分词器的选择

1. IK Analyszer

基于java语言开发的轻量级的中文分词工具包。处理速度60万字/秒，支持去除停用词，和加入用户词典。占用内存小，是项目常用的分词工具。

本项目也采用该分词器处理用户的输入。

IK Analyser设置用户自定义词典和停用词非常方便，只需要在 IKAnalyzer.cfg.xml 中配置词典的文件名和地址即可。



2. Stanford Word Segmenter

基于CRF模型的用Java编写的分词工具，支持对英文、中文、阿拉伯语进行分词。在分词时，载入相关模型即可对相应语言进行分词，但是载入模型较费时间，同时整个程序占用空间较大（主要是模型占用空间大），不予使用。

3. NLPIR

同时实现了分词与词性标注和命名实体识别，其java版本ictclas4j可以去除停用词，分词正确率达到97.58%，分词和词性标注处理速度为31.5KB/s。考虑到分词器初始化占用时间较长，分词速度不够快，不采用。

4. Jieba

Python中文分词组件，支持去除停用词和加入用户自定义词典，同时用户自定义词典可以为每条短语赋予权重。但因是Python编写的，没有采用。

5. Paoding

使用Java开发，1秒可分词100万汉字。支持加入用户自定义词典和去停用词，但在网上查阅资料得知Paoding的分词准确率不够高，因为选择使用IK Analyser而非Paoding。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 分词器 | 编写语言 | 支持去停用词 | 支持自定义词典 | 分词速度 |
| IK Analyszer | Java | **√** | **√** | 60万字/秒 |
| Stanford Word Segmenter | Java |  |  | 较慢 |
| NLPIR | C、Java | **√** |  | 31.5KB/s |
| Jieba | Python | **√** | **√** | 可达到1MB/s |
| Paoding | Java | **√** | **√** | 100万字/秒 |

用户输入的处理过程：

分词、去停用词

用户输入

每个短语分别匹配数据库

得到短语集

返回结果

取查询结果的交集