”司法大数据自动化标注与分析“项目说明文档

一、概述

本项目为”数据科学基础“课程大作业，包含了要求文档中要求提交的**数据集**、**研究报告**、**完整项目源码**及**演示视频**，演示视频及该说明文档均可在项目主网页中的”网页信息“选项中查看。

本文档对项目源码、研究报告、演示视频进行概述与展示。

项目文件结构如下：

```
.
├── 说明文档.pdf    # This File
├── 研究报告.pdf
├── 演示视频.mp4
├── 数据集          #数据集文件夹
|   ├── 案件文本    #案件文本文件夹
|   └── 标注        #案件标注文件夹
└── code          #项目源码文件夹
    ├── back
    |   ├── main.py     #jieba-分词程序
    |   ├── user.txt    #jieba-用户自定义文本
    |   └── server.js   #项目后端程序
    └── front
        ├── main.html   #项目主网页
        ├── innersite
        |   ├── arti.html   #手动标注分页面
        |   ├── auto.html   #自动分词+勾选标注分页面
        |   └── info.html   #项目信息分页面
        ├── css         #网页样式表文件夹
        |   ├── style_main.css
        |   ├── style_arti.css
        |   ├── style_auto.css
        |   └── style_info.css
        └── icon
            └── shortcut_icon.ico
```

本项目小组成员及分工如下：

> 吴筱权 201250103：jieba分词
>
> 陈广华 201250103：文本标注、研究报告
>
> 葛家辰 201250105：前端及前后端交互

二、前端

本项目前端采用**HTML+CSS+JS**搭建。用户打开项目主网页后，可从侧边栏中选择”手动标注“、”勾选标注“、”网页信息“三个子网页，并在本主网页中查看（子网页在"iframe"标签内展示）。

主网页

用户可在视口固定位置（fixed）的侧边栏选择子网页进行浏览。

关键代码实现：

子网页展示由iframe标签提供。

```
// JS代码
var frame = document.getElementById("innersite").innerHTML;
frame = "<iframe src='innersite/"+selectPage+".html' id='frame'></frame>";
```

```
<!-- HTML代码 -->
<div id="innersite">
        <iframe src="innersite/arti.html" id="frame"></frame>
</div>
```

手动标注

用户选择”手动标注“子网页后，可点击”上传案例文件“按钮选取文件上传，网页将文件内容显示在文本框中。用户也可直接将文本复制到文本框中，后通过”保存案件“按钮下载文本，下载文件名为”案件文本.txt“。

文本框下方提供**手动标注工具**。用户可点击”添加一条标注“按钮，生成一个空标注，在左侧下拉栏中选择或直接输入其他标注类型，在右侧输入框中输入具体标注信息，完成后点击”保存标注“按钮下载json格式标注文件，下载文件名为”标注.json“。

关键代码实现：

在用户点击”生成一条标注”按钮后，触发JS函数生成两个input标签，随后包含这两个标签的newTag变量将被加入html中。

```
var newTag=document.createElement("div");
newTag.className="tag";
newTag.innerHTML="<input type='text' list='catas' class='cata' placeholder='输入或选择标注类型'>";
newTag.innerHTML+="<input type='text' class='cont'>";
```

在用户点击”保存标注”按钮后，出发JS函数遍历的input标签，将结果提取后输出文件。

```
var tags = document.getElementsByTagName("input");
for(var i=1;i<tags.length;i+=2){    // i=1: 前有一个隐藏file类型input标签提供文件选择，由“上传案例文件”按钮触发
    //...   // i+=2: 一组两个input标签遍历
}
```

勾选标注

用户选择”手动标注“子网页后，用户上传、下载标注文本步骤如上。

文本框下方提供**勾选及手动标注工具**。用户将文本上传并在文本框中完成编辑后，点击“生成标注”按钮，网页通过**AJAX**向后端发起POST请求并发送文本框中文本，收到后端发来的JS数组形式分词数据后，将数据转化为数组，并按照数组内容分名词、动词、形容词三大类生成标注。用户可点击相应标签类型后勾选标注，完成后点击”保存标注“按钮下载json格式标注文件，下载文件名为”标注.json“。此外，用户也可用手动标注工具补充相应标注。

关键代码实现：

用户点击"生成标注"按钮后，网页向后端发起请求，并生成勾选项。

```
//发起请求
var getKeywords = new XMLHttpRequest();
getKeywords.onreadystatechange=function(){
    //...
}
getKeywords.open("POST","http://127.0.0.1:8888/",true);
getKeywords.setRequestHeader("Content-type","application/x-www-form-
urlencoded");
updateText();   //获取文本框中文本
getKeywords.send(text);
```

```
//生成勾选项
var verbKeys = document.createElement("div");
verbKeys.className="keywords";
for(var i=0;i<array[1].length;i++){
    var tempKey = "<label class='auto_label'><input type='checkbox'
class='auto_check'>";
    tempKey+=array[1][i]+"</label>";
    verbKeys.innerHTML+=tempKey;
}
var verbTitle = document.createElement("h5");
verbTitle.innerHTML="动词";
//...
```

用户切换标签类型时，触发JS函数以数组形式记录勾选标注工具的所有input标签checked属性。

```
for(var i=0;i<inputs.length;i++){
    if(inputs[i].checked)
        checkStatus[parseInt(oldId)]+="1";
    else
        checkStatus[parseInt(oldId)]+="0";
}   //checkStatus为字符串数组
```

保存勾选标注时，触发JS函数根据数组储存结果找到匹配label标签的内容，提取后输出文件。

```
var tags = document.getElementsByClassName("auto_label");
// ...
for(var i=0;i<tags.length;i++){
    if(checkStatus[1][i]=='1')
        preJson.当事人+=","+tags[i].innerText;
    //...
}
```

网页信息

网页信息提供演示视频及该说明文档展示。

三、后端

后端通过**Node.js**获取前端POST请求，将文本提取后调用**jieba**分词程序，通过标准输出流获取分词结果，然后将结果回传至前端。

分词程序中，本次分词工作中的主体函数为：

```
def divide_words(text,file_name)
```

此函数用于处理文本字符串从而完成分词，用户需要传入文本字符串，最终会打印包含名词、动词、形容词三大词性的结果列表。分词的具体方法源于jieba分词自带的词性标注功能，通过该功能可以将字符串文本中的词按照不同的词性进行分类，再通过多个if判断语句来过滤不需要的词性从而最终实现精确分词。其中较为关键的步骤在于对python中字典数据类型的运用，通过key-value的一一对应关系来过滤不需要的词并筛选所需的词，且用到了如下的四个嵌套函数：

```
def merge_place(list)
def merge_name(list)
def merge_words(list, key_word,word_attribute)
def dict_to_list(old_list, new_list)
```

由于jieba分词存在的精确度缺陷，因此设计了前三个函数分别用于合并某些系统无法自动识别的地名、人名、专有名词。篇幅原因，代码具体实现便不在此展示，最后一个函数用于将每一个字典类型的词汇合并为一个列表，从而形成名词、动词、形容词三个词汇列表，最终再将这三个词汇列表储存至结果列表中。另外，由于jieba分词模式能够使用用户自定义字典，因此根据本次数据的特殊性，特在代码的首行加入了**user.txt用户字典**，并储存了一些与本次作业相关的特殊词汇来供jieba识别，从而提高分词的准确度并减小误判的几率。

关键代码实现：

在前端请求内容传输完成后，通过Node.js中子进程的spawn方法调用main.py分词程序并回传分词结果。

```
request.on('end',function(){
    var childProcess = require('child_process');
    var process = childProcess.spawn('python', ['main.py', text]);
    process.stdout.on('data', function(data){
        response.end(data);
    });
    //...
});
```

在分词程序中，添加一行代码保证标准输出流中中文为UTF8编码，防止传输乱码。

```
sys.stdout = io.TextIOWrapper(sys.stdout.detach(),encoding="utf-8")
```

前后端接口：http://127.0.0.1:8888

运行后端程序需要下载的组件：Node.js、jieba分词库

四、研究报告

欲探究被告人性别，犯罪年龄这两个变量与刑期之间的潜在关系，小组开展了相关的数据分析，数据来源于中国裁判文书网。通过应用**描述统计、单变量的T检验、双变量相关性分析、线性回归、非线性回归，独立变量T检验**等方法得出犯罪年龄的置信区间在35.97-39.28岁之间，刑期的置信区间在59.575-96.172月之间，犯罪年龄与刑期之间相关性较差以及性别对刑期并无影响等结论。

五、演示视频

详见项目主网页。