YNews 设计文档

殷翊文 2017011485 计 75

2018.9

一、版本说明

YNews 使用 Python 3.7.0 编写，后端使用 Django 2.1.1，共爬取了人民网的 4136 篇新闻信息，并据此数据库建立了一个新闻搜索系统。

使用方法： YNews 目录下在控制台输入 python manage.py runserver，后在浏览器打开 http://127.0.0.1:8000/即可进入首页。

二、设计思路

1.新闻爬取与预处理部分

/data/getdata.py:

爬取网页部分。从人民网首页 http://www.people.com.cn/开始，使用 HTMLParser 解析文件，将所有<a>标签的 href 属性值，即 url 链接提取出来，加入 pagelist（此处判断是否重复加入，防止两个网页之间的死循环）。再从 pagelist 中依次访问这些 url，提取下一个网页内的 url 并加入 pagelist。同时，每次提取到 url 时，使用正则表达式判断此 url 是否为新闻页面的格式，如果是则将此 url 链接到的 html 文件保存下来，存储在/data 文件夹里，仅为了备份。为了解压方便，已删除这些 html 文件。

/data/cutdata.py:

内容预处理。得到所有新闻的 html 文件后，用 HTMLParser 将网页内新闻的标题、发布时间和正文提取出来，存入 news 字典中，key 为每篇新闻的 ID。针对网页内正文提取不干净的情况（如经常出现登录人民网、微信扫一扫和视频推荐等字样），适当地过滤文本内容。每篇新闻提取后的结果也储存在/data 文件里，仅为了方便人工查看。在得到 news 字典后，使用 jieba 分词系统，并对除了标点、\s 等内容外的每个词建立倒排索引 index 字典，key 为词，value 为含有该词的新闻 ID（可重复）。将 news 字典和 index 字典用 pickle 打包成 newspkg 和 indexpkg，方便传入 django 后端。

/data/simi.py:

此为推荐搜索的预处理。得到 news 字典后，使用 jieba 分词中的 tfidf 模型，即关键词获取（analyse.extract_tags），提取出每篇新闻的关键词。对每篇新闻，比对其它新闻的关键词与这篇新闻的相同词的个数，选出排名最高的前四篇，将其 ID 作为相关推荐新闻，存储在 rcmd 字典里。将 rcmd 字典也 pickle 成文件 rcmd。

2.搜索后端及前端显示部分

主要代码在 search/views.py 中。

首先将 pickle 文件 newspkg, indexpkg 和 rcmd 解码成三个字典 news, index 和 rcmd，并将 news 加工成缩略的版本 newspiece 字典。分页采用后台分页，根据要显示的新闻数量计算出总页码，根据 url 末尾的参数得到当前页数，并将对应的新闻切片传入 html 文件。

当接到搜索请求时，POST 方法得到搜索的内容和时间限制。在 searchfor 函数中搜索并计时。searchfor 函数将关键词根据空格分成多个关键词，并对每个关键词提取出 index 字典中的索引列表，根据时间要求过滤后计算出列表中每个新闻 ID 出现的次数并排序，同时在新闻内容中找到出现关键字的部分，截取这一部分的 200 字摘要。最后得到结果列表，在 views.result 中处理分页并传给 html 显示。

新闻页的相关推荐只需要使用 rcmd 字典即可。

前端可在首页、搜索页、结果页和新闻详情页之间跳转。排版部分用到了 CSS，高亮显

示使用了 JavaScript。

打开首页时采用了重定向方法，自动跳转至首页第一页。

三、功能说明

1.首页

首页分页展示所有新闻，每页 30 篇。点击【全站搜索】进入搜索主页，点击新闻标题可跳转至新闻详情页。

2.搜索页

搜索页参考了当前主流搜索引擎的布局，支持多关键字、时间过滤功能。点击【返回首页】返回至新闻主页，点击【搜索】进入搜索结果页。

3.结果页

结果页显示当前时间过滤信息，搜索耗时和搜索总量，关键字有高亮效果，摘要自动定位至关键字部分。

新闻总量 4136 篇，搜索“中国”平均搜索耗时约 11.01 ms，搜索“中国 中央 习近平”平均耗时约 12.96 ms。

点击【搜索】重新搜索，点击【高级设置】返回搜索主页，重新设置时间过滤。

结果页也使用分页模式，每页显示 10 篇。

4.新闻详情页

详情页展示新闻标题、发布时间和完整正文，底部有相关推荐，可推荐与其内容最相似的四篇新闻，并有超链接。点击【返回搜索页】回到搜索主页，【返回首页】返回 YNews 主页。