# 新闻检索平台

## 1 功能概述

本软件为一个新闻检索平台，实现了检索及显示10000条来自人民网的新闻的功能。

### 1.1 主页面

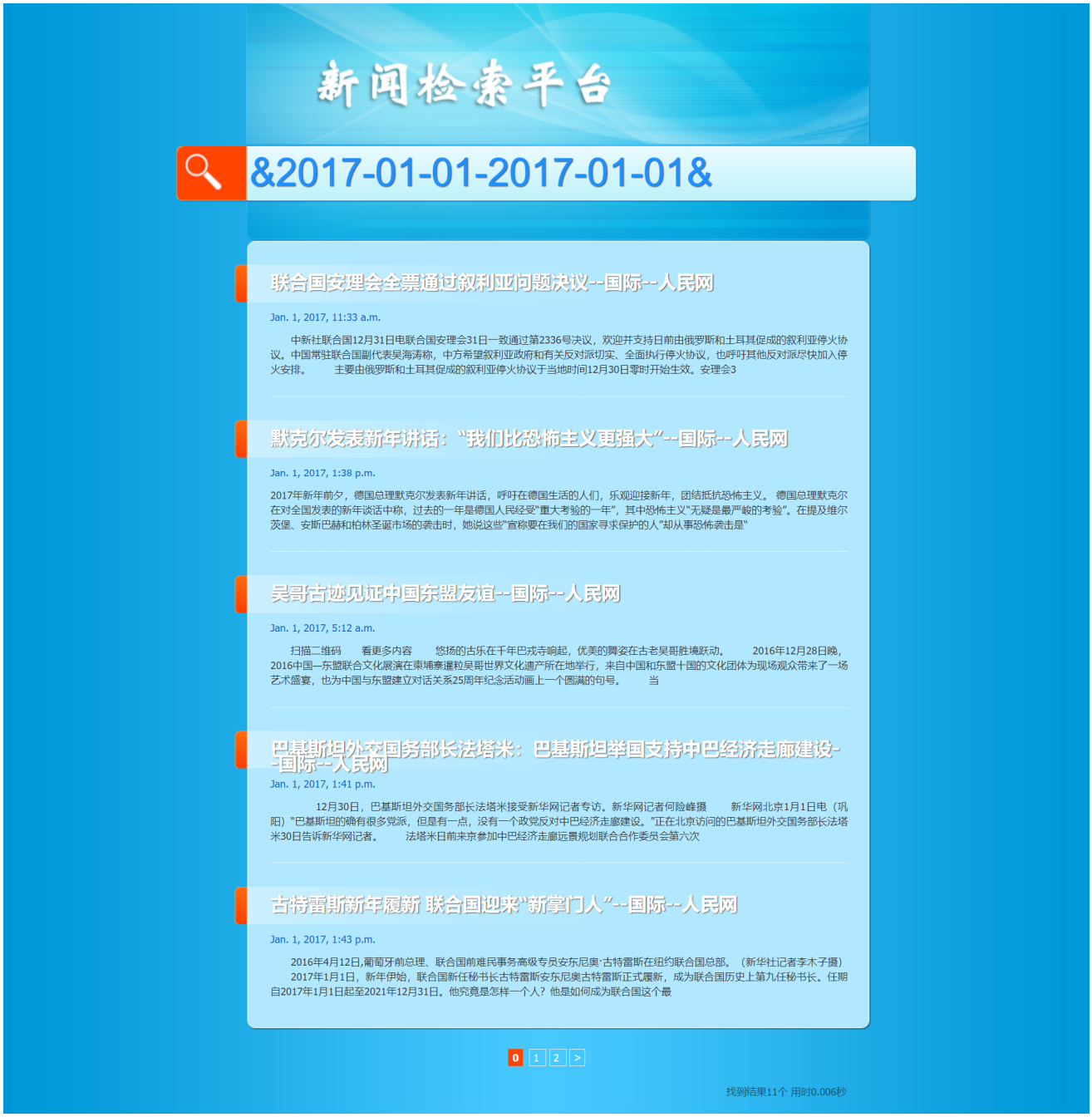
输入127.0.0.1:8000/search后，可得到以下初始界面。


可在本页面上端的搜索框中输入关键词进行搜索，也可通过下方的翻页键浏览新闻。

## 1.2 搜索结果界面

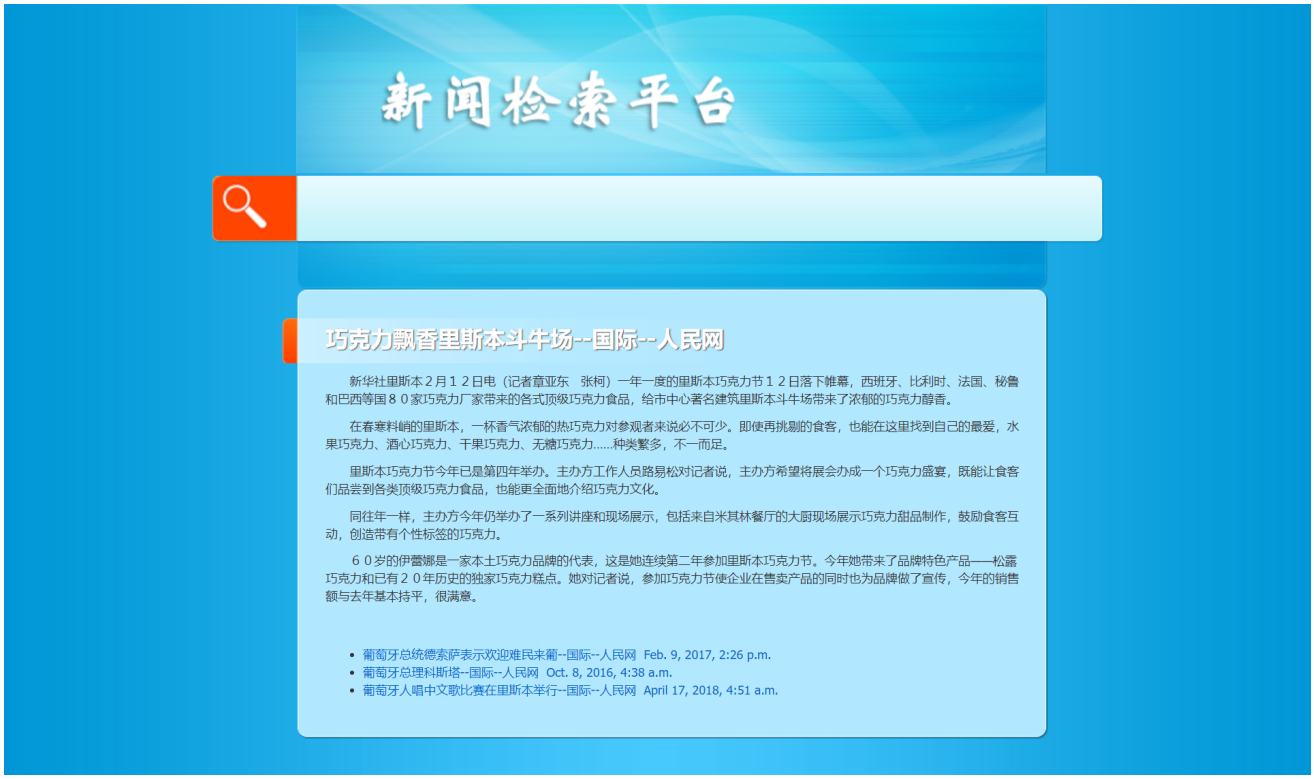

上图为搜索结果页面，搜索关键词为"习近平 普京 特朗普"，搜索引擎会返回所有与关键词部分匹配的页面，且匹配程度越高，排名越靠前。对于出现在标题、摘要中的关键词，会高亮显示。

同时，搜索引擎还支持按时间检索，若搜索关键词中包含用&&包起来的两个日期，则会返回发布时间在这两个日期之间的文章。下图是输入&2017-01-01-2017-01-01&后的返回页面。




## 1.3 详情界面

点击上述两个页面中的新闻标题后，可进入新闻的详情界面。




详情页面展示了新闻的全文，并推荐3篇数据库中相似性最高的文章。

# 2 代码实现

## 2.1 相关配置

爬虫使用python2，其版本号为2.7.13，使用的第三方库有`requeests 2.19.1`,`pandas 0.23.4`

其余部分使用python3，其版本号为3.7.0，所使用的第三方库有：`Django 2.1.1`,`jieba 0.39`,`numpy 1.14.5`,`pandas 0.23.3`

## 2.2 爬虫

从网页http://world.people.com.cn/n1/2018/0911/c1002-30284709.html开始，对网页的相关新闻版块进行bfs，提取每个网页的标题、发表时间、正文，从0开始按爬取时间对所有爬取到的网页进行编号，把标题、发表时间存至'`info.csv`文件中，将正文存到以编号命名的txt文件中。爬虫代码位于`scripts/crawler.py`中

## 2.3 数据处理

从爬取到的文件中提取文章摘要，存到`info.csv`中。

使用`jieba`的search模式对每篇文章的标题及正文进行分词，记录文中出现的词及出现次数（词在标题中每出现一次按照5次计入），使用出现次数在2次及以上的词建立倒排索引，在索引表中同时存储出现次数，倒排索引存储于`mix.csv`中

上述代码主要位于`scripts/segment.py`中

使用`tf-idf+jaccard similarity`进行相关文章推荐，首先计算每个关键词的`idf`值，导入到`mix.csv`中，再枚举所有文章对，计算每一对文章的相似程度，存到`similarity`文件夹中以文章编号命名的csv文件中。此部分代码位于`scripts/similarity.py`中

之后，将`info.csv`、`mix.csv`、`similarity/*.csv`中内容导入Django创建的sqlite3数据库中，此部分代码位于主要位于`database_insert.py`中

## 2.4 搜索算法

使用打分的方法进行搜索。

若开启了时间检索，则为位于这个时间区间内的文章加20000分，并将`threshold_time`值设为20000，否则将`threshold_time`值设为0.

若使用了关键词检索，则将`threshold_text`值设为1000，否则设为0。对于检索使用的每个关键词，若在某篇文章中出现，则为该篇文章加$1000+log(t)$次，其中t表示关键词在文章及标题中的出现次数（标题中每出现一次按5次计算）。每个关键词有1000分的基础分保证了出现关键词较多的文章得分会较高，使用`log`函数是为了适当平衡出现次数多的关键词和出现次数少的关键词的贡献。

之后，将得分大于等于`threshold_time+threshold_text`的文章返回，返回前将文章按得分从高到低排序。

## 2.5 服务器搭建

服务器使用Django进行搭建。

前端分为2个页面`index.html`与`article.html`，`index.html`用于展示主页面及搜索结果，并使用JavaScript实现关键词高亮。`article.html`用于展示新闻详情及新闻内容。

这两个页面都是在网上下载的网页模板的基础上添加了搜索栏，更改了上部的网站名称而得到的，网站模板位于`template`文件夹中。

后端使用数据库进行交互，当用户点击文章详情时，读取本地静态文件进行显示，当用户进行搜索时，调用2.4中所描述的搜索算法进行搜索。