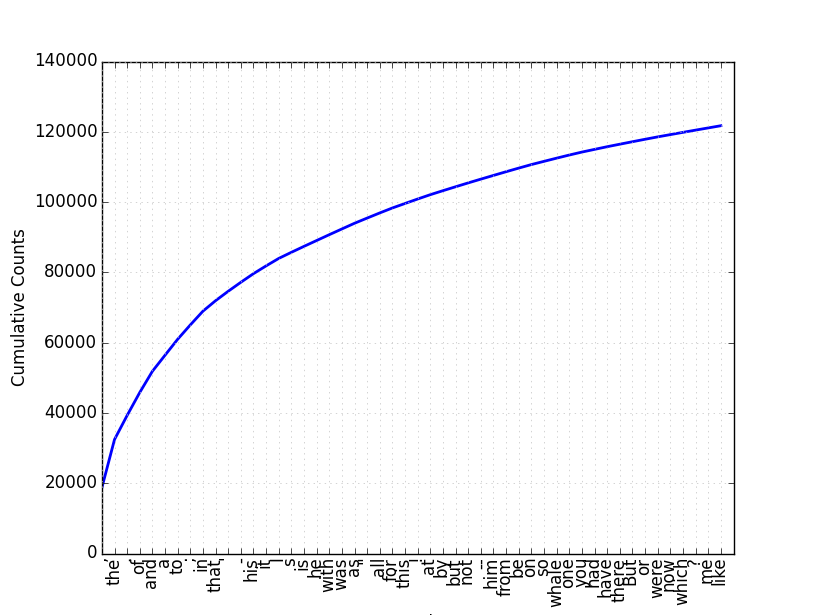
or

for k in fdist1.keys():

print fdist1[k] #得到对应的value

有空可以测试一下，看看是否能够运行？

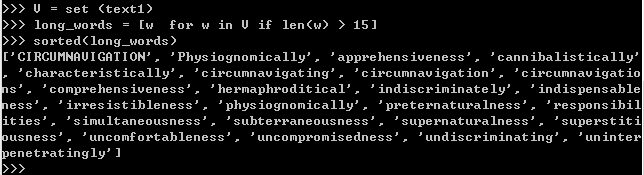


我们来看看前50的词有多少次出现。总共大约120000次，约占总数46%。

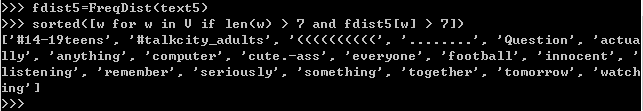
与高频词汇相对应的是那些只出现了一次的低频词汇，可以用hapaxes()函数查看，这里我就不截图了，因为有9000多个低频词汇，刷屏了。。。这些词汇太多，不结合上下很难分析出有用信息。

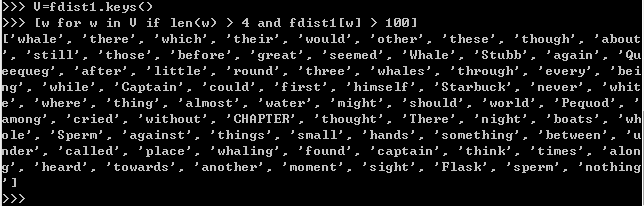
细粒度的选择词

在高频词中，像is，the这些单词很短，大多不是名词动词。那长词会否能提供更多信息呢？

这里我们取白鲸记中长度超过15的词来分析。这些词多是名词，形容词和副词。选15是因为这样长的词才足够少，能一次展示出来。但这样却忽略了一次稍短一些的有用词汇。

综上，折衷的办法也许是取一些出现次数多的长词，我们以聊天语料库为例，选取词长超过7且出现次数也超过7的。有些词有用的，例如computer，football。。。



回到白鲸记中，选取词长超过4且出现次数超过100词的词语，这里面显然包含了很多重要词汇。whale一词出现次数超过了there，毫无疑问是主题词，whaling捕鲸，Stubb，Queequeg，Pequod应该是地名，Captain队长， StarBuck星巴克， CHAPTER看出来是分章节的， boats跟船有关，还不止一条船， Sperm精液， Flask烧瓶，感觉还和化学扯上了关系。

对比一下梗概：

捕鲸船“裴廊德”号船长亚哈，在一次捕鲸过程中，被凶残聪明的白鲸莫比·迪克咬掉了一条腿，因此他满怀复仇之念，一心想追捕这条白鲸，竟至失去理性，变成一个独断独行的偏热症狂。他的船几乎兜遍了全世界经历辗转，终于与莫比·迪克遭遇。经过三天追踪，他用鱼叉击中白鲸，但船被白鲸撞破，亚哈被鱼叉上的绳子缠住，掉入海中。全船人落海，只有水手以实玛利（《圣经》中人名，意为被遗弃的人）一人得救，[1]

主人公是captain，内容与白鲸和捕鲸有关，最重要的词汇发现了，但似乎缺少了主人公的名字。因为主人公Ahab名字长度为4。看来，这样的方法还是不足以便捷地找出所有重要信息，因为缩短长度限制的话，可恶的，that，have这些又该出来了。

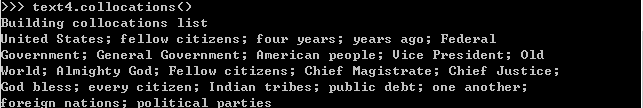
搭配和双连词

搭配是不经在一起出现的词序列。搭配的特点是其中的词不嗯能够被类似的词置换。

获取搭配要从使用bigrams()函数寻找词对（双连词）开始。搭配基本上是频繁的双连词。

bigrams()函数用来生成双连词

collocations()函数可以用来寻找搭配



这一节大致就这样了，看来搜索高频词，然后去掉一些常见词例如and，is。。。就可以得到主题词了。所以我们需要建立一个包含is，and这些对主题分析无用的常用词的词汇库。