Len(text3)-计算文本数量

Sorted -排序

Set（text3）-获得词汇表

Len（text3）/len（set（text3））

Text3.count(“smote”)

链表-存储文本的方式

1. 可以对链表加法运算
2. text[173]-索引元素的位置
3. text3.index（‘awaken’）-索引出现的位置
4. 抽取子链表-切片 text[1:20] 切片m：n 表示元素m…n-1

字符串

Name = ‘monty’

Name\*2----------‘montymonty’

Name+’!’--------‘monty!’

链表连接起来组成单个字符串-‘’.join([‘monty’,’python’])-monty python

字符串分割成链表------‘monty python’.split()----[‘monty’,’python’]

NLTK频率分布中定义的函数

fdist = FreqDist(samples)----创建给定样本的频率分布

fdist.inc(samples)----增加样本

fdist[‘monstrous’]—计数给定样本出现次数

fdist.freq(‘monstrous’)—给定样本的频率

fdist.N()—样本总数

fdist.keys()—以频率递减顺序排序的样本链表

for samples in fdist:遍历样本

fdist.max()—数值最大的样本

fdist.tabulate()—绘制频率分布表

fdist.plot()—绘制频率分布图

fdist.plot(cumulative=True)—绘制累计频率分布图

fdist1 < fdist2 ---测试是否小于

2020/8/28

s.startswith(t)

S.endswith(t)