# 大作业： 基于Markov过程的文本生成

我们知道英语是一个结构化的语言，由于语法以及语义方面的限制，一个单词后面一般会跟着某些单词，然后在这些单词后面有可能还跟着其他单词。在不知道具体的英语语法结构以及语义的基础上，通过与一大段结构规范的英文文本对应，我们大概能够知道多个单词组成的一个句子是否是一个正常的英文句子。

考虑句子"I love roses and tulips. I hope I get roses for my birthday." 对于单词I，我们发现在I后面可以跟love, hope, get，可以随机选取一个单词，比如说get；然后get后面可以跟roses；接下来roses后面可跟and或者for，比如说选择and；然后在and之后跟tulips。这样我们生成的句子就是 I get roses and tulips。

上面描述的过程实际上是一个一阶Markov过程，即在时刻t+1（这里是下个单词)仅仅依赖于上一个时刻t的状态（这里是上一个单词）。而两阶Markov过程中，下一个单词依赖于前面两个单词。我们的作业只要求基于一阶Markov过程来生成英文句子。

程序应该包括两个部分，首先是分析一段英文文本来产生一个字典，该字典记录了合法的单词序列，即某个单词后面可以跟哪些单词，注意到在英文文本中有些单词序列出现了多次，这个时候也应该反映其出现的频率。一个最简单的办法就是将那些单词记录多次。

句子的第一个单词如何选取？字典里面应该记录那些出现过的作为句首的单词。怎么判断一个单词为句首的单词？文本的第一个单词为句首单词，一个句子一般以英文句号、问号和感叹号（.？！）结束，因此在句子结束分隔符之后的单词为句首单词。

在分析文本产生字典后，就可以生成一段英文文本了，首先从句首单词中任选一个单词，然后根据选中的单词查询字典后选择下一个单词，直到遇到句子结束类型的单词，或者达到一个句子限定的单词个数（比如为20）为止。在生成文本时要能够判别句子结束，因此在第一阶段分析文本时不要去掉那些句子分隔符。

下面给出了程序的一些关键设计，仅供参考，你可以对此进行优化。

假设分析的文本如下：

X Y Z. X Z Y? Y X Z! Z Z Z. Y Z Y.

在分析这段文本后，产生的字典可能如下：

{'': ['X', 'X', 'Y', 'Z', 'Y'],

'X': ['Y', 'Z', 'Z!'],

'Y': ['Z.', 'X', 'Z'],

'Z': ['Y?', 'Z', 'Z.', 'Y.']}

  上述字典中key=’’对应的为句首单词，key=‘X’对应的为在X之后出现的单词…

最后生成的多个句子可能是：

Y X Z Z Y?

Z Z Y.

Z Z Y?

…

你的程序应该至少包含3个函数，分别是parse、generate和main函数。parse函数应该至少包含一个参数text，分析text中的文本来产生字典。generate函数应该至少包含两个参数，一个是要产生的句子个数，另一个是每个句子最多多少个单词。main函数包含了相应的测试代码，包括读取训练文本文件，调用parse和generate函数等。

如果训练的文本为beatles的'Help!'歌词，产生的10个句子可能如下：

And now these days are gone, I'm not so much younger than today.

Won't you can, I'm not so insecure.

Now I just need someone, help.

Help me if you know I do appreciate you can, I'm feeling down.

But now these days are gone, I'm not so much younger so much younger than today.

And I find I've changed my life has changed my mind and opened up the ground.

And I do appreciate you please, please help me.

Won't you know I never needed anybody's help me, get my feet back on the doors.

But every now and then I was younger so many ways.

Won't you can, I'm feeling down.