**设计思路**

本系统的目标是构建一个关联地理信息与医疗领域信息的知识库，系统设计也围绕着这一目标进行。

对于构建一个知识库而言，主要任务分为两个方面：一是数据的收集和整合，二是设计知识库的使用规则。对于融合地理信息和医疗领域知识的知识库而言，我们需要的数据主要有以下几个方面：1）全国地理实体数据；2）与地理实体对应的地名数据；3）全国医疗机构数据；4）大学研究所等医疗领域科研成果数据。以上数据整合后构建一个融合地理与医疗的整体知识库，这个知识库就是我们所需要的数据源。在此数据源的基础之上，本系统进行基于地理信息的医疗知识推荐。推荐的依据是通过命名实体识别提取的地理相关信息，推荐结果是知识库中存储的医疗领域知识。

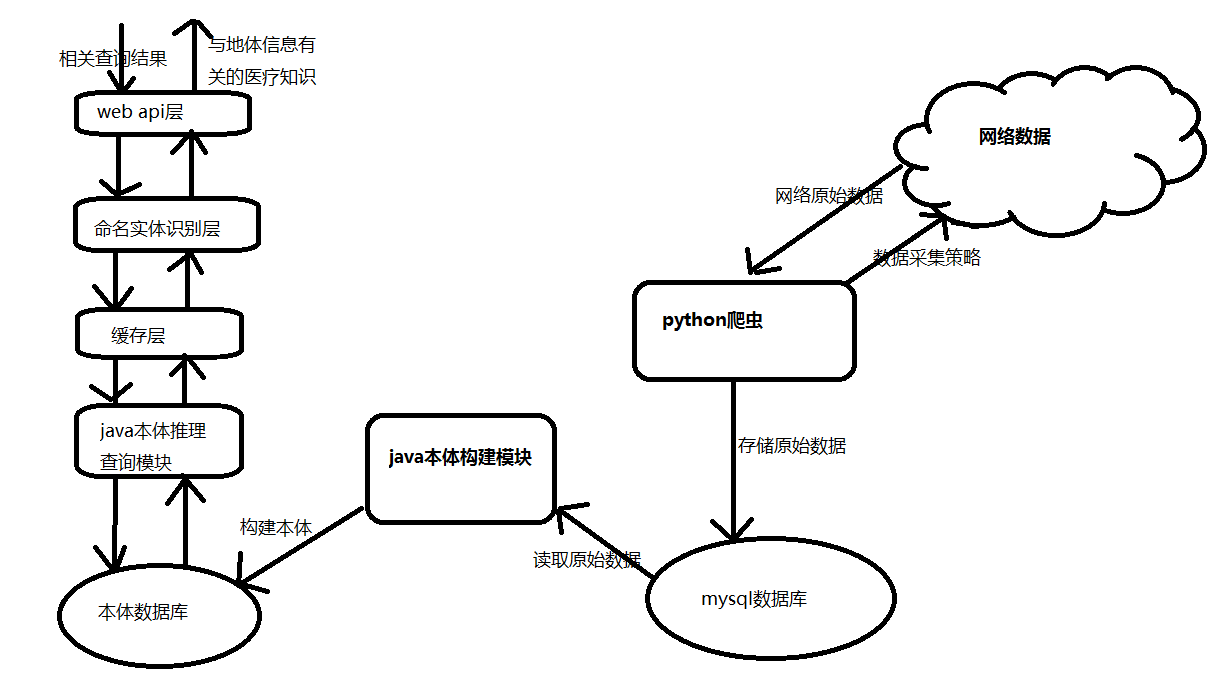
因此，系统开发需要经历两个阶段：数据收集阶段和功能开发阶段。在数据收集阶段，任务是收集构建知识库所需的数据并建立知识库。在功能开发阶段，任务是根据构建的知识库设计推理查询机制并实现，从而达到使用知识库进行基于地理的医疗信息推荐。

综合两个阶段来看，所需的开发环境如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **用途** | **产品** | **版本** | **备注** |
| 服务器系统 | centos | 7.0 | 作为web app的服务器和数据库服务器 |
| 数据库 | mysql | 5.1 | 存储抓取的原始数据 |
| 爬虫开发语言 | python | 3.5 | 作为开发爬虫的程序设计语言 |
| 本体构建与操作语言 | java | 1.8 | 使用java构建本体并开发web接口服务 |
| Web容器 | tomcat | 7.1.3 | 作为java web应用的servlet容器 |

下面分别就两个阶段做深入阐述：

1. 数据收集阶段：该阶段的数据来源主要是网络资源和api调用资源。网络资源包括中国行政区划、中国医疗机构目录、地名百科等等。api资源使用包括地理位置经纬度的确定、各个行政区划人文景点等地理实体数据、大学研究论文数据等。数据采集使用python爬虫实现，对于简单的网页数据直接使用python内置的urllib实现，对于js动态生成的页面数据，无法直接爬取，使用selenium自动化测试框架模拟爬取。爬取结果暂存在mysql数据库中，以便之后构建本体知识库。本体的构建使用java语言实现，使用java的jdbc数据库连接接口和jena工具可以实现将数据库中的数据读取出来并构建成本体。
2. 功能开发阶段：功能开发使用java语言开发，web服务通过mvc模式实现，最终的系统运行在linux服务器环境中。其中，model层的功能是对于构建好的本体库的查询使用jena框架进行查询和推理，并返回推荐知识，view层的功能是展示返回的结果，controller层的功能是调度请求。考虑到开始提供服务之后，知识库内容变更较小，为了提高响应速度，减小数据库和cpu压力，拟采用内存缓存的形式对查询结果进行缓存，然后采取定时过期删除的策略防止缓存与知识库内容不符合。综上，整个系统的架构设计大致如下图所示：



**医院数据获取**

医院及医疗机构数据主要作为与地理相关的信息返回，数据内容包括：医院相关医疗信息（重点科室、研究方向等）、医院地理信息（位置、交通等）、医院专家信息。这些信息源主要选取网络资源，网络上有很多按行政区域划分的中国医院名录。

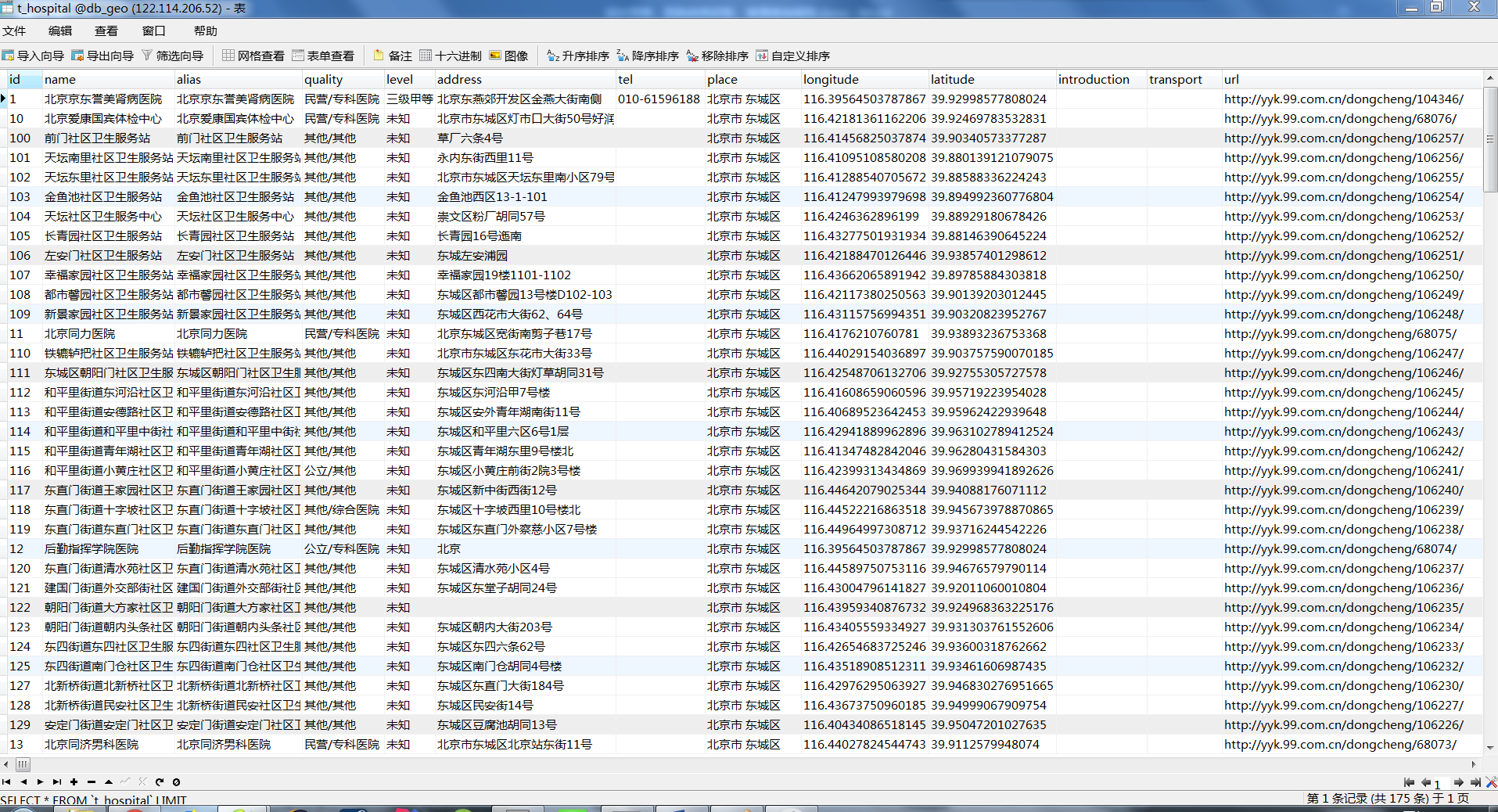
例如：<http://yyk.99.com.cn>



分析页面源码可知，该页面结构简单，抓取容易，可以直接使用爬虫获取相关数据，部分代码如下：



试验抓取部分数据存入数据库中，发现可以简单的实现：



**推理查询规则**

对于构建好的本体库，查询的输入为地理实体、经纬度和疾病名称组成的查询请求，查询结果为相关医疗机构信息、大学研究信息组成的响应。

因此本体数据库的查询过程为：

1. 对于查询请求的每一个经纬度对参数，查询附近一段距离（配置参数）内的医疗机构和大学研究院等地理实体，将这些地理实体按照距离查询点的距离排序形成集合，组成的集合记为S1
2. 对于查询的每一个地理实体名称，找到对应的地理实体对象，若地理实体为poi点，则取该点的经纬度，和①一样，查询附近医疗机构和大学并排序，将查询结果加入S1。否则，将所有行政区划地理实体组成集合T1。
3. 对T2中的行政区划集合进行整理：
4. 对于具有从属关系的行政区划，先查询范围较小的行政区划中的医疗机构等信息，若查询的信息量达到预设的标准（可配置参数），则忽略范围较大的行政区划。否则，对于较大的行政区划执行查询操作，直到没有更高级别的行政区划。
5. 对于不具有从属关系的行政区划，分别查询每个行政区划内的医疗机构、大学研究等数据，并对所有查询出的医疗知识取并集，形成S2。
6. 对S1和S2取并集，形成S3。
7. 按照查询请求中的疾病信息，与S3中每一组医疗机构和大学研究信息计算相似程度并按照一定权重（可配置参数）与之前的排序权重相加，使用最终权重排序。
8. 若S3元素数量大于预设的的查询结果数量N，则按照权重排序取前N条结果，否则返回全部排序结果。