**复旦大学计算机科学技术学院**

**2019-2020学年第一学期期末论文课程评分表**

**课程名称：** 自然语言处理 **课程代码：** COMP130141.01

**开课院系：**  计算机科学技术学院

**学生姓名：** 吴茜茜 **学号：** 17300200036 **专业：**  计算机科学与技术

**论文名称：** 基于知识图谱的复旦选课问答系统

**成绩：**

|  |
| --- |
| **论文评语（教师填写）：**  **任课教师签名：**  **日 期：** |

**目 录**

**1 选题意图**

**2 系统框架**

**3 系统搭建**

3.1 知识图谱建立

3.1.1 数据来源

3.1.2 效果展示

3.2 问题与模版设计

3.3 推理引擎

3.3.1 基于朴素贝叶斯方法的问题分类器

3.3.2 基于SnowNLP的情感分析

3.3.3 分词与问题匹配

3.3.4 图数据库查询

**4 实验**

4.1 问题分类器实验效果

4.2 针对课程评价的情感分析

4.2.1 课程评价数据获取

4.2.2 课程评价的文本分析

4.2.3 情感分类器训练

4.3 系统运行效果

4.3.1 运行各个问题效果示例

4.3.2 聊天式界面搭建

**5 总结**

**6 附录**

6.1 项目文件列表

6.2 系统运行方法

**基于知识图谱的复旦选课问题系统**

吴茜茜 17300200036

**Abstract**

选课是复旦同学们比较关心的问题。一方面，由于教务系统在选课方面提供的查询较少，而且为搜索式界面，同学们查询课程的便捷性和友好性不够；另一方面，许多同学在选课时难以获得针对该课程的评价（如给分、平时任务量等）。本项目通过收集复旦大学教务处提供的课程信息和公众号“复小七”的课程评价信息，基于自然语言处理中的文本分类、情感分析、分词、关键词抽取和知识图谱等技术，建立了复旦大学选课问答系统。用户可以提问关于院系、教师的各类授课信息以及具体课程的评价情况，共计十类问题。最后，使用Flutter建立了本系统的手机端界面。项目代码、数据及运行方法见附录部分。

**1 选题意图**

选课是复旦同学非常关心的问题。如下图，选课系统提供的搜索界面中查询条件较少，同学们难以根据开课院系与授课教师为条件查询。另外，针对某门课程的评价信息也难以获得，比如对于模块课、选修课，同学们比较关心的任务量重不重、平时的点名情况等。在日月光华BBS，民间选课交流群以及选课公众号“复小七”中都有一些课程的评价信息，但是比较分散，给同学们的查询带来不便。

因此，本项目旨在解决选课信息查询中的不便，通过对话的方式，获得用户的查询问题内容后进行分析，将结果以文字回答的方式提供给用户。

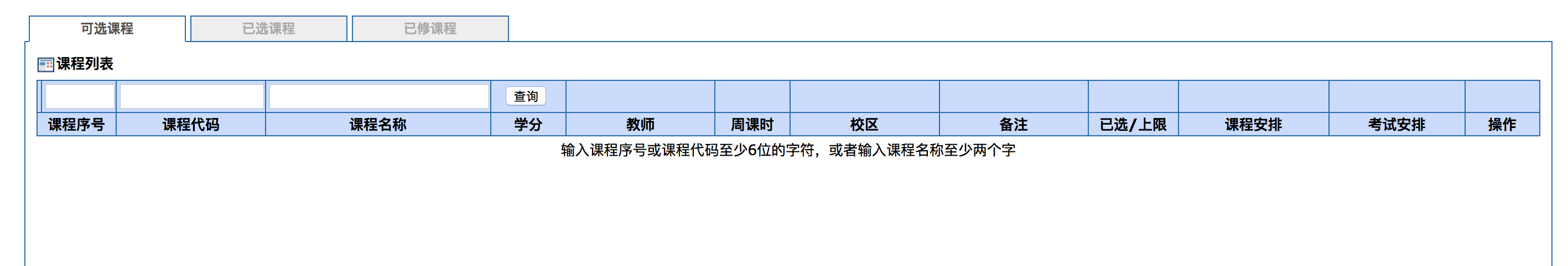


图1. 复旦大学选课系统界面

**2 系统框架**

整个系统的逻辑如图2所示。

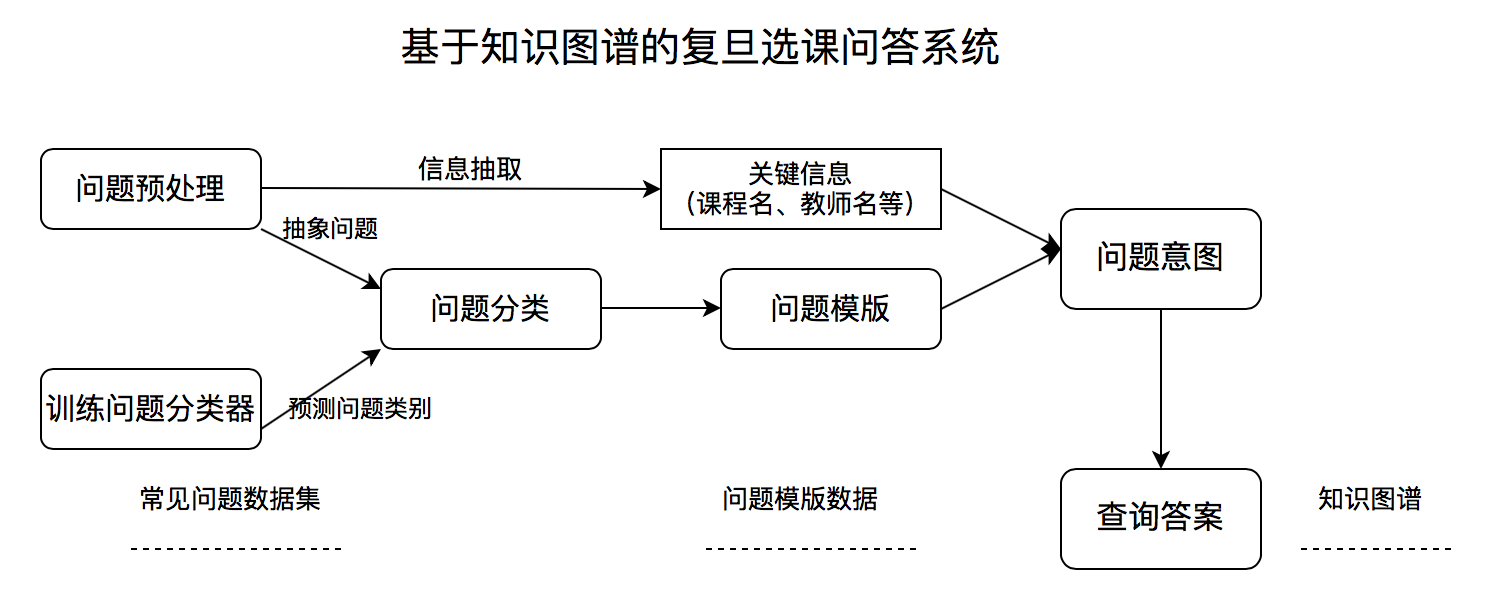


图2. 项目工作流程

* **问题预处理：**

我们需要对用户的问题进行预处理。首先需要获取问题中的询问对象，也就是主语，是教师、开课院系还是课程，这涉及到自然语言处理中的命名实体识别。接着我们需要了解用户的意图，这通过与训练一个问题分类器来识别。

* **抽取关键信息：**

对于关键信息抽取中的命名实体识别，在项目实验过程中，我采用的词性标注的方法提取。增加对人名（授课教师）、院系（开课院系）等的标注，可以把关键信息抽取出来。

* **问题模版：**

在训练问题分类器中，将用户的问题归纳为许多类，如：某位老师上的课、某个学院开设的某种类型的课程等。在这个模块下，我们需要对各个类别进行抽象，比如对于询问某位老师教的课，我们可以把它抽象成：



* **查询答案：**

了解用户意图后，我们就可以根据用户的意图进行查询，得到答案后返回，这部分主要是对图数据库进行查询操作。部分问题需要对查询结果进行进一步筛选，比如查询评价不错的课，需要对查询结果中的课程评价使用一个预训练的情感分类器，筛选出情感偏向积极的课程。

**3 系统搭建**

**3.1 知识图谱建立**

知识图谱以三元组表示实体间关系，常见的存储方式有：基于RDF的存储和基于图数据库的存储。本项目采用**Neo4j**图数据库存储，存储信息包括院系、课程、授课教师、课程类型。

**3.1.1 数据来源**

本项目的课程信息来自复旦大学教务处**[[1]](#footnote-1)**中的课程表，虽然评教网站**[[2]](#footnote-2)**有各门课程的一些信息，但是缺乏具体的评价文本和对应的课程类型。因此数据集构建是人工完成，我根据自己和周围同学的上课经历，选择了七大模块、通识教育选修、思想政治、大学英语、大学体育等共计**7**个类型的课程，共**56**门课，涉及**12**个院系，**60**位教师，并对每一门课作出了文字评价。“专业必修”和“专业选修”类的课程我都选择的是计算机学院开始的课程。

因为人工建立数据集需要一定的时间，所以本项目数据集略小。需要说明的是，整个项目的技术实施并不依赖于数据集的规模，与数据规模相关的只有最后一步执行的在数据库中的查询操作。

**表1.原始数据信息**

|  |  |  |  |
| --- | --- | --- | --- |
| 文件名 | 存储内容 | 属性 | 数量 |
| Course.csv | 课程 | 课程编号、课程名、评价、开课院系 | 56 |
| Teacher.csv | 教师 | 教师id、名字 | 60 |
| School.csv | 院系名 | 院系id,院系名 | 12 |
| Course\_Type.csv | 课程类型 | 课程类型id,课程类型名 | 7 |
| Course2Type.csv | 课程与对应类型的关系 | 课程编号，课程类型id | 56 |
| School2Course.csv | 院系与开设课程的关系 | 院系id，课程编号 | 56 |
| Teacher2Course.csv | 教师授课关系 | 教师id，课程编号 | 56 |
| Teacher2School.csv | 教师与所属院系关系 | 教师id,院系id | 60 |

**3.1.2 效果展示**

确定好数据集后，我们将原始数据导入图数据库中。这里课程、课程类型、院系教师都是实体，因此在图数据库中以创建节点的方式建立，代码如下：



而对于关系型数据，通过导入关系的方式创建节点间的关系，代码如下：



创建后，该图数据库的基本信息如下，即包含四种类型的节点（教师、院系、课程、课程类别），四种类型的关系（教师授课、院系开课、课程类型、教师所属院系）。

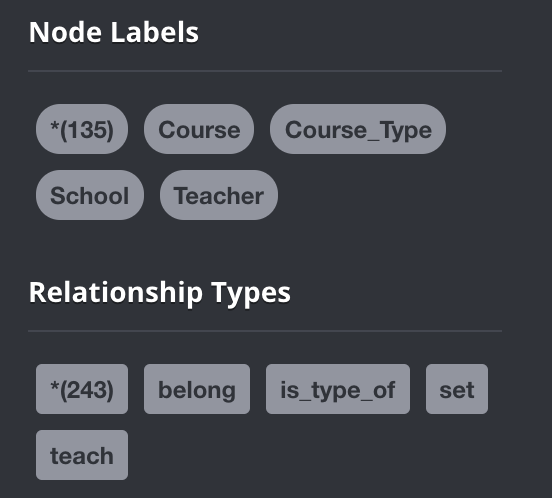
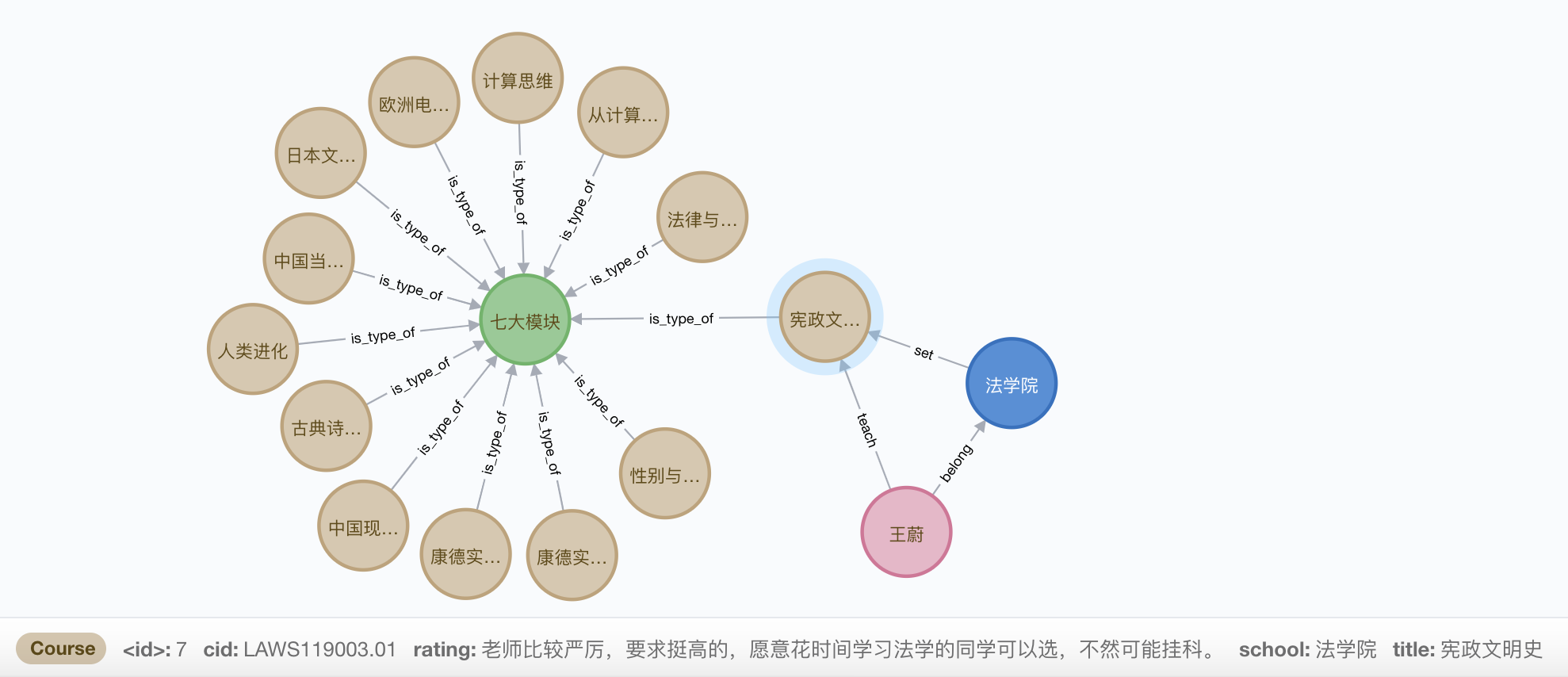


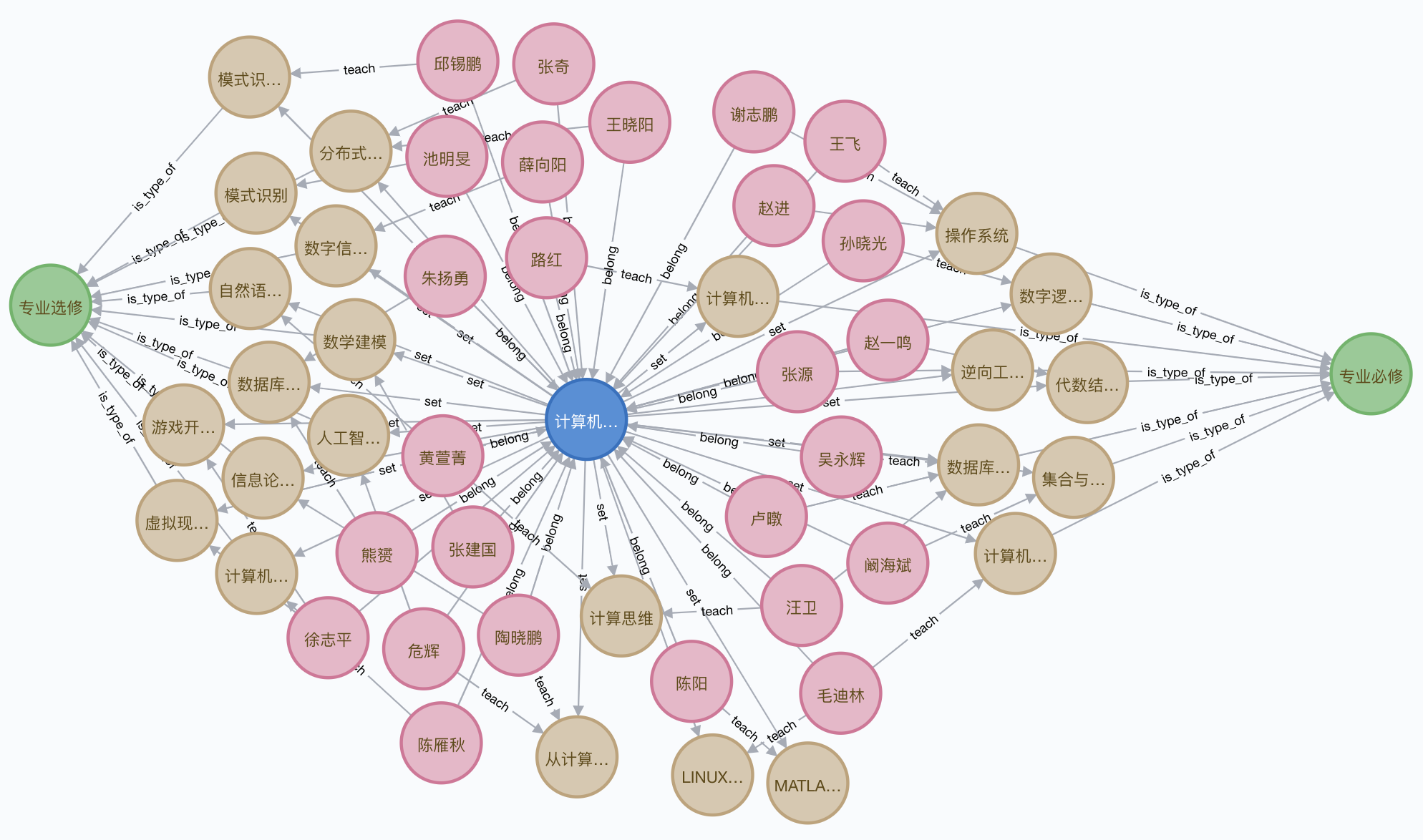
图3.图数据库的基本信息

* **Sample 1 : 课程《宪政文明史》**

如图4所示，该图数据库中展示了该门课所属类型为“七大模块”，同时罗列了其他七大模块课程；展示了该课程的教师与开课院系，并标出了教师与院系间的关系。



* **Sample 2 : 院系-计算机科学技术学院**



**3.2 问题与模版设计**

在系统里，我自定义的词性标注为：



提供的问题包括：

* 某门课程的给分情况
* 某位老师教的课程名
* 某位老师的教课数量
* 某门课程的类型
* 某位老师教的某种类型的课程
* 某个学院开设的课程
* 某个学院开设的某种类型的课程
* 某个学院开设的评价不错的某种类型的课程
* 两位老师一起教的课程
* 某个类型里评价不错的课程

共计十类问题，对应的模版如下：



**3.3 推理引擎**

对于用户提出的问题，我们的工作步骤是：对原始问题进行词性标注、实体提取，之后将原始问题转化为部分实体以标注的词性替代的问题。对于该问题，我们通过预训练一个问题分类器来识别该问题对应的问题模版，最终在图数据库中执行查询操作，将结果返回给用户。

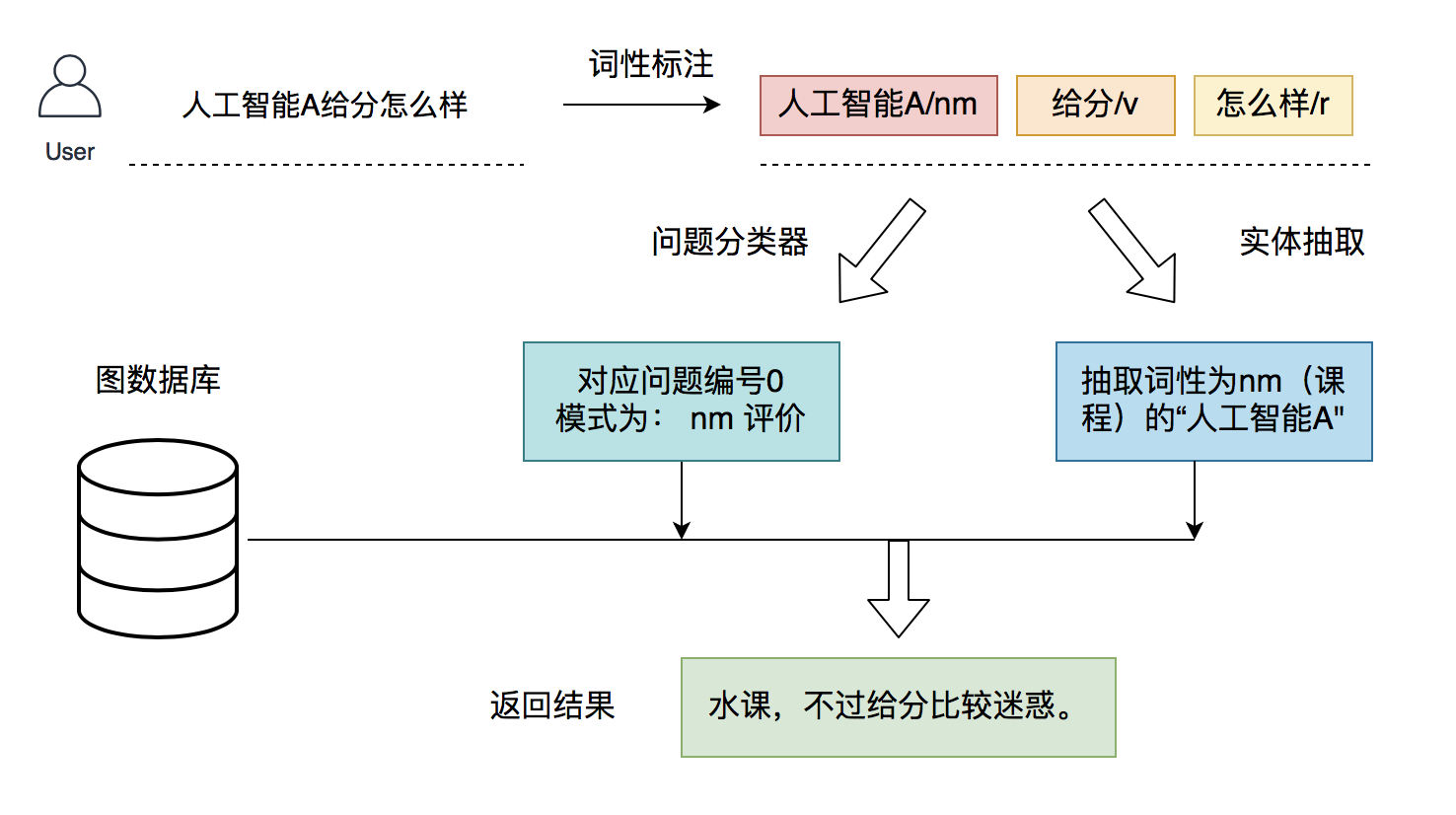


图5.推理过程

**3.3.1 基于朴素贝叶斯方法的问题分类器训练**

由于提供十类问题，对每一类问题，都有不同的问法，因此考虑建立一个分类器来识别用户提出的问题。该分类器的训练数据即是每一类问题的各种问法，和对应的问题号为标签。问题0号对应某门课程的评价情况，我提供的训练数据如下：



这样一共十类问题，每个问题的输入都是这个问题的所有可能的问法。由此我们得到了一组带有标注的训练数据。使用朴素贝叶斯分类器进行问题分类。

**3.3.2 基于SnowNLP的情感分析器**

由于本系统提供查询评价不错的课程的功能，因此需要判断哪些课程为“评价不错”。这里我考虑预训练一个情感分类器。通过收集大量课程评价信息，并进行标注，得到一定规模的正负样本，训练出一个对课程评价的情感分类器。

之后，我们可以直接调用这个情感分类器来判断课程是否属于“评价不错”范畴，将该门课的评价信息输入到训练好的情感分析器中，输出的结果越接近1，则说明评价越好。

**3.3.4 分词与问题匹配**

由于我对课程名、教师名自定义了词性类型，因此在调用jieba进行词性标注前，需要加入这些标记词的信息，部分标注如下图所示。另外，与课程评价相关的信息，如“给分”，“无限B+”,“CD刻录机”等也要加入。



之后，对于原始问题，我采用jieba进行分词，得到该句话的词性标注。将部分信息替换为词性标注（如将“人工智能A”课程名替换为标注“nm”），得到的新问题输入问题分类器，输出即为该问题对应的问题模版编号0。

**3.3.4 图数据库查询**

对提供的十类问题，每一类问题的解决思路如下表所示：

**表2.每类问题对应的查询思路**

|  |  |  |
| --- | --- | --- |
| 问题号 | 内容 | 查询方法 |
| 0 | 某课程给分情况 | 查询课程名为用户输入的课名，返回其评价内容 |
| 1 | 某位老师所教课程 | 查询该教师所教授的课程，返回所有课程名 |
| 2 | 某门课的类型 | 查询名称为该课名的课程，返回其对应类型 |
| 3 | 某位老师教的某种类型的课程 | 查询该教师所教的全部课程，对于每一门课程，查询该门课的类型，只保留类型为用户意图类型的课程。 |
| 4 | 某个院系开设的课程 | 查询该院系所开设的课程，返回所有课程的名字 |
| 5 | 某个院系开设的某种类型的课 | 查询该院系所开设的课程，对于每一门课，查询该门课的类型，只保留类型为用户意图类型的课程 |
| 6 | 教师a与教师b共同上的课 | 分别查询教师a与教师b所教的课，  返回的课程名为两个集合的交集。 |
| 7 | 某位教师的授课数量 | 查询该教师所教的课，返回查询结果的长度 |
| 8 | 某个学院开设的某个类型的给分不错的课 | 在5的基础上，使用预训练的模型计算该门课的评价内容的情感得分，取大于某一阈值（如0.5）的为结果 |
| 9 | 某个类型的课程中评价不错的课 | 查询该类型对应的所有课程，对每一门课，使用预训练的模型计算该门课的评价的情感得分，取大于某一阈值（如0.5）为结果 |

具体的查询语句为（以查询某门课程的评价情况为例）：



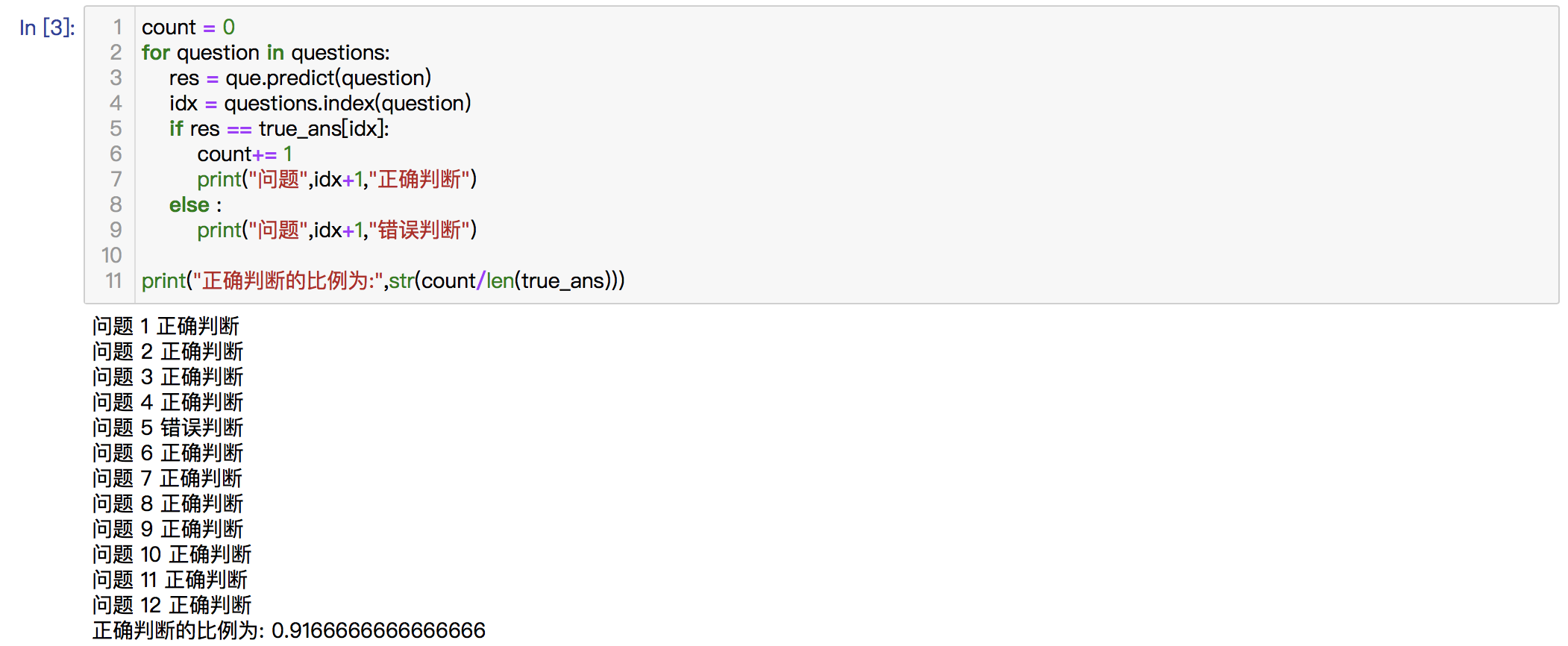
将查询后的结果加上一些前缀或者后缀修饰语，即可组成最终的答案。比如查询某位老师教的课，加上“xx老师所教的课有：”作为答案的前缀修饰。

**4 实验**

**4.1 问题分类器实验效果**

在**3**中我们介绍了训练问题分类器中的训练数据集构建，构建完成后，我们来测试该分类器的预测效果。





可以看到，使用朴素贝叶斯分类器的模型拟合不错，基本可以学习到每个问题中的语言模型，对用户的提问进行正确的判断。对于问题4、5，即某院系开设的课程、某院系开设的某类课程，该模型不太能准确分辨。我想这是由于目前还没有用词性标注替换原问题来预测导致的，下图展示的是用词性标注替换后的预测结果，两个都预测正确。



由于问题类别数目少，一共十类；而且每一类间词频差异较明显，因此使用朴素贝叶斯分类器可以训练出不错的效果，对于用户的提问能够正确的判断。

**4.2 针对课程评价的情感分析**

**4.2.1 课程评价数据获取**

复旦课程评价的获取方法有日月光华BBS论坛，评教网站和“复小七”选课公众号。由于BBS的评价过于分散，评教网站上展示的文本评价结果非常少，我最终决定爬取公众号“复小七”的推送，建立课程评价数据集。该公众号推出七大模块及美育课程的评价**[[3]](#footnote-3)**，我考虑对其中的界面进行解析，提取文本评价。使用Beautiful Soup 和Re模块进行页面的解析，即可获得所有的评价数据。



图6. 公众号“复小七”的课程评价示例

一共获取三百多条评价数据，但是其中包含一些噪音数据，训练的模型容易发生过拟合；或者情感取向比较模糊，难以判断其归类问题。我通过人工标注数据集中的文本情感（1代表积极，0代表消极），最终筛选出**100**条有效、合理的评价文本，部分实例如下图：

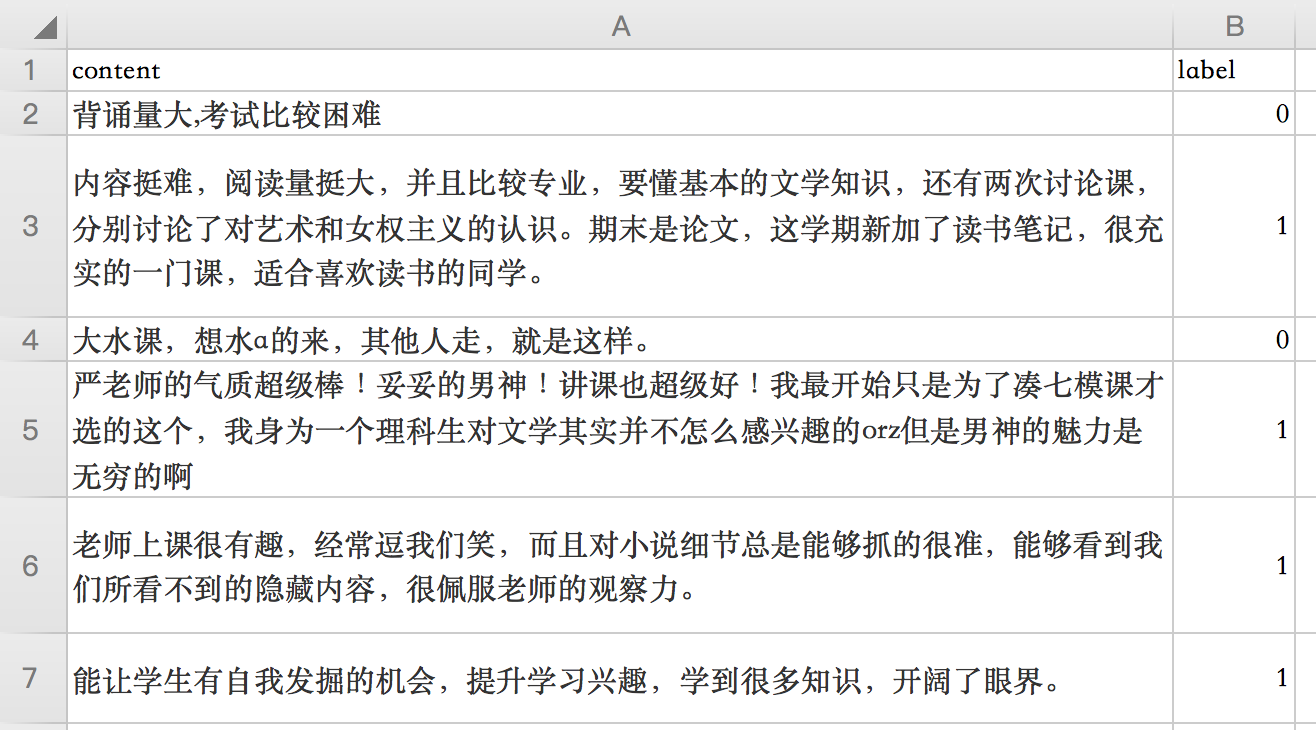


图7. 课程评价部分数据

**4.2.2 课程评价的文本分析**

在**100**条评价文本中，正样本**75**个，负样本**25**个。我分别对正负样本进行词云分析，结果如下：



图8. 课程评价的词云，左图为负样本，右图为正样本

可以看到同学们评价较高的课程，主要是给分好、有趣、教师授课内容专业。而评价一般的课程，主要特点给分不好、事情多、无聊等。

**4.2.3 情感分类器训练**

在本系统中，使用到情感分析模型是在对课程进行筛选，筛选出评价不错的课程。因此，使用到的预训练模型计算速度要快。考虑到深度学习模型，如基于RNN的模型，本身是时序模型，不能并行计算，速度略慢。我这里依然采用传统的机器学习方法进行情感分类，牺牲了一定的准确度，但是模型的计算速度非常快。

我使用**SnowNLP**进行情感分类，具体代码如下：



由于SnowNLP包中内置了情感分析工具，于是对同一句话“**水课，学不到什么内容**”,我分别运行了原包中内置的情感分析工具和通过评价文本训练后的情感分析模型。可以看到内置的模型中，对这句话的情感得分达到了0.54；但实际上这句话是对课程较为消极的评价，我们训练好的模型中这句话的情感得分0.28，更偏向消极情绪，因此更为准确。



图9是训练好的模型对课程评价的运行结果（这几个例子都是我随便写的，并不在训练集中）。可以看到，对于正面评价，情感得分接近**1**；对于负面评价，得分接近于**0**。这说明我们训练的情感分析模型较为有效。



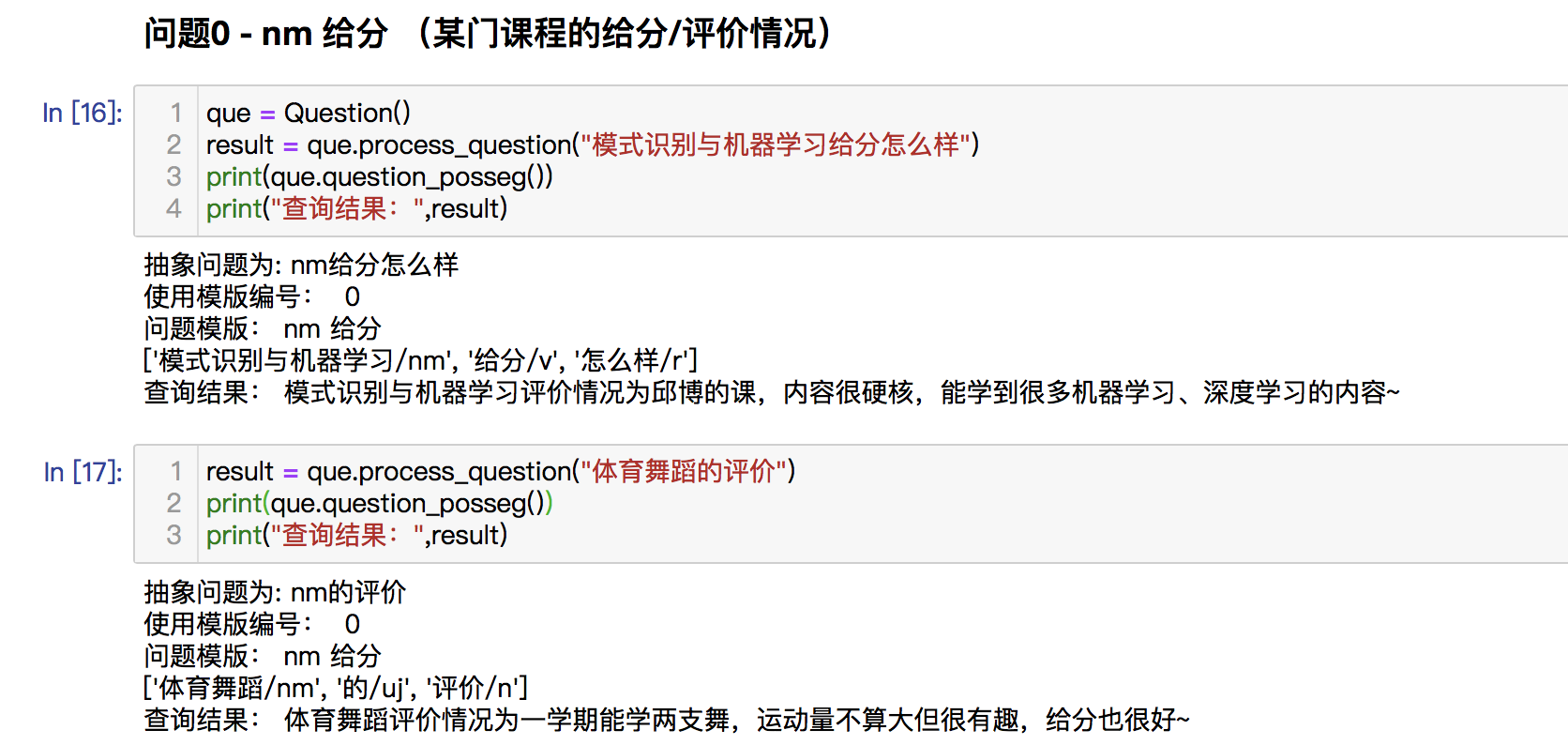
图9. 训练的情感分析模型运行结果

**4.3 系统运行效果**

前面已对系统的各个部件进行了说明，将各部分进行组装（词性标注、问题分类、实体抽取、问题匹配、答案查询与返回）后，系统便可以正确工作。

下面分别展示本系统运行各类问题的结果。

**4.3.1 运行各类问题效果示例**

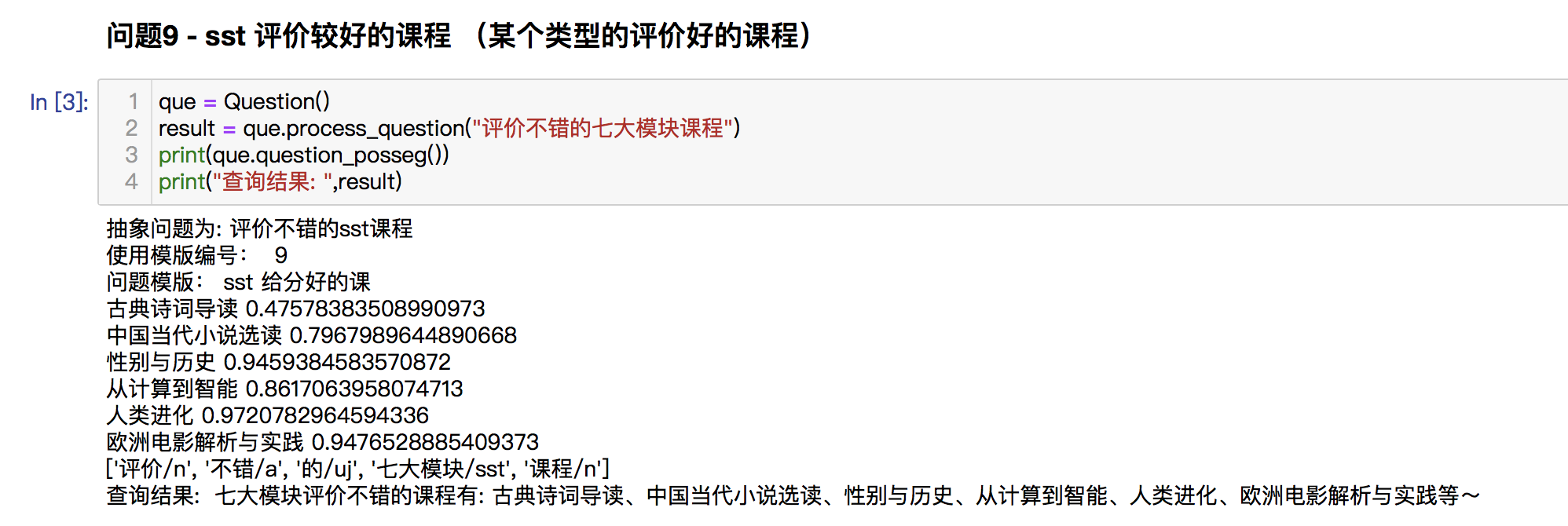












(注：在进行“评价较好”的课程筛选时，我将阈值设定为0.4)

可以看到，对于这十大类问题，本系统基本上可以正确识别、从图数据库中调出答案并返回。其中问题8、9展示的课程名称后的数字为调用文本情感分析后的模型对该门课课程评价的情感分析得分，我以0.4作为“评价不错”的阈值。

**4.3.2 聊天式界面搭建**

由于本项目是一个问答系统，如果真正给用户使用，需要一个聊天式的界面。在本项目做完后我还有几天的空闲时间，因此就做了一个前端的界面。考虑到移动端使用相比较网页更为便捷，我搭建了移动端的界面。

移动端使用Flutter**[[4]](#footnote-4)**建立。Flutter是谷歌公司最近几年推出的移动端框架，最大的优点是可以跨iOS和Android平台。具体的搭建思路与课程关系并不大，这里不再赘述，仅展示最终的效果。

移动端界面如图10所示，分别是iPhoneXR和Android手机的模拟机效果图。

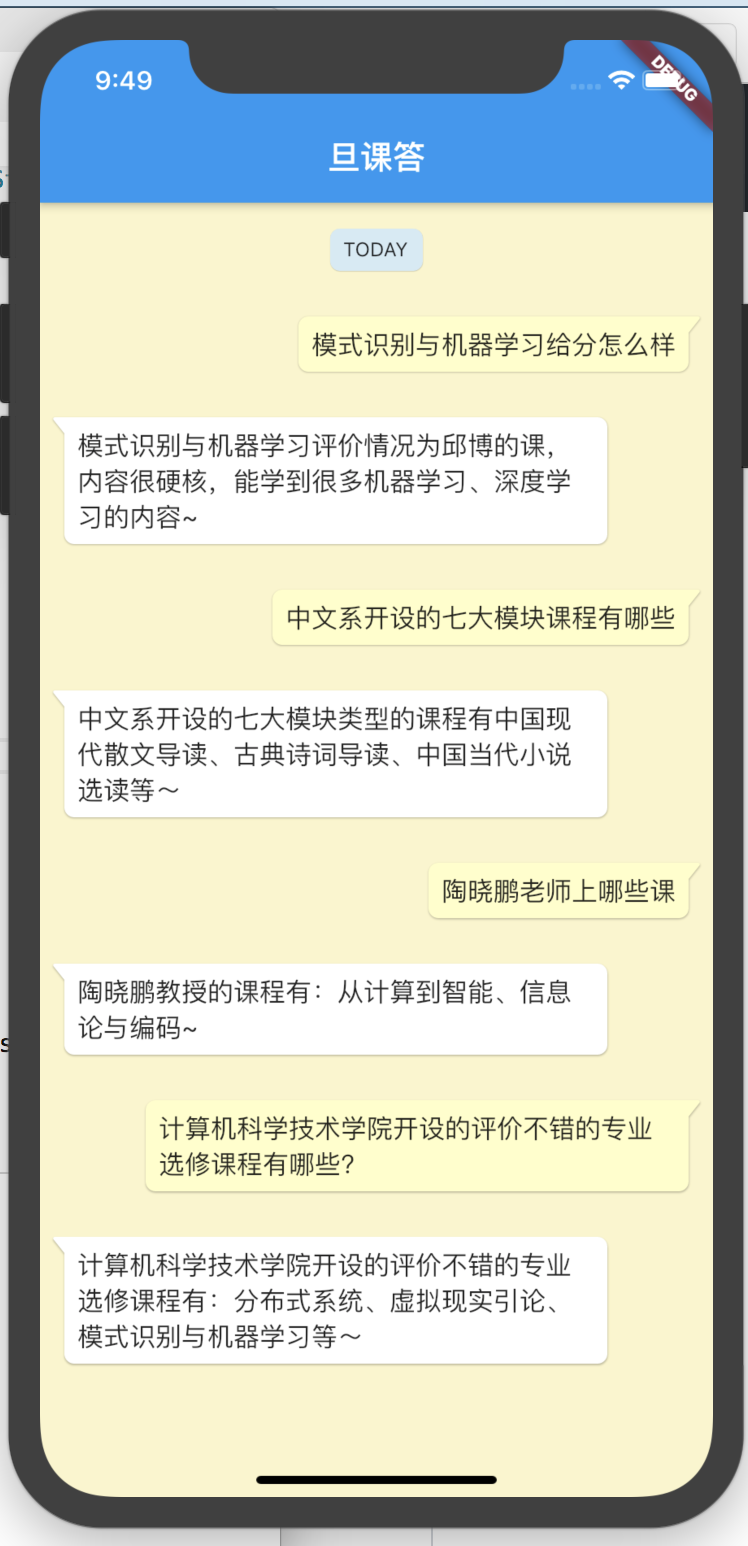
 

图10. 本系统问答移动端界面，左侧为iPhone模拟机，右侧为安卓模拟机

**5 总结**

一开始做这个项目只是完成课程的要求，但是当我一步步推进、尤其是可以做成移动端的App应用后，觉得这个项目真的可以开发下去，而且系统运行的结果正确流畅，能够给同学们选课提供一定的帮助和指导。

这个项目的完成让我对问答系统的建立有了一定的认识，从分词到问题匹配、实体抽取和图数据库查询，每个方面的知识和技术都有了一定的掌握。本系统使用的两个分类模型，基于朴素贝叶斯的问题分类和SnowNLP的文本情感分类，也让我对文本分类任务的处理过程和方法有了更深的了解。

当然，本项目也存在着缺陷。一个是现有的数据规模较小，只涵盖了复旦课程的子集；二是提供的问题不够全面；另外，如果真正给用户使用，训练的识别问题的分类器和情感分析模型的准确度都需要再提高，这需要更充分的测试。此外，本系统的容错能力也不够，比如只有输入完整的“计算机科学技术学院”才能识别，如果出现“计算机学院”、“计院”或者有错别字，都不能返回正确的答案。

**6 附录**

**6.1 项目文件列表**



由于elearning上提交的文件大小和数目有限制，我仅上传了系统的核心文件，即Course\_QA文件夹下的几个python文件。整个项目我上传在我的Github仓库**[[5]](#footnote-5)**中。

**6.2 系统运行方法**

此系统需要的依赖包括：

* Python3
* Jieba
* Sklearn
* Snownlp
* Py2neo

此外，需要下载图数据库Neo4j桌面端，并将本系统的数据（/data子目录下所有.csv文件）导入图数据库中。

环境部署完成后，需要先开启图数据库，之后训练文本情感分类模型（run -> /review/sentimentTrain.py）.最后，run -> questionProcess.py 即可看到问题的运行情况。

最后，说明一下Github该项目上的作者是我：

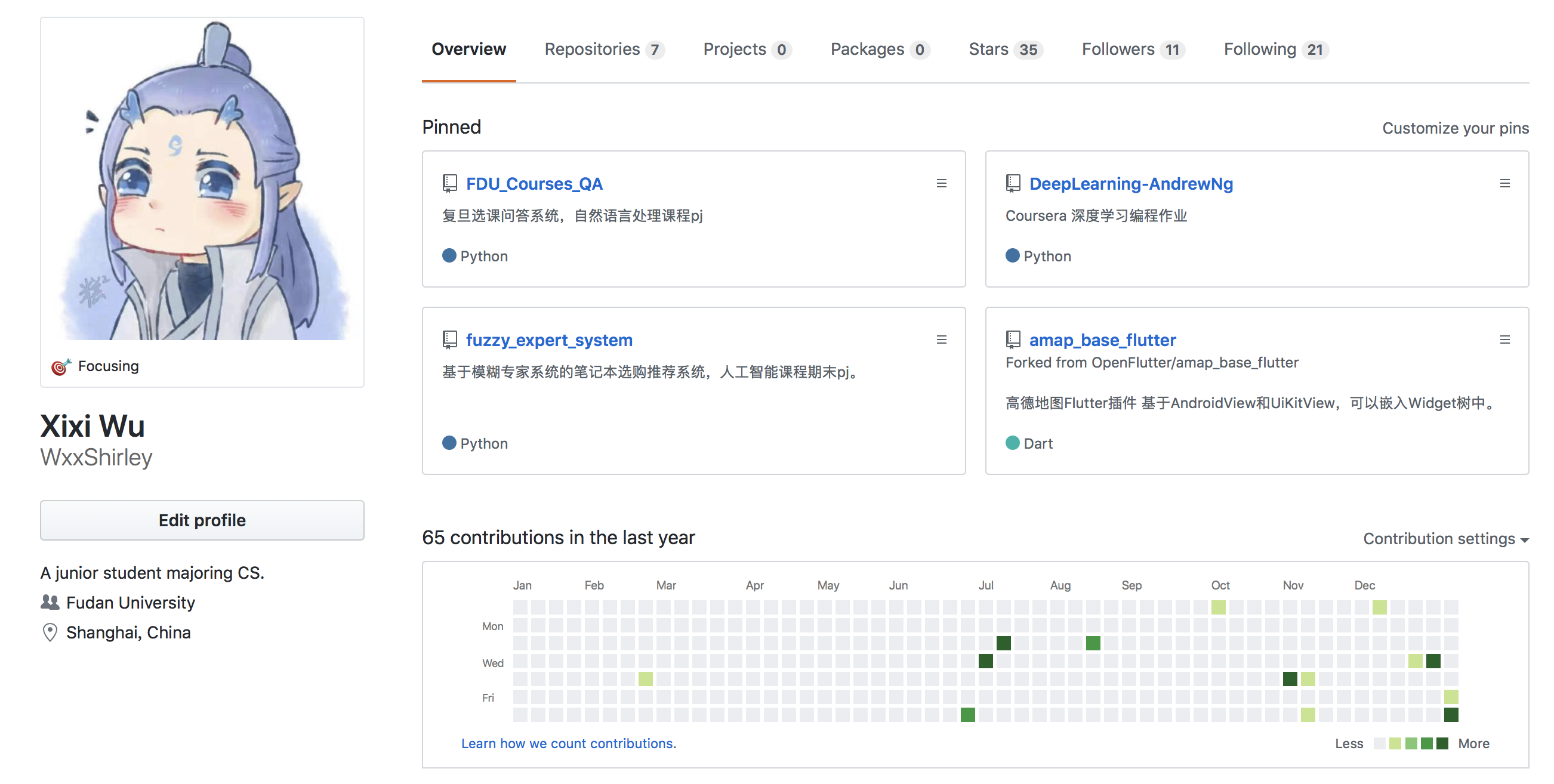


图11. 我的Github账号

1. [http://www.jwc.fudan.edu.cn](http://www.jwc.fudan.edu.cn/) [↑](#footnote-ref-1)
2. [http://ce.fudan.edu.cn](http://ce.fudan.edu.cn/) [↑](#footnote-ref-2)
3. <https://mp.weixin.qq.com/s/gLVSHTU9Cu2miAUCFqbdkA> [↑](#footnote-ref-3)
4. [https://flutter.dev](https://flutter.dev/) [↑](#footnote-ref-4)
5. <https://github.com/WxxShirley/FDU_Courses_QA> [↑](#footnote-ref-5)