小组项目总结

1. **分工**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **全员协作** | **段孝辰** | **秦浩洋** | **李善灡** | **潘光林** |
| 前端+后端设计 | 登录与注销的前端开发 | 文件解析功能开发（PHP） | 文件解析功能开发（Python） | 网络搜索功能的开发 |
| 需求文档的编写 | 登录与注销的功能开发 | 项目界面的前端开发 | TB的新建与编辑功能开发 | 机器翻译功能的开发 |
|  | 翻译界面前端开发 | 项目界面的功能开发 | TB界面的前端开发 |  |
|  | 翻译界面基础功能与复制原文、添加术语功能的开发 | TM界面的前端开发 |  |  |
|  | 翻译界面的翻译资源支持功能开发 | TM界面的前端开发 |  |  |
|  | TM的新建与编辑功能开发 | TM的导入与导出功能开发 |  |  |
|  | 回收站界面的前端开发 | TB界面的前端开发 |  |  |
|  | 回收站界面的功能开发 | TB的导入与导出功能开发 |  |  |
|  | 代码整合 |  |  |  |
|  | Demo视频录制 |  |  |  |

1. **项目难点**
2. Word文档的php解析实现（秦浩洋）

文件解析是在本项目中非常重要的一功能需求，其中包括xlsx格式的各类翻译资源文件，pdf/xlsx/docx格式的各类翻译文件，而其中最复杂就要数docx文件。docx文件中元素多样、分层复杂，需要逐级理解一个docx文件的层次，并从中取出需要翻译的内容。



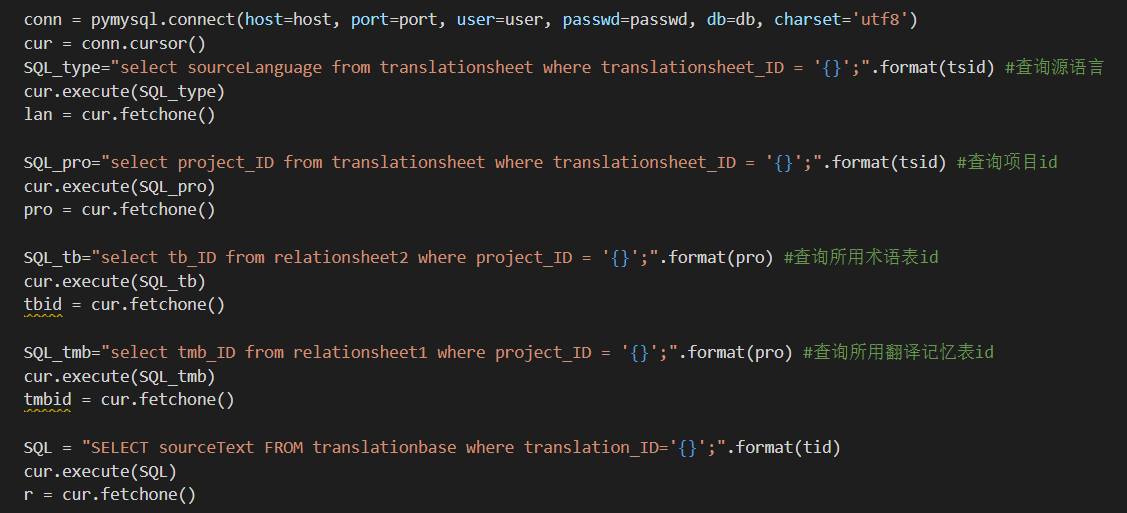
本项目中使用了PHPword包中提供的方法，解析出了docx普遍具有的Section-Element-Text的三级结构，从中筛选出具有待翻译内容的Element，最终形成了呈现在翻译界面中的文字内容。



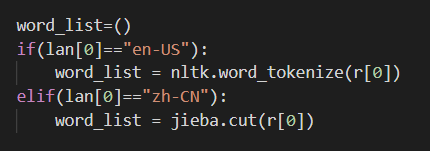
2. 翻译资源支持功能服务化（段孝辰）

在开发翻译界面时，由于涉及到的功能很多，全部使用php来写会使加载速度过长，因此决定将翻译资源部分整个服务化，使用python的flask框架写成api来使用。

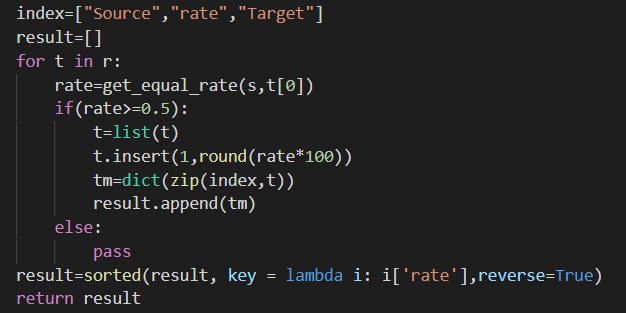
设计该服务的主要思路为，传入翻译表id句段id，根据翻译表id从数据库中查询对应的项目id和源语言。根据项目id可以进一步查询获得当前所使用的TM表和Term表。



根据源语言判断使用哪一种分词工具，中文使用jieba分词，英文使用nltk。使用句段id获得该句段的原文内容，进行分词后，就可以进行术语和翻译记忆的查询了。



翻译记忆查询中涉及到模糊搜索的内容，但是由于我们使用的是xampp，其中自带的Mariadb并不支持中文分词，所以采用了另一种方式：先搜索出所有翻译记忆，然后分别计算匹配度，将大于0.7的内容作为返回值。这样就解决了模糊搜索和计算匹配度的问题。



在具体使用过程中，由于涉及跨域的问题，因此最终返回值包装成了以下形式，为其添加header后就可解决该问题。

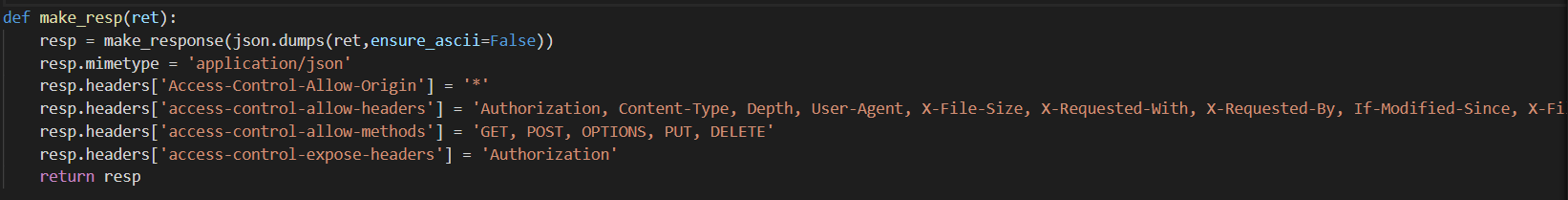
res = {

'status': 'ok',

'msg': '',

'data': []

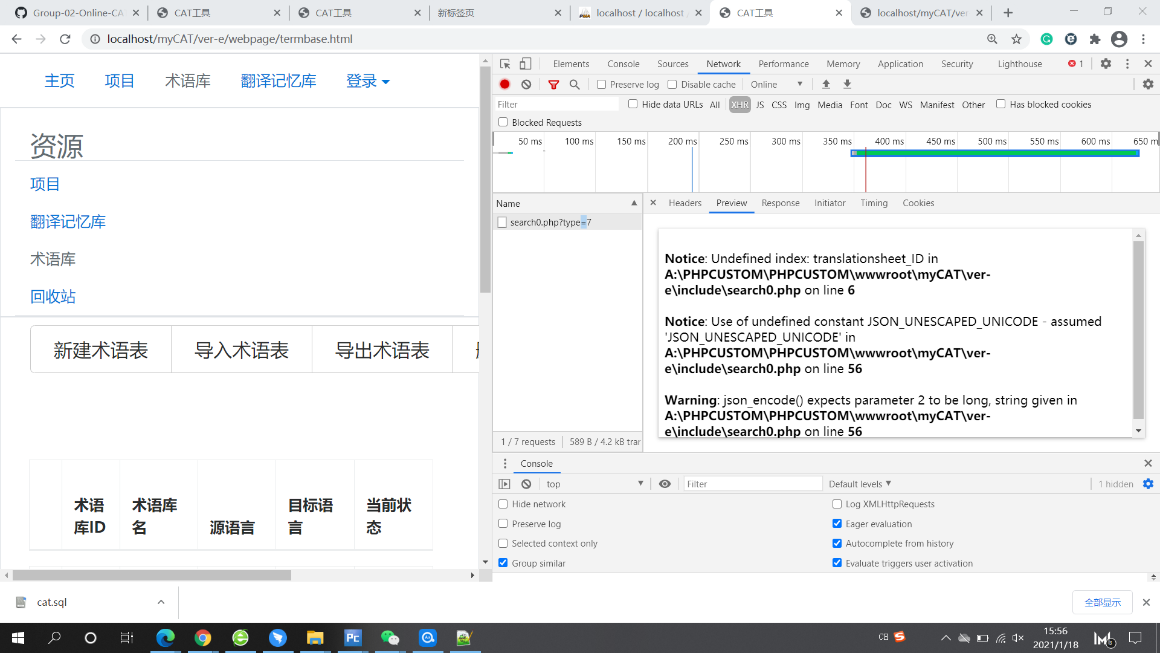
}



为了方便前端获取及处理数据，‘data’中的内容被包装为[term:{"source":,"rate":,"target":},tm:{ "source":,"rate":,"target":}]的形式，其中rate为匹配度。

3. 环境配置（李善灡）

在编写术语库前端页面的新建术语表时，遇到的一个问题是：虽然插入数据库的操作是成功的，但是后端数据怎么也不在前端显示，我的jGrid表格始终是空白一片，一开始以为是jGrid的代码有问题，全网搜各种“jGgrid数据库数据不显示”，都没效果，就这个问题整整卡了一天。



报错中的一条：Warning: json\_encode() expects parameter 2 to be long, string given in A:\PHPCUSTOM\PHPCUSTOM\wwwroot\myCAT\ver-e\include\search0.php on line 56

让我最终发现是由于自己的PHP版本过低，这种情况下有两种解决方案，1. 升级PHP 2把PHP不认识的参数改一下。我先根据CNSDN的一个帖子使用第二种方法进行了修改： 

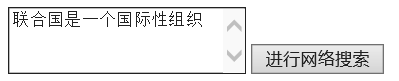
（来源：<https://blog.csdn.net/username666/article/details/109612075>）

果然，数据这回成功显示了：



1. **项目特色**
2. 网络搜索功能的开发：

网络搜索功能独立，实现起来较为快速简单，其中较麻烦的点在于思路的构建。一开始我的想法是从文字框入手，对选中内容进行处理，但这样就无法灵活查询任意字段，增加了此CAT工具的局限性。



转变思路，通过调用window.getSelection更快捷的实现了获取选中内容，实现同时多源查找，使代码量缩减了70%。

1. 机器翻译功能的开发：

此功能须借助外部开源机翻API，同时不同提供自然语言处理的相关企业机翻API接口使用方法不一。除此之外，部分机翻API接口收费，缩小了可选范围。开发该功能时，主要精力花费在了寻找各语言服务提供商的机翻API接口上。例如：百度、有道、讯飞、谷歌、腾讯等市场巨头。经过屡次查询和调试，受收费情况和可用性的限制，最后决定在CAT工具中引入百度和有道的机翻接口。具体思路为：将注册后获取到的各应用ID和密钥存入数据库，工具使用者可以按需调用心仪的机翻功能。



