

**《基于知识图谱的自动问答系统》**

**设计说明书**



北京航空航天大学

2016-01

版本变更历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 提交日期 | 主要编制人 | 审核人 | 版本说明 |
| 1 | 2015.12.31 | 方凯 |  | 完成1、2(部分)、3(部分)、 |
| 2 | 2015.01.01 | 杨东东 |  | 完成2(部分)、3(部分) 、  4、5、6(部分)、7 |
| 3 | 2015.01.03 | 李睿霖 |  | 完成4（部分）、5（部分）、6（部分），修改整个文档 |
| 4 | 2016.01.18 | 杨东东 |  | 修改 |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1. 范围 1](#_Toc440893320)

[1.1标识 1](#_Toc440893321)

[1.2系统概述 1](#_Toc440893322)

[1.3文档概述 1](#_Toc440893323)

[1.4术语和缩略词 1](#_Toc440893324)

[1.4.1数据库(Database) 1](#_Toc440893325)

[1.4.2自然语言处理(NLP) 1](#_Toc440893326)

[1.4.3矩阵论(Matrix) 2](#_Toc440893327)

[2. 引用文档 2](#_Toc440893328)

[2.1书籍 2](#_Toc440893329)

[2.2批文 2](#_Toc440893330)

[2.3引用资料 2](#_Toc440893331)

[3. 需求概述 3](#_Toc440893332)

[3.1概述系统特性和需求 3](#_Toc440893333)

[3.2加细节 3](#_Toc440893334)

[3.2.1问题分析 3](#_Toc440893335)

[3.2.2信息检索 3](#_Toc440893336)

[3.2.3答案抽取 3](#_Toc440893337)

[3.3 新需求 4](#_Toc440893338)

[4. 体系结构设计 4](#_Toc440893339)

[4.1总体结构 4](#_Toc440893340)

[4.1.0用例图 5](#_Toc440893341)

[4.1.1活动图 5](#_Toc440893342)

[4.1.2时序图 6](#_Toc440893343)

[4.1.3状态图 6](#_Toc440893344)

[4.1.4类图 6](#_Toc440893345)

[4.2功能分配 7](#_Toc440893346)

[4.3关键问题及解决方案 7](#_Toc440893347)

[4.3.1问题分析模块及解决方案 7](#_Toc440893348)

[4.3.2问题分类模块及其解决方案 8](#_Toc440893349)

[5. 接口设计 8](#_Toc440893350)

[5.1用户界面设计 8](#_Toc440893351)

[5.2外部接口设计. 9](#_Toc440893352)

[5.2.1用户登录 9](#_Toc440893353)

[5.2.2用户注册 10](#_Toc440893354)

[5.2.3获取信息列表 10](#_Toc440893355)

[5.2.4提交信息 12](#_Toc440893356)

[5.2.5用户获取明细 12](#_Toc440893357)

[5.3内部接口设计 13](#_Toc440893358)

[6.数据结构设计 13](#_Toc440893359)

[6.1公共数据结构设计 13](#_Toc440893360)

[6.2数据库设计 13](#_Toc440893361)

[7.详细设计 14](#_Toc440893362)

[7.1用户模块 14](#_Toc440893363)

[7.1.1概述 14](#_Toc440893364)

[7.1.2接口说明 14](#_Toc440893365)

[7.1.3内部设计（静态） 14](#_Toc440893366)

[7.1.4内部设计（动态） 14](#_Toc440893367)

[7.2语音模块 15](#_Toc440893368)

[7.3.1概述 15](#_Toc440893369)

[7.3.2接口说明 15](#_Toc440893370)

[7.3.3内部设计（静态） 15](#_Toc440893371)

[7.2.4内部设计（动态） 15](#_Toc440893372)

[7.3问题分析 15](#_Toc440893373)

[7.3.1概述 15](#_Toc440893374)

[7.3.2接口说明 16](#_Toc440893375)

[7.3.3内部设计（静态） 16](#_Toc440893376)

[7.3.4内部设计（动态） 17](#_Toc440893377)

[7.4 信息检索 17](#_Toc440893378)

[7.4.1概述 17](#_Toc440893379)

[7.4.2接口说明 17](#_Toc440893380)

[7.4.3内部设计（静态） 17](#_Toc440893381)

[7.4.4内部设计（动态） 18](#_Toc440893382)

[7.5答案抽取 18](#_Toc440893383)

[7.5.1概述 18](#_Toc440893384)

[7.5.2接口说明 19](#_Toc440893385)

[7.5.3内部设计（静态） 19](#_Toc440893386)

[7.5.4内部设计（动态） 20](#_Toc440893387)

# 范围

## 1.1标识

文档标识号：A2015-00-02-00

文档标题：<<基于知识图谱的自动问答系统>>--系统设计说明书

版本号：4.0

## 1.2系统概述

基于知识图谱的自动问答系统，是以中文为载体的系统，其数据库为以百度百科、维基百科、互动百科为主，运用其中的知识性信息进行人机交互以达到自动问答的目的的系统。采用目前发展中的实体分词技术、实体消歧技术、语法分析技术、语义分析技术等作为基础，综合开发而成。

系统尚在开发过程中；

投资方：无；

需方：用户；

用户：问答需求方；

开发方：杨东东，李睿霖，方凯；

运行现场：Android系统。

## 1.3文档概述

本文档用于阐述系统概况以及系统设计说明。该文档使用时保密性一般，由于未涉及核心代码，使用时可以半公开。

## 1.4术语和缩略词

### 1.4.1数据库(Database)

MySQL:关联[数据库管理系统](http://baike.baidu.com/view/68446.htm)

Neo4j:高性能的,NOSQL图形数据库

Redis:内存数据库，以高效的查找存储著称，并具有数据持久的特性

### 1.4.2自然语言处理(NLP)

Entity Linking:实体链接

Page Rank:Google开源的一个搜索算法

Entity Ambiguation:实体歧义

Trie Tree:前缀树

kNN: k近邻算法 (kNN，k-NearestNeighbor)

LSA: 隐式语义分析 (Latent Semantic Analysis)

Markov Model:马尔科夫模型

Lucene: 一个开放源代码的全文检索引擎工具包，是一个全文检索引擎的架构

FudanNLP：一个中国国内做得还算不错的NLP处理开源包

### 1.4.3矩阵论(Matrix)

PCA:主成分分析，用于矩阵维度的降维方法 (Principal Component Analysis)

SVD:矩阵奇异值分解 (singular value decomposition method)

[panion](http://cn.bing.com/dict/search?q=Companion&FORM=BDVSP6&mkt=zh-cn) [Matrix](http://cn.bing.com/dict/search?q=Matrix&FORM=BDVSP6&mkt=zh-cn)：伴随矩阵

[ompanion](http://cn.bing.com/dict/search?q=Companion&FORM=BDVSP6&mkt=zh-cn) [Matrix](http://cn.bing.com/dict/search?q=Matrix&FORM=BDVSP6&mkt=zh-cn)：友矩阵

# 引用文档

## 2.1书籍

《软件项目管理》 朱少民，韩莹 编著，人民邮电出版社

《软件项目管理》 Rajeev T Shandilya编著 科学出版社

## 2.2批文

本项目的经核准的计划任务书和上级机关的批文:第九届《大学生创新创业训练计划》

## 2.3引用资料

《<基于知识图谱的自动问答系统>系统需求规格说明书》

《Open Question Answering Over Curated and Extracted Knowledge Bases》

《syntactic constraints on paraphrases extracted from parallel corpora》

《基于维基百科的自动词义消歧方法》 史天艺

《一个中文实体链接语料库的建设》 舒佳根

《基于角色标注的中国人名自动识别研究》张华平 刘群

《基于层叠隐马尔可夫模型的中文命名实体识别》俞鸿魁 张华平 刘群

《基于角色标注的中文机构名识别》俞鸿魁 张华平 刘群

《基于最大熵的依存句法分析》 辛霄 范士喜 王轩 王晓龙

《社区问答系统中若干关键问题研究》 [廉鑫](http://www.cnki.net/kcms/detail/search.aspx?dbcode=CDFD&sfield=au&skey=%e5%bb%89%e9%91%ab&code=25126167;)

《汉语大词汇量连续语音识别系统研究进展》倪崇嘉 刘文举 徐波

# 需求概述

## 3.1概述系统特性和需求

用户发送文字信息给自动问答系统，然后通过自动系统的分析然后返回给用户尽可能使他所想要的信息。该系统从用户发送信息给客户，然后通过自动问答系统中的问题分析模块，信息检索模块和答案抽取模块共三个算法模板检索出准确率最高的答案返回给用户。能够与用户进行聊天，能够回答事实型问题、列举型问题、定义型问题和交互型问题等。

根据这些需求，问答系统需由三个模块组成：问题分析模块，信息检索模块和答案抽取模块。如何在问题分析阶段充分理解用户的提问意图，如何在信息检索阶段把相关文档检索出来，如何在答案抽取阶段准确地把答案从相关文档中抽取出来，这三个方面是问答系统要解决的核心，上图给出了问答系统的经典模型。用例中，用户发送文字信息给自动问答系统，然后通过自动系统的分析然后返回给用户尽可能使他所想要的信息。该系统从用户发送信息给客户，然后通过自动问答系统中的问题分析模块，信息检索模块和答案抽取模块共三个算法模板检索出准确率最高的答案返回给用户。

## 3.2加细节

### 3.2.1问题分析

问题分析是问答系统的第一个模块，它负责对用户提出的中文自然语言问题进行深入细致地理解和分析，为信息检索模块和答案抽取模块奠定良好的基础。问题分析阶段的处理结果直接影响着后续阶段的处理。该阶段的分析主要包括词法分析（分词、词性标注）、问题分类、关键词提取及关键词扩展。其中问题分类是此模块的核心，它能有效减少候选答案的空间，提高系统返回答案的准确率，同时问题的类型也决定了答案抽取中一些策略的选择。由于问句与陈述句的句型结构有着明显差别，为方便在答案抽取时进行相似度计算，必要时还需要对问句进行重写。

### 3.2.2信息检索

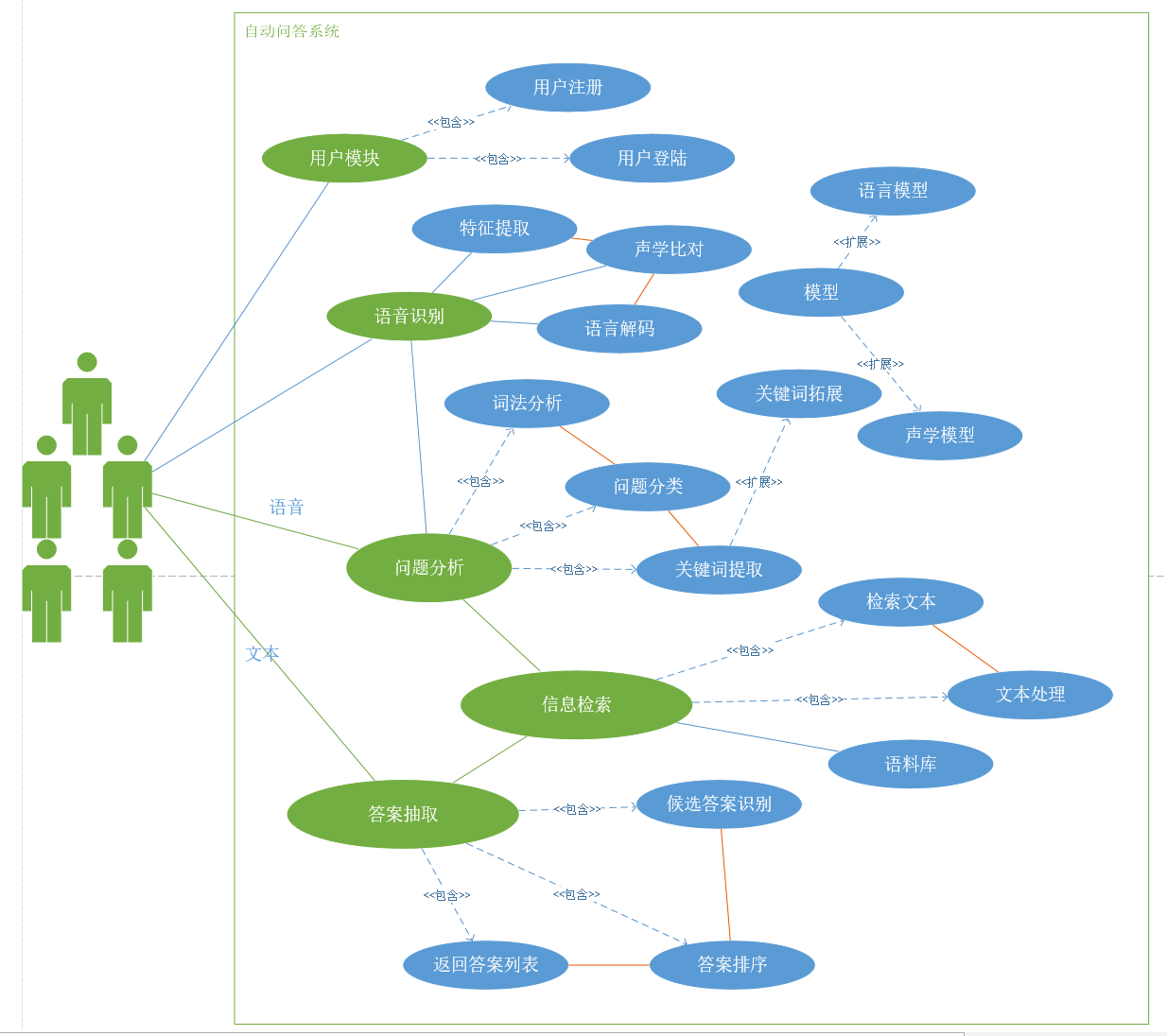
信息检索模块的功能是根据问题分析阶段的关键词提取及扩展生成的查询式在数据库、知识库或互联网[26]上检索所有可能与答案相关的段落、文章或者网页，并做一定的分析处理，将处理结果提交给答案抽取模块进行下一步操作。问答系统中的信息检索同传统的基于关键词匹配的信息检索类似，可以调用现有的比较成熟的各种检索模型。

### 3.2.3答案抽取

答案抽取是问答系统的最终模块，也是最关键的模块。这一模块所应用的答案抽取算法的优劣直接关系着整个问答系统的性能。答案抽取的对象是信息检索模块提交的相关文档，候选答案可以是一句话、一段摘要也可以是一个词，问题类型不同，所得到的答案形式也是不同的，所以答案抽取时还需依据问题类型采取相应的策略。由于信息检索阶段返回的相关文档通常较多，问答系统会抽取出大量的候选答案[27]。因此，候选答案的评估和排序是必不可少的，并最终返回一个最佳答案或按置信度高低排序的答案列表。

## 新需求

1. 之前的部分都对系统如何获取信息，并没有做详细的声明，只是说明了自动问答系统需要一段文本的输入，因此在此，考虑到各种实际情况还有现成的技术，决定将输入自动问答系统的信息媒介允许为语音输入。因此在自动问答系统中需要添加语音转为文字的功能，再进一步接受系统中别的模板的处理。具体的改变的用例模板如图所示。
2. 同时，之前并没有进行用户注册登录等用户信息的模块。此部分功能有成熟的模板可以进行套用，用户方面的信息已经成为了现代的每一个应用所必须会有的，因此我们的系统也具有这样的模块。



# 体系结构设计

## 4.1总体结构

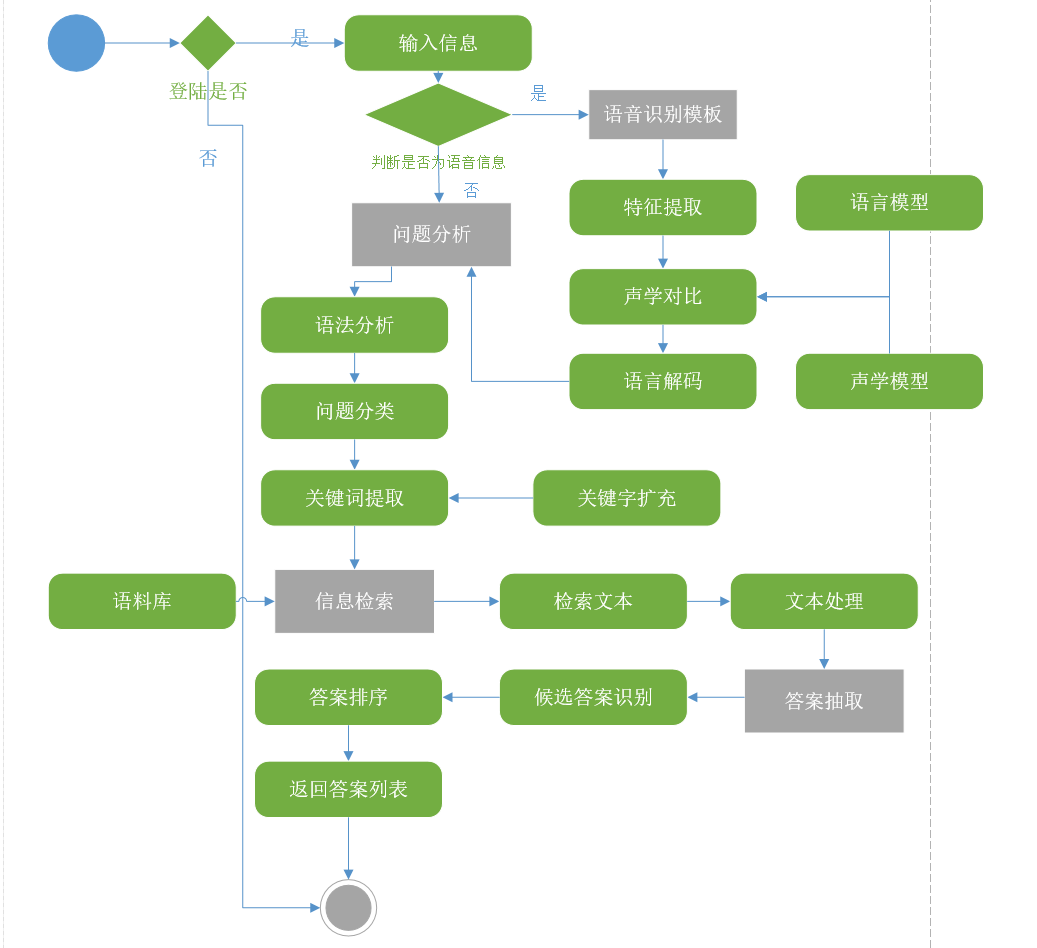
用一系列图表给出系统的总体结构，并对图中的相关内容进行说明。可能包括软件体系结构、硬件体系结构、技术体系结构、支撑体系（部署和实施方案）结构等各个方面，可根据实际情况每个方面分为一小节来写。

采用面向对象分析与设计方法，建议从UML的4+1视图描述软件结构：完善用例图、活动图，给出类图、时序图、状态图。

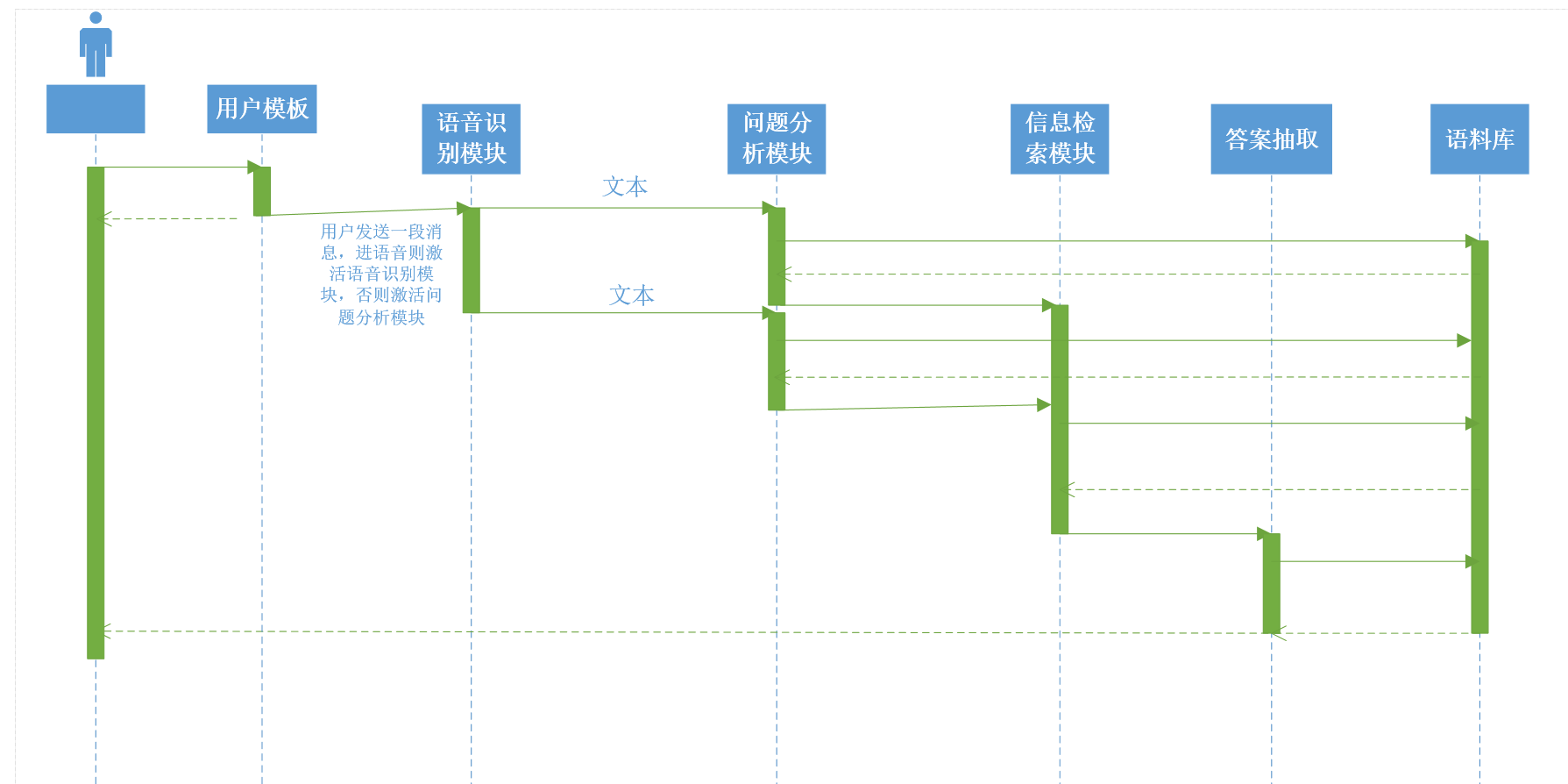
### 4.1.0用例图

如上所示3.3节所示。

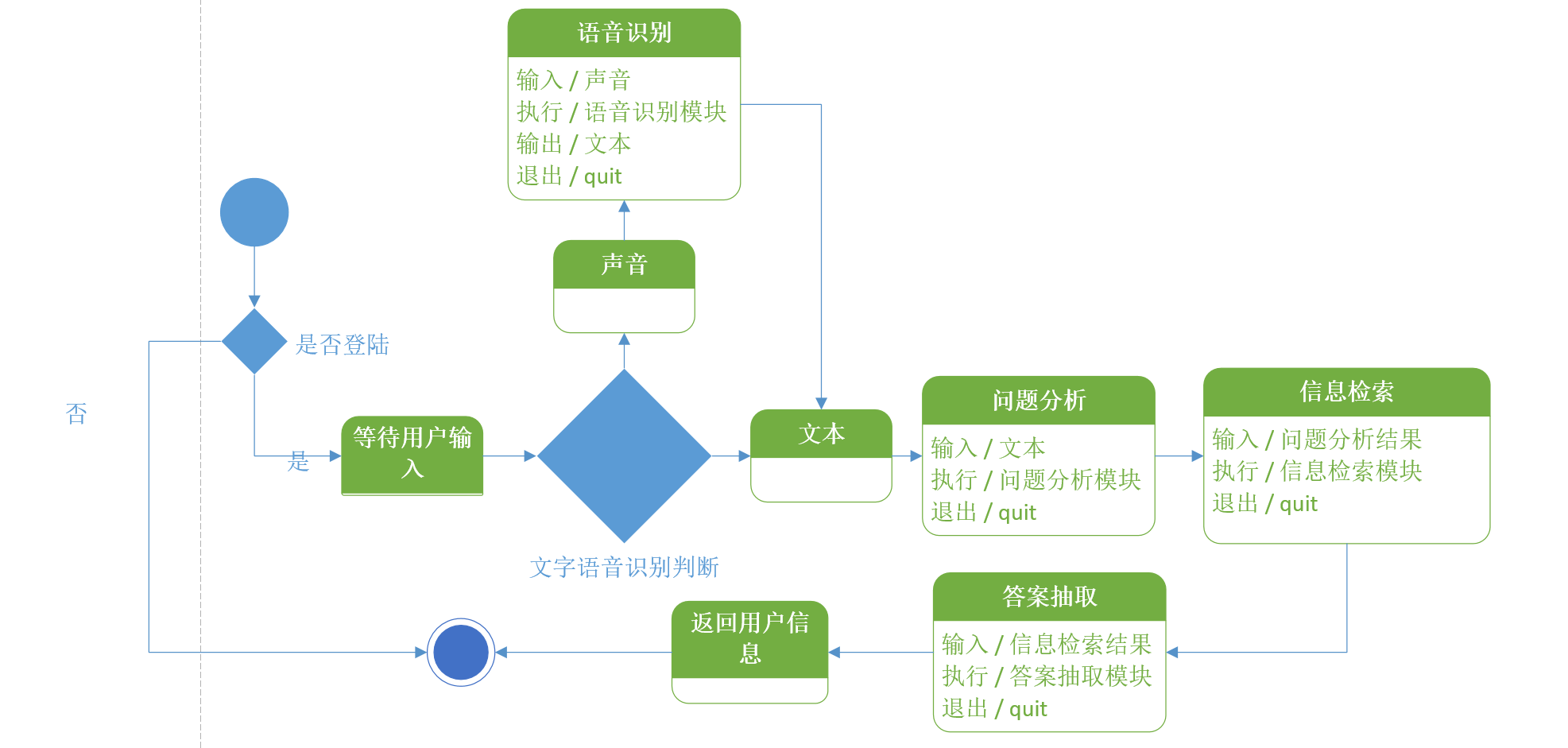
### 4.1.1活动图



### 4.1.2时序图

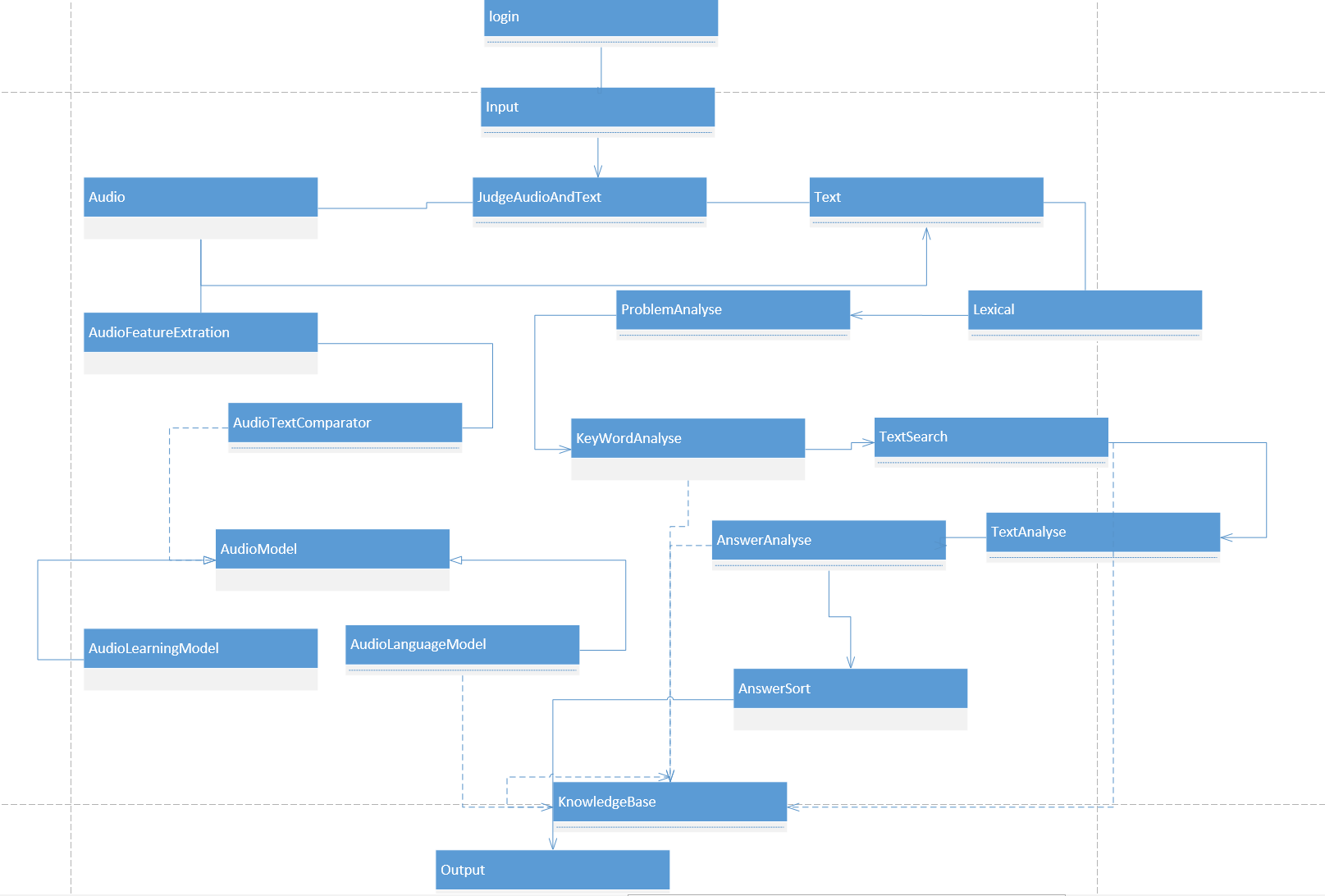


### 4.1.3状态图



### 4.1.4类图

以下类图是从系统层面的类图，是一个具有象征意义的大类。



## 4.2功能分配

说明需求文档当中各项功能要求同总体结构的对应关系。

## 4.3关键问题及解决方案

给出系统总体结构中所涉及的关键问题的设计决策和解决思路。

### 4.3.1问题分析模块及解决方案

由于在用户输入问题中，并非所有的词都能同等的表达相同重要性的含义，因此需要对问题进行预处理，并为后续的问题分类和关键词提取等提供依据。预处理的功能由自然语言处理模块完成，该模块一般包括：中文词法分析模块、命名实体识别模块和句法分析模块。

1. 实现中文词法分析、命名实体识别和句法分析等自然语言处理模块

对用户问句进行分词、词性标注、命名实体识别和句法分析等自然语言处理，可以为问题分类提供依据，同时也使从问句中提取关键词、扩展关键词成为可能。

1. 实现问题分类模块

将用户输入的问题按不同的分类标准归入不同的类别，可以为问题提取关键词提供依据，同时也确定了答案抽取的基本对象。

1. 实现问题关键词提取

根据问题的类别，和自然语言处理的结果，将问题中的关键词提取出来，经过扩展之后构成信息检索的检索词，同时也成为答案排序时的依据。

1. 实现问题关键词的扩展

之所以需要扩展关键词，是因为在正确的答案句子中，很多关键词并不是问题中的关键词，而是问题关键词的同义词、近义词甚至是相关词扩展。

1. 实现查询式的生成

结合一定的启发式规则生成合适的查询，以提高检索模块的召回率。

### 4.3.2问题分类模块及其解决方案

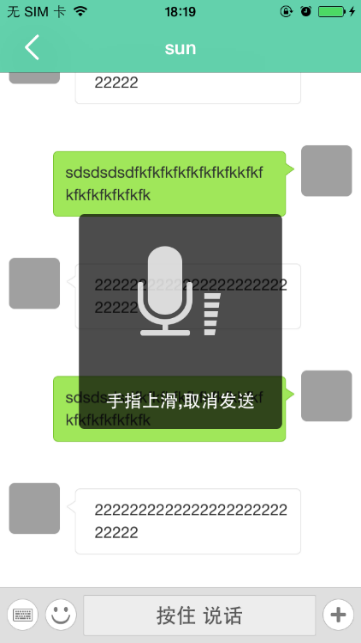
问题分类是问题分析模块的核心，其目的是根据预期的答案类型把问题划分到相应的语义类别，对后续的答案抽取和选择有重要的导向作用。首先，它有效减少了候选答案的空间，其次，决定了答案抽取的策略。该模块的好坏直接影响着问答系统的性能。但是，在一般的中文问答系统中，问题分类是很复杂也是难度很大的领域之一。这主要归因于它与文本分类的两点不同：一、一个问题往往只有一句话，可获取的上下文信息量很少，从而能用于问题分类的特征很有限；二、许多问题并不能严格划分到明确的领域，限制了问题分类的准确性。

# 5. 接口设计

## 5.1用户界面设计

给出系统用户界面的总体设计决策，和典型的用户界面风格。

说明：如果系统提供了对其他系统的接口，如从其他软件系统导入、导出数据，必须在此说明。

图为用户界面友好的聊天界面，支持用户输入模块(包含输入框、录音、表情选择、拍照、从相册选取照片功能)、录音模块。用户可以通过界面发送文字、语音、图片等，进行使用自动问答系统的提问功能和聊天功能。

## 5.2外部接口设计.

对系统的软件和硬件外部接口进行说明。

## 5.2.1用户登录

|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | URL | | |
| 接口名称 | Login | | |
| 接口方向 | 网站后台-->手机端 | | |
| 接口说明 | 用户登录 | | |
|  | **参数名称** | **参数类型** | **参数说明** |
| 输入值类型 | Key | String | Key密钥保护数据 |
| Type | String | 调用接口的名称 |
| Uid | String | 用户名 |
| Pwd | String | 密码 |
| 返回值类型 | 返回结果具体结构如下 | | |
| result | Int | 200成功300失败 |
| 说明 | 返回JSON格式数据 | | |
| 返回值范例 | {  "result": 200,  } | | |

## 5.2.2用户注册

|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | URL | | |
| 接口名称 | MeInsert | | |
| 接口方向 | 网站后台-->手机端 | | |
| 接口说明 | 添加会员 | | |
|  | **参数名称** | **参数类型** | **参数说明** |
| 输入值类型 | Key | String | Key密钥保护数据 |
| Type | String | 调用接口的名称 |
| Uid | String | 用户名 |
| Pwd | String | 密码 |
| Phone | String | 手机 |
| QQ | String | QQ |
| Mail | String | 邮箱 |
| 返回值类型 | 返回结果具体结构如下 | | |
| result | Int | 200成功300失败 |
| 说明 | 返回JSON格式数据 | | |
| 返回值范例 | {  "result": 200,  } | | |

## 5.2.3获取信息列表

|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | URL | | |
| 接口名称 | MesDetail | | |
| 接口方向 | 网站后台-->手机端 | | |
| 接口说明 | 查看商家信息列表 | | |
|  | **参数名称** | **参数类型** | **参数说明** |
| 输入值类型 | Key | String | Key密钥保护数据 |
| Type | String | 调用接口的名称 |
| Field | string | 排序字段（默认“发布时间”） |
| OrderType | String | Desc降序 asc升序 |
| Page | Int | 页数 |
| PageSize | int | 页大小（即返回数据条数） |
| 返回值类型 | 返回结果具体结构如下 | | |
| result | Int | 200成功300失败 |
| PageCount | Int | 总页数 |
| 用户名 | String | 商家的用户名 |
| 店名 | String | 店名 |
| 简介 | String | 简介 |
| 内容 | String | 内容 |
| 价格 | String | 价格 |
| 发布时间 | String | 发布时间 |
| 分类名称 | String | 分类名称 |
| 类型名称 | String | 类型名称 |
| 图片 | String | 图片 |
| 已售 | String | 已售 |
| rowNum | Int | 序号 |
| 说明 | 返回JSON格式数据 | | |
| 返回值范例 | {  "result": 200,  "PageCount": 5,  "MerList": [  {  "rowNum":"1"  "用户名": "用户名",  "店名": "店名",  "简介": "简介",  " 内容": " 内容",  "价格": "价格",  "发布时间": "发布时间",  "分类名称": "分类名称",  "类型名称": "类型名称",  "图片": "图片",  "已售": "已售"  }, {  "rowNum":"2"  "用户名": "用户名",  "店名": "店名",  "简介": "简介",  " 内容": " 内容",  "价格": "价格",  "发布时间": "发布时间",  "分类名称": "分类名称",  "类型名称": "类型名称",  "图片": "图片",  "已售": "已售"  }  ]  } | | |

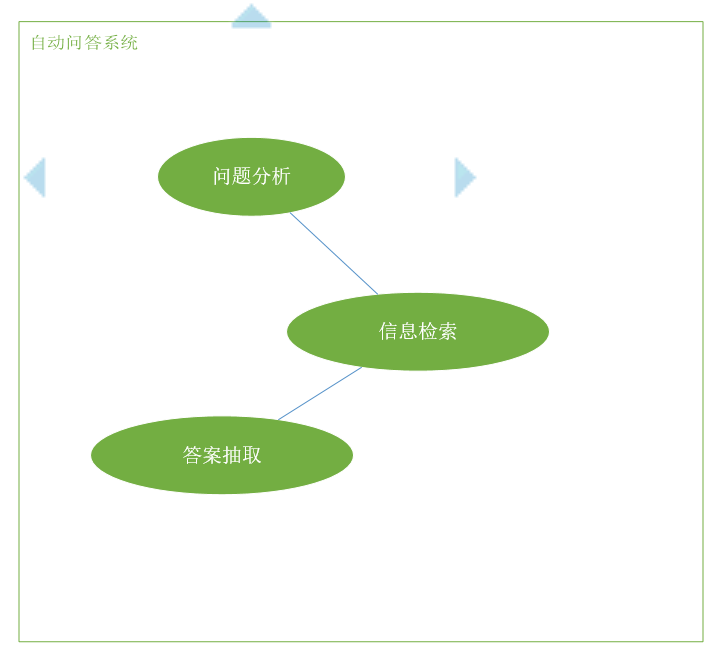
## 5.2.4提交信息

|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | URL | | |
| 接口名称 | OrderDown | | |
| 接口方向 | 网站后台-->手机端 | | |
| 接口说明 | 提交订单 | | |
|  | **参数名称** | **参数类型** | **参数说明** |
| 输入值类型 | Key | String | Key密钥保护数据 |
| Type | String | 调用接口的名称 |
| Josn | String | Josn字符串 |
| 返回值类型 | 返回结果具体结构如下 | | |
| result | Int | 200成功300失败 |
| 说明 | 返回JSON格式数据 | | |
| 返回值范例 | {  "result": 200,  } | | |

## 5.2.5用户获取明细

|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | URL | | |
| 接口名称 | UserVoucherMake | | |
| 接口方向 | 网站后台-->手机端 | | |
| 接口说明 | 查看商家列表 | | |
|  | **参数名称** | **参数类型** | **参数说明** |
| 输入值类型 | Key | String | Key密钥保护数据 |
| Type | String | 调用接口的名称 |
| MxID | String | 明细ID |
| 返回值类型 | 返回结果具体结构如下 | | |
| result | Int | 200成功300失败 |
| PageCount | Int | 总页数 |
| 说明 | 返回JSON格式数据 | | |
| 返回值范例 | {  "result": 200,  "PageCount": 5,  "MerList": [  {  ]  } | | |

## 5.3内部接口设计

对系统各构件（模块）之间的接口进行说明，如下图所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | Set<String> | | |
| 接口名称 | LexicalAnalyse | | |
| 接口方向 | TextInput->LexicalAnalyse | | |
| 接口说明 | 实现中文词法分析、命名实体识别和句法分析,问题分类模块,问题关键词提取,问题关键词的扩展,查询式的生成 | | |
|  | **参数名称** | **参数类型** |
| 输入值类型 | Text | String |
| isAudio | Boolean |
| 返回值类型 | 返回结果具体结构如下 | | |
| result | Set<String> |
| 说明 | 返回下一个模块需要处理的数据集 | | |

问题分析接口：

|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | Set<String> | | |
| 接口名称 | InfoSearch | | |
| 接口方向 | LexicalAnalyse -> InfoSearch | | |
| 接口说明 | 实现爬虫实现,相关性评估, 网页覆盖率共三个部分 | | |
|  | **参数名称** | **参数类型** | **参数说明** |
| 输入值类型 | words | Set<String> | 问题分析结果 |
| 返回值类型 | 返回结果具体结构如下 | | |
| result | Set<String> | |
| 说明 | 返回下一个模块需要处理的数据集 | | |

信息检索接口：

答案抽取接口：

|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | Set<String> | | |
| 接口名称 | AnswerAnalyse | | |
| 接口方向 | 信息检索->答案抽取 | | |
| 接口说明 | 实现候选句过滤；候选答案识别；候选答案排序 | | |
| 输入值类型 | **参数名称** | **参数类型** | **参数说明** |
| words | Set<String> | 检索分析得到的字符集 |
| 返回值类型 | 返回结果具体结构如下 | | |
| result | String | |
| 说明 | 返回用户自动问答系统所计算得到的答案 | | |

# 6.数据结构设计

## 6.1公共数据结构设计

对程序当中所定义的全局变量、常量、全局数据结构或类当中Public数据结构进行定义和说明。

Trie树（字典树）为前缀树，是为了支持泛型、遍历、储存、载入，为自然语言处理中最常用的高效检索数据结构。

## 6.2数据库设计

对程序当中所采用的数据库相关的名称和标识符、在数据库当中的位置、定义、度量单位、格式和值域、敏感程度、数据项名、缩写词和代码，包括用于数据库的规格说明等进行定义。建议给出数据库表结构。

说明：数据库表结构可以参考如下，字段值域不够可以自己增加。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据库的表结构 | 表名称 | 设计者 | 审核者 | 完成日期 |
|  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 字段代号 | 名称 | 类型 | 值域 | 数据项名 | 索引或键 | 备注 | 缩写词 |
|  |  |  |  |  |  |  |  |

文档数据库

主要使用文档进行初步存储。其中词典有两个形态：文本文件(filename.txt)和缓存文件(filename.txt.bin或filename.txt.trie.dat和filename.txt.trie.value)。

文本文件

采用明文储存，UTF-8编码，CRLF换行符。

缓存文件

就是一些二进制文件，通常在文本文件的文件名后面加上.bin表示。有时候是.trie.dat和.trie.value。后者是历史遗留产物，分别代表trie树的数组和值。

修改了任何词典，只有删除缓存才能生效。

词典说明：

词典分为词频词性词典和词频词典。

词频词性词典

每一行代表一个单词，格式遵从[单词] [词性A] [A的频次] [词性B] [B的频次] 。

词频词典

每一行代表一个单词，格式遵从[单词] [单词的频次]。

每一行的分隔符为空格符或制表符

少数词典有自己的专用格式，比如同义词词典兼容《同义词词林扩展版》的文本格式，而转移矩阵词典则是一个csv表格。

redis数据库表的设计

# 7.详细设计

每一小节给出一个模块（构件）的详细设计方案。包括模块概述、模块的接口说明（即输入、输出）、以及内部结构设计。其中内部结构又可以考虑从静态、动态结构两个方面阐述；静态结构应给出该模块（构件）的类结构（类图），动态结构应给出该模块关键业务流程的交互模型（顺序图），还可根据实际情况给出状态图（某个构件或对象的状态迁移）和活动图（某个算法的实现流程）等内容。

## 7.1用户模块

### 7.1.1概述

### 7.1.2接口说明

### 7.1.3内部设计（静态）

该模块（构件）的类结构（类图）

### 7.1.4内部设计（动态）

关键业务流程的交互模型（顺序图），还可根据实际情况给出状态图（某个构件或对象的状态迁移）和活动图（某个算法的实现流程）等内容。

## 7.2语音模块

### 7.3.1概述

### 7.3.2接口说明

### 7.3.3内部设计（静态）

该模块（构件）的类结构（类图）

### 7.2.4内部设计（动态）

关键业务流程的交互模型（顺序图），还可根据实际情况给出状态图（某个构件或对象的状态迁移）和活动图（某个算法的实现流程）等内容。

## 7.3问题分析

### 7.3.1概述

问题分析是问答系统的第一步，负责对用户输入的问题进行充分地理解。对于中文问答系统来说，这个任务是相对复杂而又困难的。问题分析模块主要实现以下几个功能：

实现中文词法分析、命名实体识别和句法分析等自然语言处理模块：对用户问句进行分词、词性标注、命名实体识别和句法分析等自然语言处理，可以为问题分类提供依据，同时也使从问句中提取关键词、扩展关键词成为可能。

实现问题分类模块：将用户输入的问题按不同的分类标准归入不同的类别，可以为问题提取关键词提供依据，同时也确定了答案抽取的基本对象。

实现问题关键词提取：根据问题的类别，和自然语言处理的结果，将问题中的关键词提取出来，经过扩展之后构成信息检索的检索词，同时也成为答案排序时的依据。

实现问题关键词的扩展：之所以需要扩展关键词，是因为在正确的答案句子中，很多关键词并不是问题中的关键词，而是问题关键词的同义词、近义词甚至是相关词扩展。

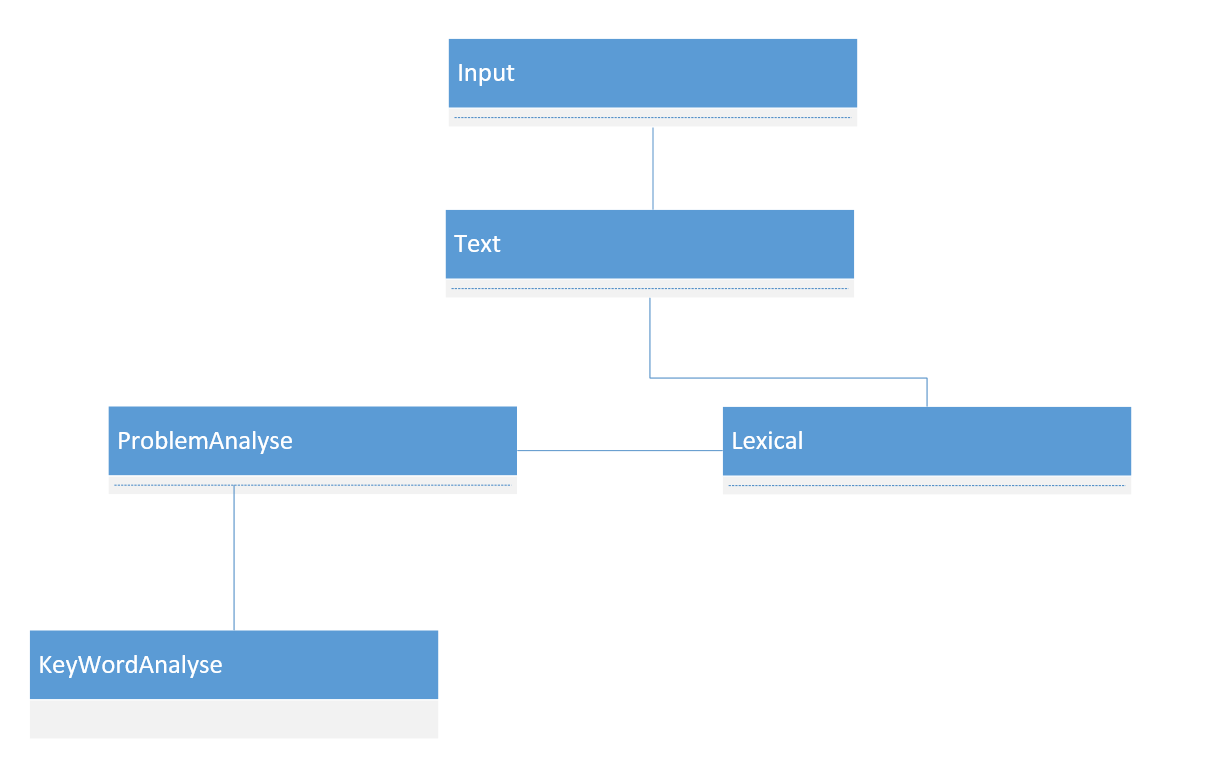
实现查询式的生成：结合一定的启发式规则生成合适的查询，以提高检索模块的召回率。

### 7.3.2接口说明

|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | Set<String> | | |
| 接口名称 | LexicalAnalyse | | |
| 接口方向 | TextInput->LexicalAnalyse | | |
| 接口说明 | 实现中文词法分析、命名实体识别和句法分析,问题分类模块,问题关键词提取,问题关键词的扩展,查询式的生成 | | |
|  | **参数名称** | **参数类型** | **参数说明** |
| 输入值类型 | Text | String | 输入的文本信息 |
| isAudio | Boolean | 是否为语音信息的记录 |
| 返回值类型 | 返回结果具体结构如下 | | |
| result | Set<String> | 返回 |
| 说明 | 返回下一个模块需要处理的数据集 | | |

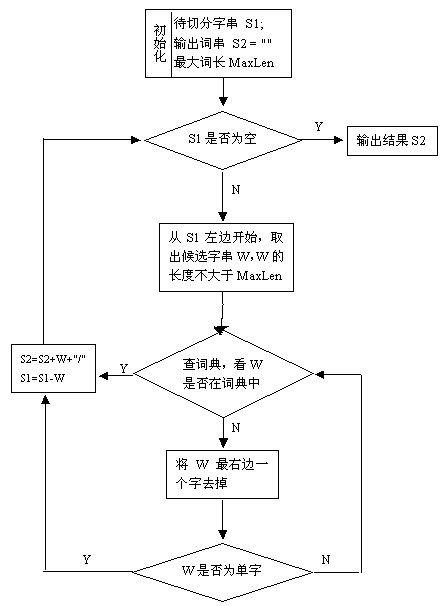
### 7.3.3内部设计（静态）

该模块（构件）的类结构（类图）



### 7.3.4内部设计（动态）

关键业务流程的交互模型（顺序图），还可根据实际情况给出状态图（某个构件或对象的状态迁移）和活动图（某个算法的实现流程）等内容。

下面是例举的分析问题模块中重要的算法流程图：

## 7.4 信息检索

### 7.4.1概述

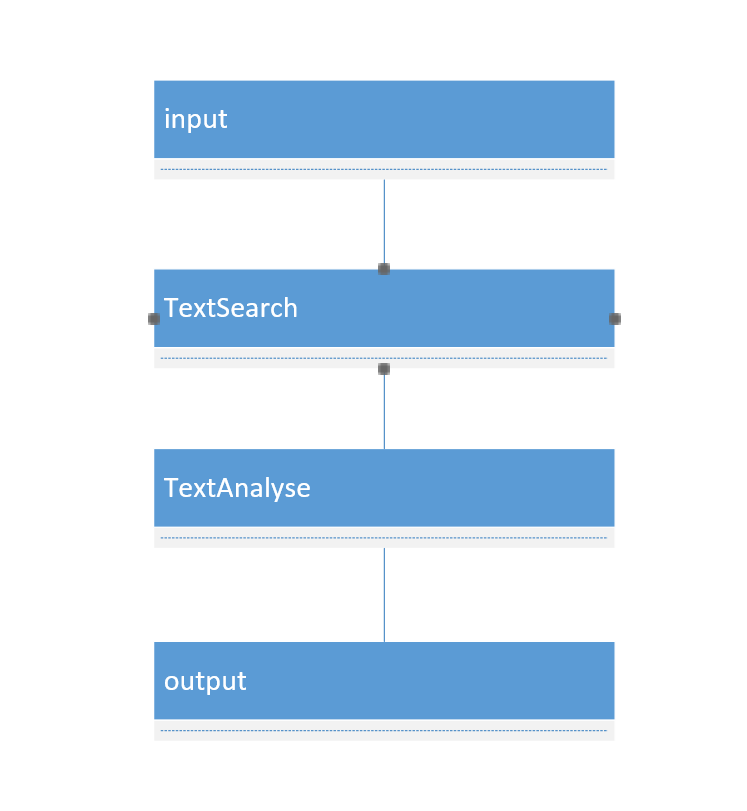
对于开放域问答系统，比较常用的方法是将整个网络看作一个半结构化文档的知识库，利用搜索引擎处理和过滤信息。在本系统的实现中，信息检索模块也是直接调用Internet上已有的搜索引擎：Google和百度，并将返回结果中的网页片断(Snippets)作为答案抽取的资源。

网络搜素引擎的运行机制是，首先由网页获取器Spider根据地址管理器中的地址在互联网上进行搜索网页，然后将获得的不同格式的网页进行格式统一化处理，接着对处理后的网页进行索引化管理，并将其放入索引数据库中。当用户进行检索信息时，检索器根据用户的检索请求对网页进行相关度的计算，并对结果进行排序，最后将排序结果返回给用户。

### 7.4.2接口说明

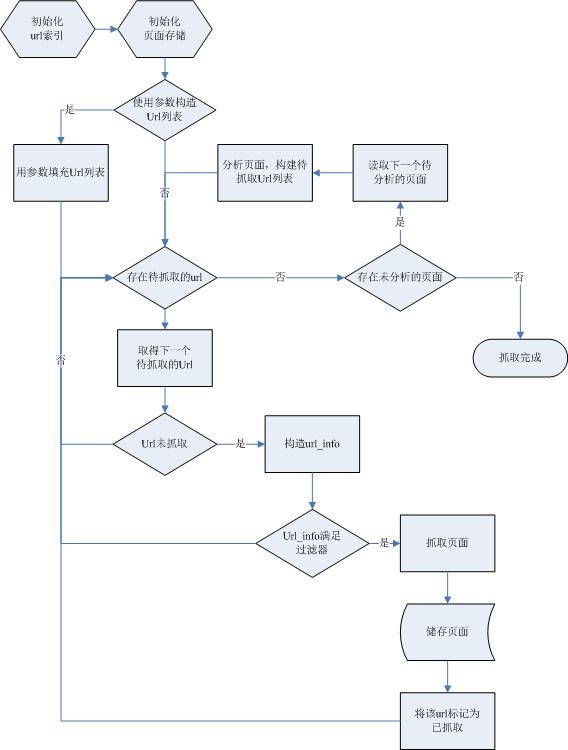
|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | Set<String> | | |
| 接口名称 | InfoSearch | | |
| 接口方向 | LexicalAnalyse -> InfoSearch | | |
| 接口说明 | 实现爬虫实现,相关性评估, 网页覆盖率共三个部分 | | |
|  | **参数名称** | **参数类型** | **参数说明** |
| 输入值类型 | words | Set<String> | 问题分析结果 |
| 返回值类型 | 返回结果具体结构如下 | | |
| result | Set<String> | |
| 说明 | 返回下一个模块需要处理的数据集 | | |

### 7.4.3内部设计（静态）

该模块（构件）的类结构（类图）

### 7.4.4内部设计（动态）

关键业务流程的交互模型（顺序图），还可根据实际情况给出状态图（某个构件或对象的状态迁移）和活动图（某个算法的实现流程）等内容。



## 7.5答案抽取

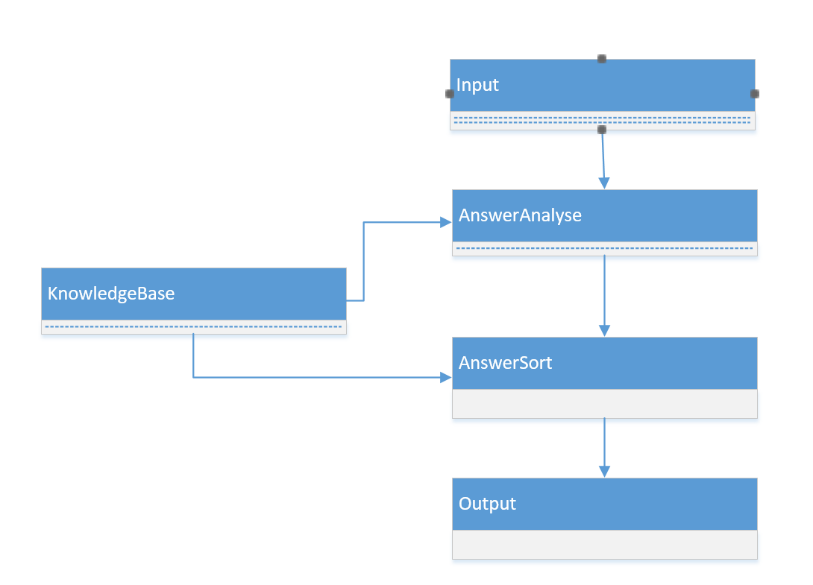
### 7.5.1概述

答案抽取模块是问答系统的最终模块，也是最核心的模块，它的主要工作是根据问题的类型从信息检索模块所得到的结果中找出相关的词或短语作为问题的答案返回给用户。答案抽取模块性能的优劣直接影响着整个问答系统的性能。在本系统中，答案抽取分以下几个步骤：候选句过滤；候选答案识别；候选答案排序。

### 7.5.2接口说明

|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | Set<String> | | |
| 接口名称 | AnswerAnalyse | | |
| 接口方向 | 信息检索->答案抽取 | | |
| 接口说明 | 实现候选句过滤；候选答案识别；候选答案排序 | | |
| 输入值类型 | **参数名称** | **参数类型** | **参数说明** |
| words | Set<String> | 检索分析得到的字符集 |
| 返回值类型 | 返回结果具体结构如下 | | |
| result | String | |
| 说明 | 返回用户自动问答系统所计算得到的答案 | | |

### 7.5.3内部设计（静态）

该模块（构件）的类结构（类图）

### 7.5.4内部设计（动态）

关键业务流程的交互模型（顺序图），还可根据实际情况给出状态图（某个构件或对象的状态迁移）和活动图（某个算法的实现流程）等内容。

### 模式匹配

由于目前中文命名实体识别可识别的命名实体类型有限，只有一些人名、地名、机构名等等，不能满足本文问答系统所有问题类型要求的预期答案形式。因此有必要针对某些问题类型设计相应的答案模式，利用模式匹配的方法在候选句中抽取答案。下面列出了本系统的部分答案模式集，分别是具体时间（Time）类、星期（Week）类、百分比（Percent）类：

Time：\*点｜\*时｜\*点\*分｜\*时\*分

Week：周\*｜星期\*｜礼拜\*

Percent：\*分之\* | 百分之\* | \*%

答案模式集主要是根据问题类型的常见答案句结构而建立的[ ]，其中“\*”是通配符，但是对于不同类型的问题，通配符常常有一定的词性要求。如果候选句能与相应问题类型的某个答案模式匹配，则匹配位置代表一个候选答案。

具体算法步骤如下：

|  |
| --- |
| 1. 根据问题类型取出所有相应的可选答案模式，形成队列，并在所有可选答案模式及当前候选句末尾各加一个结束标志； |
| 1. 取出第一个可选答案模式； |
| 1. 设置答案模式字符串指针AP和候选句字符串指针CP分别指向当前答案模式的第一个字符和候选句的第一个字符，初始化候选答案匹配位置MP=-1； |
| 1. 若AP指向结束标志，转7）；否则，从答案模式的AP位置起逐个向右读取字符，直至遇到第一个“\*”标志或结束标志（记录此位置为EP），将答案模式从AP到EP位置所构成的子串赋给一个临时字符串变量ss，并置AP=EP+1，若EP位置对应结束标志，则AP=AP-1； |
| 1. 若CP指向结束标志，转9）；否则从候选句的CP位置起逐个向右读取字符，并将临时字符串ss与候选句进行匹配，若匹配（当ss以“\*”结尾时，要求候选句中与“\*”对应位置内容不为空且词性与答案模式在此位置要求的词性相符），转6）；否则，若MP==-1转9），否则转8）； |
| 1. 令CP为候选句最后一个匹配字符的下一个位置；若MP==-1，则令MP=匹配的第一个字符位置，转4）； |
| 1. 保存候选答案的匹配位置及匹配的答案模式序号，重置MP =-1； |
| 1. 若CP未指向候选句结束标志，则重置AP指向当前答案模式的第一个字符，重置MP=-1，转4）； |
| 1. 取下一个可选候选答案模式，转3）；若无可选答案模式，输出所识别的该候选句中全部候选答案及匹配位置； |