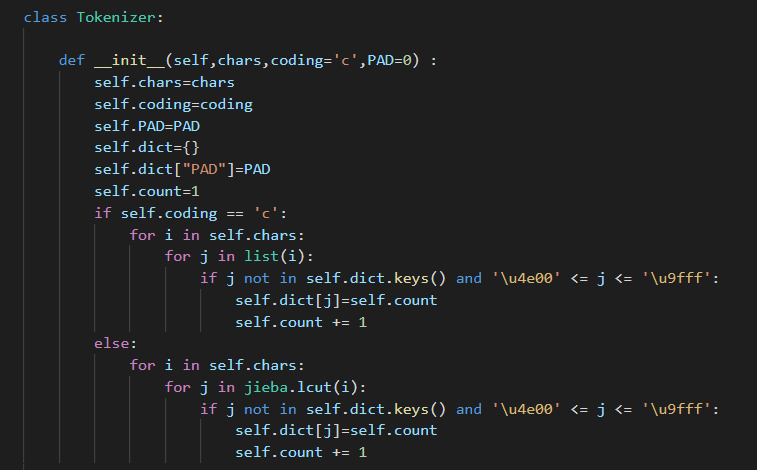
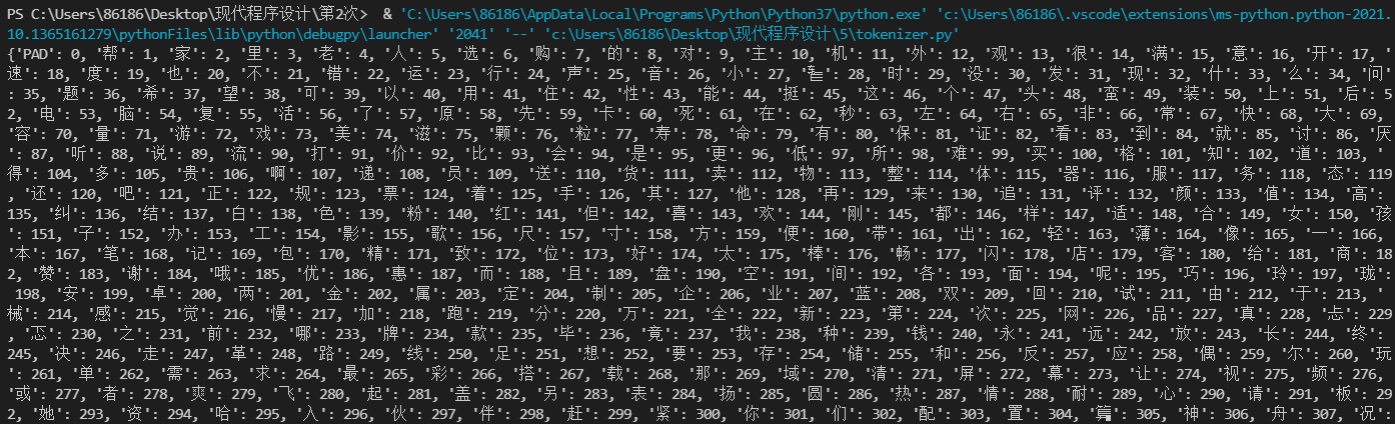
1. 进行初始化

**传递进来字符串列表后，此时根据模式（w为词模式，c为字模式）判断进行词典的初始化。同时要判断字符是否为汉字，如果不是汉字不记录进入词典**

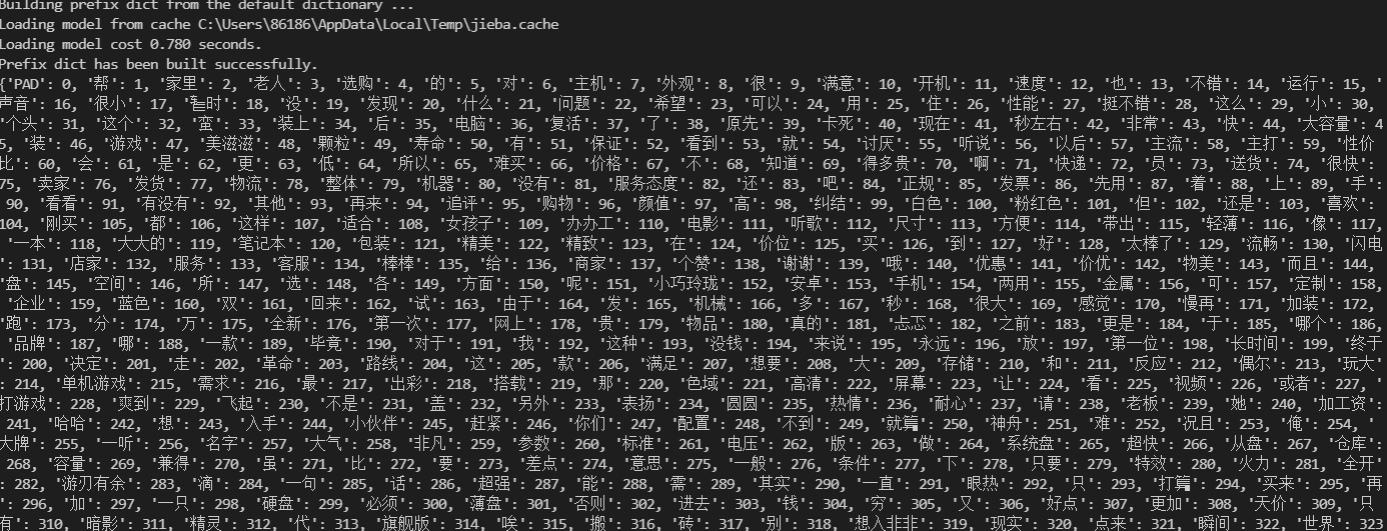


运行结果：

字模式：

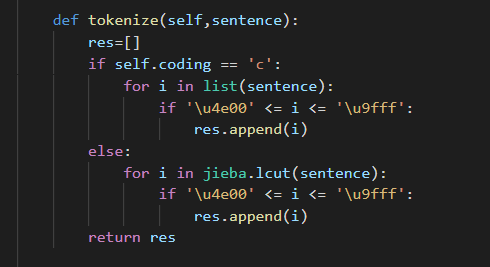


词模式：



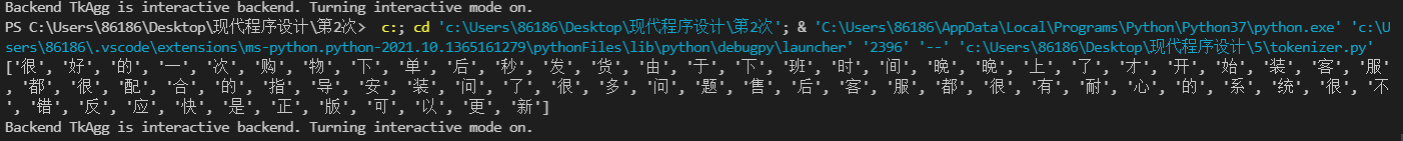
1. 2. tokenize(self, sentence) 输入一句话，返回分词(字）后的字符列表(list\_of\_chars)。

**根据不同的模式进行分词（分字），同时筛除不是汉字的字符**

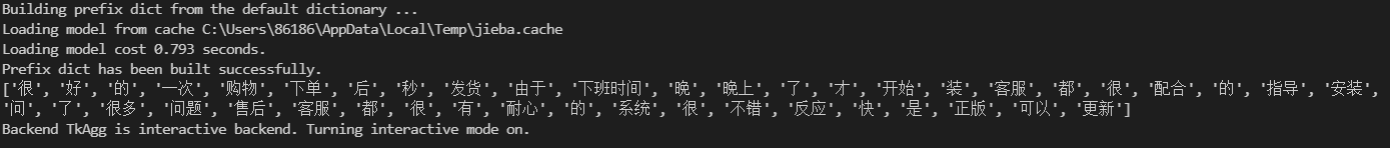


运行结果：

字模式：

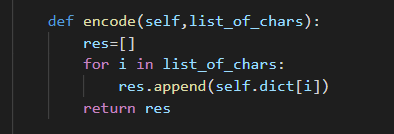


词模式：

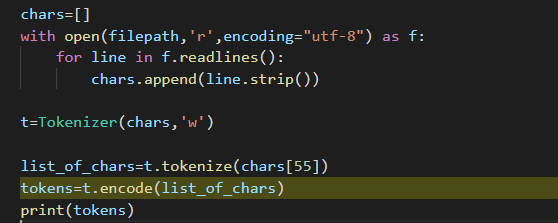


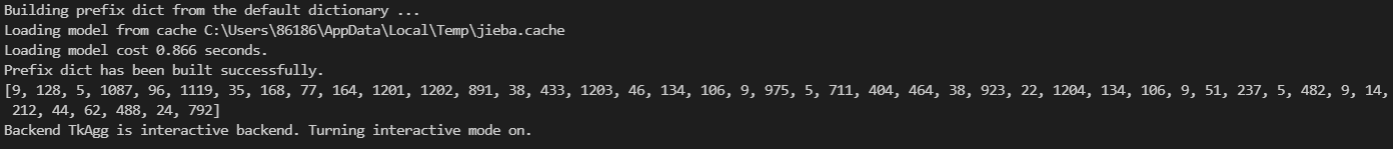
1. encode(self, list\_of\_chars) 输入字符(字或者词）的字符列表，返回转换后的数字列表(tokens)

**根据词典生成相应的数字列表**

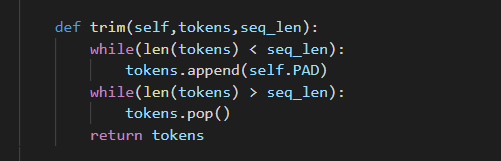


运行结果：

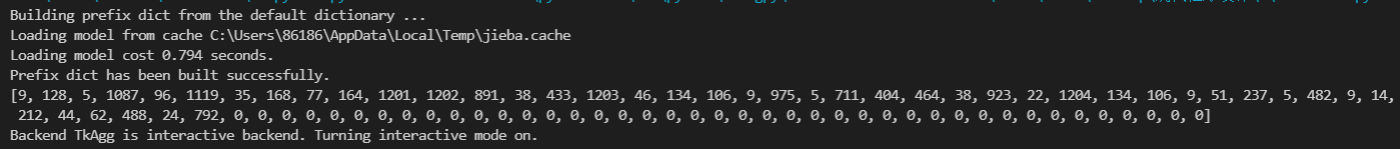




1. trim(self, tokens, seq\_len) 输入数字列表tokens，整理数字列表的长度。不足seq\_len的部分用PAD补足，超过的部分截断。

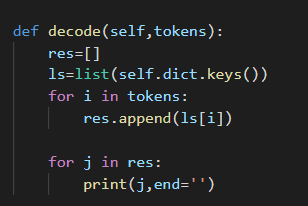


运行结果：

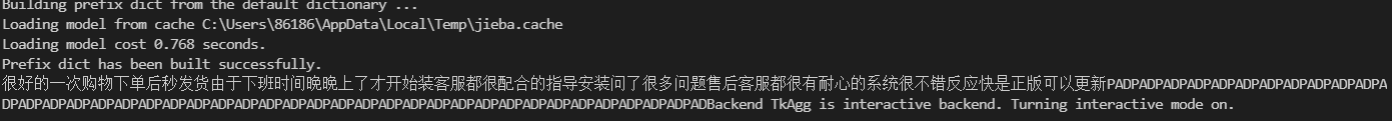


1. decode(self, tokens) 将模型输出的数字列表翻译回句子。如果有PAD，输出'[PAD]'。

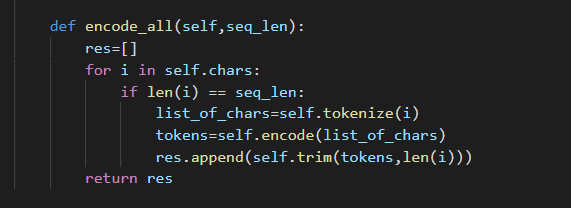
**注意到词典中的值刚好对应键的索引值，故生成键的列表，利用键值当作索引值**



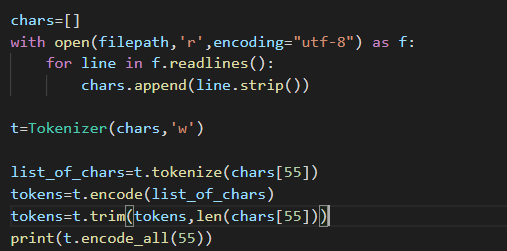
运行结果：

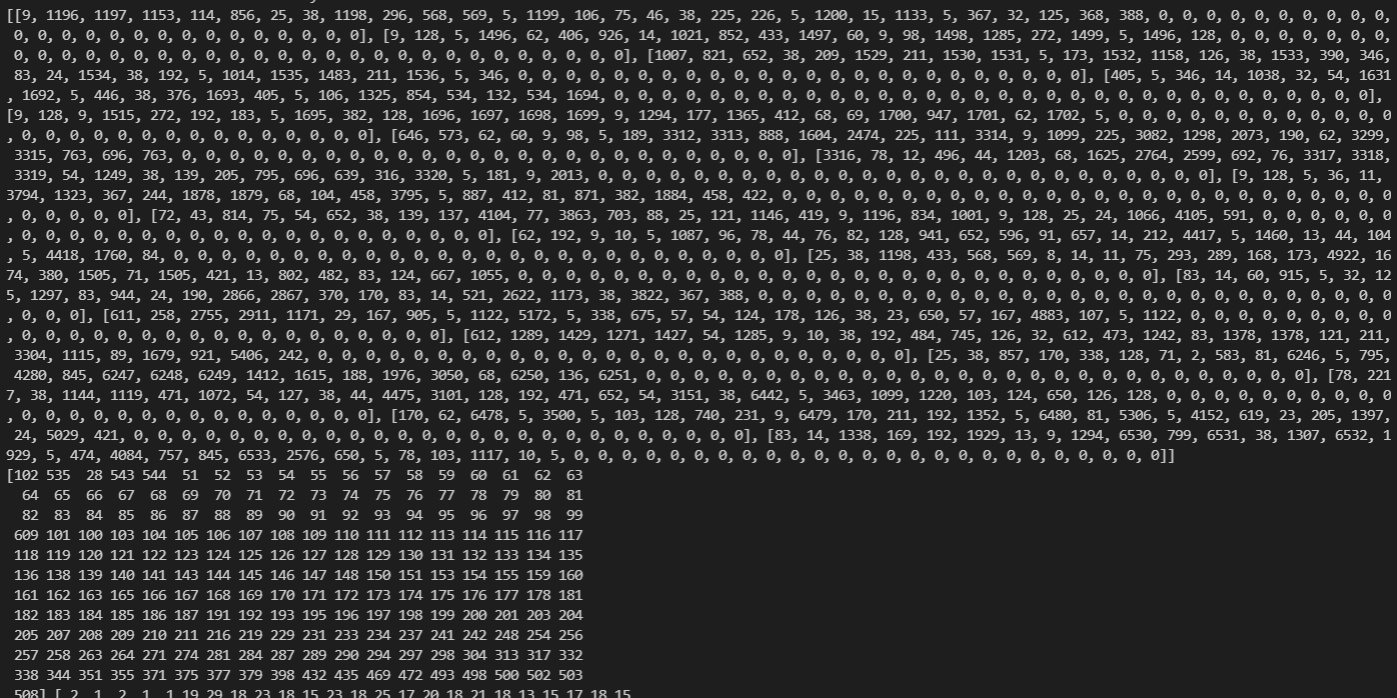


1. encode\_all(self，seq\_len) 返回所有文本（chars)的长度为seq\_len的tokens。



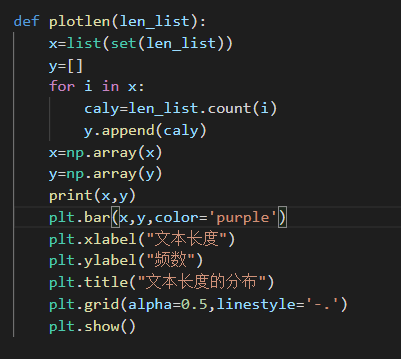
运行结果：

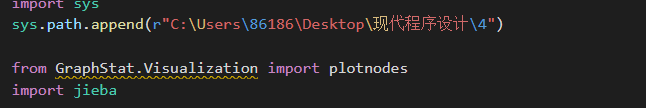




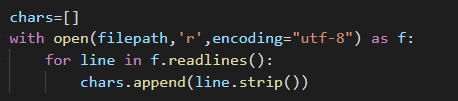
1. seq\_len是句子的长度，实际任务中一般怎么来确定一个合适的长度，请以前次作业中的评论文本或微博文本为例，通过文本的长度分布来进行观察和讨论。

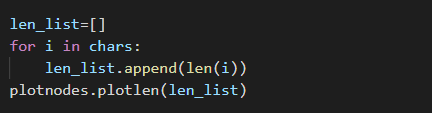
**在上次作业的基础上，在plotnodes模块中加入对文本长度分布的函数，同时在本次作业的环境变量中加入包的路径，导入对应的模块**

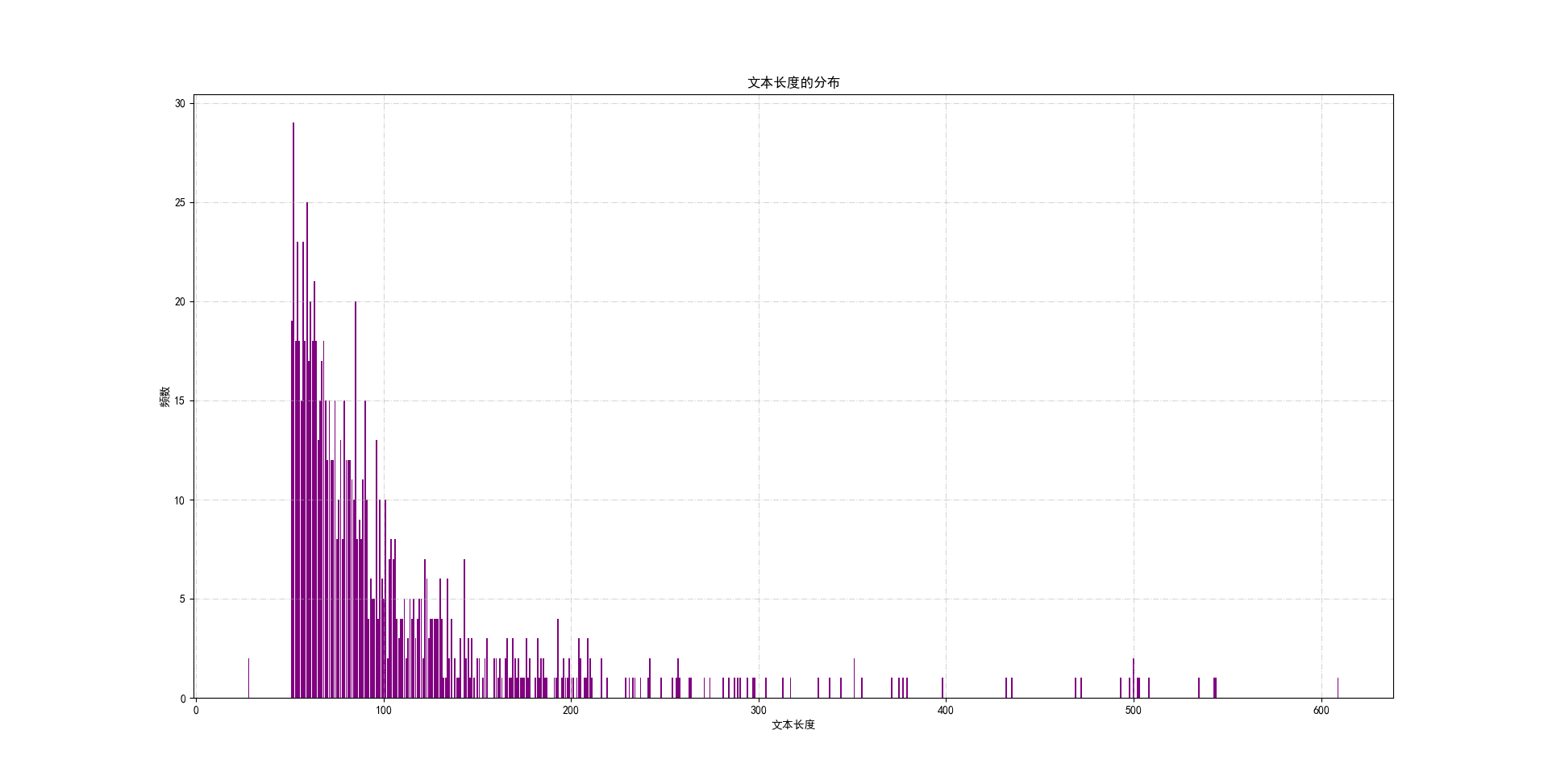




运行结果：







**将图放大可以发现，京东的评论中，长度为52的频数最多，为29次**