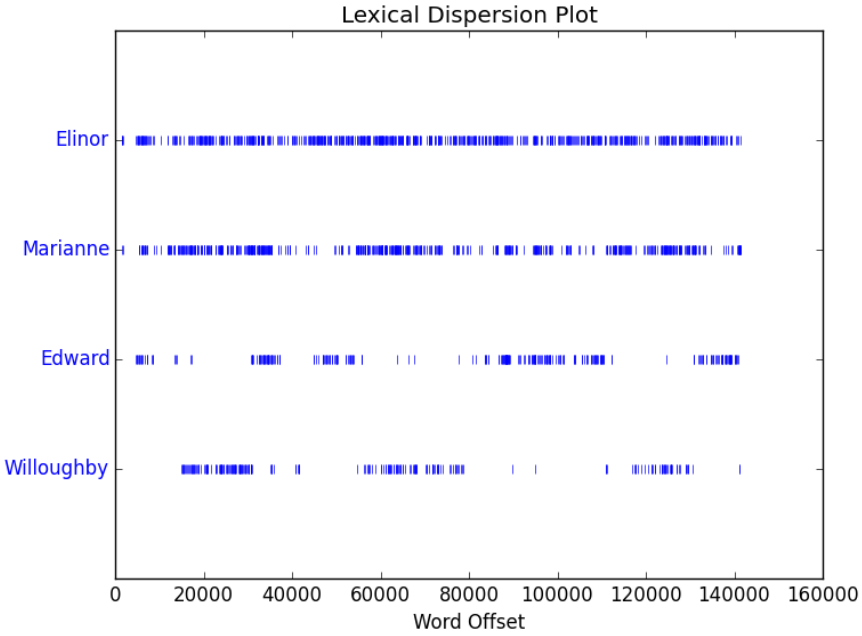
中文信息处理 作业一

13300160096 陈丹露

完整代码见lab1.py
运行结果见result2.txt/result3.txt

1.制作text2（《理智与情感》）中四个主角：Elinor，Marianne，Edward 和Willoughby 的分布图。在这部小说中关于男性和女性所扮演的不同角色，你能观察到什么？




```
>>> from nltk.draw.dispersion import dispersion_plot
>>> words = ['Elinor', 'Marianne', 'Edward', 'Willoughby']
>>> dispersion_plot(text2, words)
```

从分布图中可以看出，两位女性角色显然是站主导地位的，因而几乎贯穿了整本小说。而男性角色是次要地位，并且两位男主角几乎是交替地出现在时间轴上。这结果与简·奥斯汀的女性小说的风格吻合。

2.在聊天语料库（text5）中查找所有以字母t开头的词，按字母顺序显示出来。找出text5中所有5个字母的词。使用频率分布函数（FreqDist），以频率从高到低显示这些词。

```
>>> V = set(text5)
>>> word5 = [w for w in V if (len(w) == 5) and ((w[0] == 't') or (w[0] ==
'T'))]
>>> sorted(word5)
['THERE', 'TRUTH', 'Temp.', 'Tense', 'Thank', 'Thats', 'Think', 'Three',
'Thugs', 'Tower', 'Track', 'Troll', 'Truth', 'table', 'taken', 'takes',
'talks', 'tapes', 'taste', 'teach', 'tears', 'teens', 'teeth', 'texan',
'texas', 'thang', 'thank', 'thanx', 'thats', 'theft', 'their', 'there',
'these', 'thing', 'think', 'third', 'those', 'thows', 'three', 'threw',
'throw', 'thumb', 'times', 'tired', 'title', 'today', 'topic', 'torah',
'total', 'touch', 'tough', 'towel', 'track', 'trash', 'tried', 'tries',
'troll', 'trout', 'truck', 'truss', 'trust', 'tryer', 'tryin', 'tummy',
'tuned', 'turns', 'twice', 'twoel', 'typed', 'typin']

>>> fdist = FreqDist(w for w in text5 if len(w) == 5)
>>> fdist.most_common()
[('there', 120), ('wanna', 107), ('.....', 73), ('hello', 71), ('about', 70),
('where', 63), ('right', 54), ('think', 54), ('would', 53), ('girls', 48),
('thats', 45), ('never', 45), ('whats', 41), ('night', 41), ('gonna', 37),
('still', 33), ('today', 29), ('sorry', 28), ('didnt', 28), ('going', 27),
('again', 23), ('first', 22), ('wants', 21), ('doing', 21), ('looks', 21),
('guess', 21), ('maybe', 20), ('phone', 20), ('great', 20), ('could', 20),
('bored', 20), ('their', 19), ('later', 19), ('howdy', 19), ('other', 19),
('Hello', 18), ('thing', 18), ('sucks', 18), ('leave', 17), ('tryin', 17),
('))))))', 14), ('these', 14), ('thank', 14), ('least', 14), ('hands', 14),
('games', 13), ('times', 13), ('makes', 13), ('wrong', 13), ('cause', 13),
('place', 13), ('!!!!!!', 13), ('music', 13), ('sleep', 12), ('party', 12),
('those', 12), ('sweet', 12), ('being', 12), ('watch', 12), ('outta', 12),
('gotta', 12), ('naked', 12), ('stuff', 11), ('honey', 11), ('cream', 10),
('Music', 10), ('names', 10), ('hahah', 10), ('white', 9), ('alone', 9),
('happy', 9), ('((((((', 9), ('lasts', 9), ('while', 9), ('funny', 9),

..(太长省略，见result2.txt)
```

3.写表达式找出text2 中所有分别符合下列条件之一的词。结果应该是词链表的形式：['word 1', 'word2', …]。

a. 以er 结尾；
b. 包含字母m；
c. 包含字母序列ph；
d. 除了首字母外是全部小写字母的词（即titlecase）。

```
>>> lista = sorted(w for w in set(text2) if w.endswith('er'))
>>> listb = sorted(w for w in set(text2) if 'm' in w)
>>> listc = sorted(w for w in set(text2) if 'ph' in w)
>>> listd = sorted(w for w in set(text2) if w.istitle())
>>> listall = lista + listb + listc + listd
>>> listall
```

（结果见result3.txt）