## 基于自然语言的新闻摘要处理

1. 开发环境：
   1. Windows 10 64 位系统
   2. Java 9.0.1
   3. Intellij IDEA 2017.3
   4. Maven 3.3.9 （Intellij 已经集成的）

为保证正确运行，请尽量使用相同的开发环境，Java 8 应该也可以，但是7 及以前不行，因为用到了 Java 8 及以上的新特性。

注意：涉及到中文文本的处理，请使用 UTF-8 编码！

1. 项目结构：
2. Data 文件夹，是NLPIR 工具的支持文件，包括使用证书等

因为该工具的证书只有1个月的有效期，需要定时去 <https://github.com/NLPIR-team/NLPIR> 项目的 License 文件夹中下载证书，本项目用到的是这两个：NLPIR-ICTCLAS分词系统授权 和 KeyExtract关键词提取授权

1. 各种 txt 文件，基本都是程序生成的，简要介绍如下：

* apriori.txt，Apriori 算法的直接输入文件，由程序生成
* jsyw\_seg\_pos.txt，带有词性标注的分词结果文件，由程序生成
* jsyw\_seg.txt，带有词性标注的分词结果文件，由程序生成，也是 Apriori 算法分析时的一个预处理输入文件
* jsyw.txt，新闻摘要文本数据，是由爬虫从网站上获取的
* mykeyextract.txt，一种自定义的关键词抽取的规则，在 NLPIR 的 KeyExtract 工具不可用时会用到，一般用不到
* result.txt，Apriori 算法生成的结果文件
* vocabulary.txt，文本分析时用到的字典，是 Apriori 算法预处理阶段和结果展示时用到的辅助文件

1. 各种png 文件，是文本分析结果的图形化结果
2. src 文件夹，项目代码，下面详细解释
3. target 文件夹：Maven 生成的项目class 文件
4. pox.xml，Maven 的配置文件
5. NewsAbstractAnalysis.iml，Intellij 项目的配置文件
6. 代码文件说明

本项目是 Maven 项目，只用到了一个第三方资源包（Intellij 的 GUI 设计包），直接导入 Maven 项目，会自动下载配置，需要等待一段时间。

一些依赖资源放在了src\main\resources 文件夹下，主要有一个 Logger （日志）工具的配置文件 log4j2.xml。另外有 NLPIR 工具运行时需要加载的几个 DLL，lib 库，也可以从 NLPIR 项目的 Github 主页上去下载，支持 Windows 和 Linux，MacOS 的话，只有分词工具有，本项目中的 win32-x86-64 在64位的 Windows 环境下使用。

文件的代码（src\main\java\com目录下）主要包括：

* 1. DataFetch.java，使用爬虫从新闻网站中收集新闻摘要数据
  2. App.java，程序的入口，具体的运行等方式请看注释
  3. Analysis 文件夹中的代码
* AnalysisUI.form, AnalysisUI.java 是通过 Intellij IDEA 内置的 GUI Designer 辅助设计的用户界面，进行文本分析的主界面
* KeyExtract.java，NLPIR 工具提供的关键词提取工具，进行了包装，对外提供了一些接口
* KeyWordUI.java，借助 JFreechart 工具包绘制一些统计图
* NLPIRSeg.java，NLPIR 工具提供的分词工具，进行了包装，对外提供了一些接口
* WordCount.java、WordWeight.java，为了能够对词的词频和重要性进行简便的排序实现封装类，用来存储数据对象的
  1. Apriori 文件夹中的代码
* Aprioir.java，Apriori 算法的实现，采用了 HashTree 的方式进行了优化
* AprioriChart.java，Apriori 挖掘到的频繁项集（频繁共现的词）进行可视化
* AprioriPrepare.java，Apriori 算法的准备工作
* AprioriResult.java，封装的数据对象，为了方便根据支持度排序设计的数据结构
* AprioriUI.form、AprioriUI.java，Apriori 算法的用户界面设计
* ShowApriori.java，对Apriori 算法结果的解析，属于辅助文件

1. 程序运行方式

通过 Intellij IDEA 导入后，从 App.java 的 main 函数进入。

1. 只使用 GUI 进行分析

* 通过 GUI 进行单句句子的分析，可以分词、词性标注、关键词提取，可以以统计图的方式显示结果

AnalysisUI aui = **new** AnalysisUI();

aui.*show*();

* 通过 GUI 运行 Apriori 算法，进行频繁项集挖掘，并展示结果

AprioriUI apui = **new** AprioriUI();

apui.show();

1. 测试每一个子功能，结果都在命令行输出

* 这是获取数据用的

分别设置 df.run() 的参数为0,1,2，获取三个来源的数据

为了使项目配置简单，请使用参数 = 0，即 df.run(0)

DataFetch df = new DataFetch();

df.run(0);

最后写入文件 jsyw.txt 中

* 使用 NLPIR 分词工具进行分词

NLPIRSeg nlpirSeg = new NLPIRSeg();

第三个参数指定是否要带上词性标注，0 代表仅分词，1,2,3 表示词性标注的粒度，一般选3

nlpirSeg.segmentFile("jsyw.txt", "jsyw\_seg.txt", 0);

* 分析新闻中的关键词（名词、动词、形容词），同样借助 NLPIR 分词工具

问题：当前的 NLPIR 的关键词提取模块的证书似乎还没有更新，暂时没法用，自己简单实现了一个

KeyExtract ke = new KeyExtract();

选择显示多少个词

int maxKeyLimit = 10;

String sentence = "法新社援引埃及国家电视台消息称，埃及北西奈省清真寺发生的袭击事件造成184人死亡、至少125人受伤。";

ArrayList<WordWeight> wwList = ke.keyword(sentence, 10);

可视化结果

ArrayList<WordCount> wcList = ke.getWordCount("法新社援引埃及国家电视台消息称，埃及北西奈省清真寺发生的袭击事件造成184人死亡、至少125人受伤。");

KeywordUI keywordUI = new KeywordUI("新闻关键词分析", sentence+"\n词语指数 TOP "+maxKeyLimit, wwList);

keywordUI.display();

* Apriori 算法，文本关系挖掘

算法的准备与运行

AprioriPrepare aprioriPrpare = new AprioriPrepare("jsyw\_seg.txt");

Apriori apriori = new Apriori(aprioriPrpare.getWordVocabluary(), aprioriPrpare.getLines(), 0.0075);

算法结果

ShowApriori showApriori = new ShowApriori();

System.out.println(showApriori.show());