## 新浪滚动新闻检索与分类

1. 开发环境：
   1. macOS High Sierra 10.13.3
   2. Java 1.8
   3. Intellij IDEA 2017.3
   4. Maven 3.3.9 （Intellij 已经集成的）
   5. MySQL 5.7.20

为保证正确运行，请尽量使用相同的开发环境，Java 9 的支持还有些问题。理论上支持 Windows、Linux 等其他环境。

注意：涉及到中文文本的处理，请使用 UTF-8 编码！

1. 项目结构（Maven 构建，可以直接导入）：
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，自动下载各种 lib
5. SpringBoot.iml，Intellij 中，该项目的配置文件
6. news.sql 是创建数据库的脚本，本地需要配置好 MySQL
7. 各种 txt 文件是程序生成的数据文件和模型文件
8. app.log 日志记录文件
9. 代码文件说明

本项目是 Maven 项目，用到了许多第三方资源包，直接导入 Maven 项目，会自动下载配置，需要等待较长一段时间，等待下载完成。

下面介绍一下主要的项目代码文件，

1. src/main/java/offline 中，是关于新闻收集、建立索引、建立训练模型的，主要是为最终的应用提供数据和封装一些接口
   1. analysis 包，与新闻的分析部分相关，NewsClassification.java 提供了一个模型的训练模块，还实现了根据新闻标题，加载训练好的模型，给出该新闻所属的类别的接口； NewsFeature.java 实现了新闻特征的构建和提取，包括了一个训练词向量的模型，以及把新闻标题文本映射为一个特征向量。train.txt，word2vec\_model.txt，corpus.txt 都是在这个过程中生成的。
   2. crawler 包，关于爬虫的相关实现
   3. lucene 包，使用 Apache Lucene 建立文本搜索索引
   4. utils 包，实现一些小的函数、接口等，包括数据库的相关操作
   5. offlineApp.java 是该子模块的入口，根据注释说明运行对应的任务
2. src/main/java/com 中，是基于 Spring Boot + Thymeleaf 渲染实现的 Web 项目，主要有用户注册、登录，新闻内容检索，新闻类别分析等功能。管理员还支持修改新闻类别的功能。
   1. configuration 包，实现了 Spring Web 项目的一些配置，结合 Spring Security 对登录、操作访问权限进行管理
   2. controllers 包，后端的实现，处理各种用户的请求，并且返回数据或者是网页内容
   3. models 包，Spring 实例化的对象，和数据库上的数据结构对应
   4. repository 包，数据持久层的 JPA 相关的实现，与 models 有一定的联系
   5. service 包，实现了对用户注册、登录服务的接口与数据库上的相关操作
   6. validator 包，此处暂时没用到，没有用这个实现，而是通过内置的操作对用户注册、登录的输入进行合法性的验证
   7. App.java 是 Spring Boot Web 项目的运行入口
3. src/main/resources 中是一些资源文件、项目配置和 Web 项目的静态资源
   1. application.properties 是 Spring Boot 项目的相关配置和一些常量设置
   2. log4j2.xml 是相关日志记录工具 Apache Log4j2 的配置
   3. logback.xml 涉及某些第三方库的日志级别设置（避免在命令行输出过多的信息）
   4. static 文件夹中是网页相关的 css 样式，js 脚本和一些图片资源
   5. templates 文件夹中是通过 thymeleaf 渲染的网页模板
4. 程序运行方式
   1. 准备工作
      1. 安装 MySQL 并配置，在 application.properties 文件中写入自己对应的数据库名称、连接地址和用户、密码等
      2. 根据 news.sql 中创建数据库名称为 news，再新建表，news、role、user、user\_role 四张表，一定要在 role 表中插入两条数据：(1, “ADMIN”)，(2, “USER”)！
      3. 在 Intellij IDEA 或者是 (My)Eclipse 中导入该 Maven 项目，会根据 pom.xml 自动下载依赖包，时间可能有些长，等待加载完毕
   2. 运行 OfflineApp 程序
      1. 根据注释说明，按步骤运行爬虫、建立索引、文本数据特征提取、文本数据分析模型的建立
   3. 运行网页程序
      1. 首先确认数据库处于可连接的状态
      2. 其次需要确保运行过 OfflineApp 相关程序，生成了搜索索引文件（如 luceneindex ） 和模型文件（ word2vec\_model.txt 和 model.zip ）在正确的路径下
      3. 运行 App.java，本地测试需要 16s 左右的时间完成项目的生成，实际可能略有偏差，当出现类似

` INFO 4752 --- [restartedMain] com.App : Started App in 15.766 seconds (JVM running for 18.071) `

的提示后，可以从浏览器访问 localhost:8080 进行操作