分词器的使用与Jieba库安装文档

秦若溪

# Jieba库的安装

分词器Tokenizer(在tokenizer.py文件中)的使用需要先安装jieba分词库，具体安装步骤如下：

1. 执行命令pip install jieba，之后会出现进度条，安装完成后显示完成

至此安装配置完成。

Jieba库相关文档：<https://github.com/fxsjy/jieba>

# 分词工具Tokenizer的使用

## 初始化与设置

分词器Tokenizer在tokenizer.py文件中。 实例化时需要传入通用词库的地址，在这过程中会读取该词库进内存，需要消耗一定时间。

## 通用词库

通用词库为“dict.txt.big”，具体位置见“Gvoice服务器端文档.docx”。词库中词语的存储格式是固定的，为：词语 （空格） 词频 （空格） 词性 （换行）

例如：

踢脚 4 n

就是词语“踢脚”，词频为4（词频越高，分词器越不会将此词拆开，一般词汇的词频为3），词性n表示名词，词性标注规则可参考<http://www.hankcs.com/nlp/part-of-speech-tagging.html>，但没有规则中这么严格。如果需要加入新词可以查看现有词库中同词性词语如何标注，并仿照该标注进行标注。

# 专业词汇分词

## 专业词词库

专业词库为construction\_dict.txt， 具体位置见“Gvoice服务器端文档.docx”。词库格式与Tokenizer中通用词库的格式相同。

## 初始化

使用专业词汇分词需要调用init\_classification方法，该方法需读入专业词库地址，初始化过程中会读取词库进内存，需要耗费一定时间。

## 分词器的使用

调用get\_keyword方法进行专业词汇分词。该方法传入一个String，如“今天天气怎么样？”，方法返回一个dict，包含’Type’与’Token’两项，其中’Type’表示分词结果的类型，有两种结果：’customized’（表示含专业词汇）或’general’（不含专业词汇）。若分词结果为含专业词汇，则‘Token’项中为所含专业词汇构成的list。若不含专业词汇，则‘Token’项中为原输入String

# 逻辑词处理功能

逻辑词处理功能用于将原自然语言的逻辑词转化为搜索引擎的搜索逻辑关键字（这里我们使用百度搜索引擎，因而引擎关键字按照百度的规范）。示例如下：

用户输入："不要辣椒和大蒜或者洋葱的川菜"

经过逻辑词处理后得到字符串：和 的 川菜 (大蒜 | 洋葱) -(辣椒)

将该串进行百度搜索，可以搜索到“川菜”条目中没有“辣椒”，且有“大蒜”或“洋葱”的搜索结果。

## 逻辑词词库

逻辑词词库文件为Logic\_words.txt（具体位置见“Gvoice服务器端文档.docx”），该词库内存储含有逻辑性质的词语。文件内每行一词，示例如下：

或者 OR

每行包含两项，第一项为逻辑词语（如示例中的“或者”），第二项为逻辑（如示例中的“OR”）。在调用init\_logic\_translator方法使会读入该词库。然后在调用logic\_translate方法时会先搜索是否含有逻辑词，然后再按对应逻辑进行处理。

## 初始化

初始化方式与分词器类似，调用init\_logic\_translator初始化，需要传入逻辑词库绝对地址。初始化过程需要读取硬盘上的逻辑词库，需要一定时间。

## 逻辑词处理器的使用

调用logic\_translate处理逻辑词，该方法传入待处理的字符串，返回一个dict。该dict有“Type”与”Content”两项： “Type”可以为”logic”，表示句中有逻辑词；或为”general”，表示句中无逻辑词。”Content”为逻辑词处理结果（字符串格式），若有逻辑词，则返回逻辑词处理后的结果，若无逻辑词，则返回原搜索字符串。

备注：若需要对含有专业词语的问题进行逻辑词处理，则也需要进行专业词库的初始化init\_classification。