

任务五 校园客服

【任务描述】

在智能客服的业务场景中，对于用户频繁问到的业务知识类问题的自动解答（以下简称为FAQ）是一个非常关键的需求，可以说是智能客服最为核心的用户场景，可以最为显著地降低人工客服的数量与成本。

本任务中校园客服以自然语言处理为主，自然语言处理在校园客服中的作用是对输入的问题进行分析，在问题库中寻找出与输入语句最相似的问题集合，然后根据排序算法，在相似问题集合中抽取出一个问题对应的答案。寻找相似问题是需要通过构建相似度模型和排序模型。这里，我们使用 YunGe API 编写一个校园客服机器人。

【任务实施】

步骤1 导入以下依赖库。

```
import requests
import json
```

步骤2 定义一个函数upload_data，调用Yunge API，上传question数据。

```
def upload_data(question):
    url = 'https://testapi.smartyg.com/api/smart_qa'
    keys = '5f15a18f3f03f7e88020acb1c2f8c93c'
    post_json = {
        'keys':keys,
        'question':question,
        'command':'upload_data'
    }

    result = requests.post(url,json.dumps(post_json)).text
    data = json.loads(result)
    if data['flag'] == 'success':
        print('上传数据成功')
```

requests中的参数含义如下：

参数	含义
keys	请求的keys
question	传输的语句
command	指定

步骤3 定义一个函数xiaoming，调用Yunge API，发送text，获得回复。

```
def xiaoming(text):
    res = ''
    url = "https://testapi.smartyg.com/api/post_gossip"
    keys = "5f15a18f3f03f7e88020acb1c2f8c93c"
    post_json = {
        'keys':keys,
        'question':text
    }

    result = requests.post(url, json.dumps(post_json)).text
    data = json.loads(result)
    if data["flag"] == "success":
        res = data["answay"]
    return res
```

requests中的参数含义如下：

参数	含义
keys	请求的keys
question	传输的语句

步骤4 定义question_dict，创建问答对，然后调用upload_data上传数据

```
question_dict = {
    '饭堂一餐多少钱': '一餐开销约10~15元',
    '学校饭堂在哪': '饭堂位置：西区西苑餐厅(西区宿舍门口), 东区一湖接（彩短桥附近）',
    '哪里的宵夜好吃': '宵夜推荐东区1楼，有炒粉炒饭，生滚粥等应有尽有',
    '学校的门禁时间': '周日~周四门禁时间：23:00，周五~周六门禁时间：00:00',
    '图书馆开放时间': '图书馆开放时间：8:30~22:00'
}
upload_data(question_dict)
```

步骤5 调用xiaoming，进行问答。

```
while True:
    question = input('我:')
    answay = xiaoming(question)
    print('客服: ',answay)
```

完整参考代码如下：

```
import requests
import json

def upload_data(question):
    url = 'https://testapi.smartyg.com/api/smart_qa'
    keys = '5f15a18f3f03f7e88020acb1c2f8c93c'
    post_json = {
        'keys':keys,
        'question':question,
        'command':'upload_data'
    }

    result = requests.post(url,json.dumps(post_json)).text
    data = json.loads(result)
```

```

if data['flag'] == 'success':
    print('上传数据成功')

def xiaoming(text):
    res = ''
    url = "https://testapi.smartyg.com/api/post_gossip"
    keys = "5f15a18f3f03f7e88020acb1c2f8c93c"
    post_json = {
        'keys':keys,
        'question':text
    }

    result = requests.post(url, json.dumps(post_json)).text
    data = json.loads(result)
    if data["flag"] == "success":
        res = data["answay"]
    return res

question_dict = {
    '饭堂一餐多少钱': "一餐开销约10~15元",
    '学校饭堂在哪': '饭堂位置：西区西苑餐厅(西区宿舍门口), 东区一湖接（彩短桥附近）',
    '哪里的宵夜好吃': '宵夜推荐东区1楼，有炒粉炒饭，生滚粥等应有尽有',
    '学校的门禁时间': '周日~周四门禁时间：23:00，周五~周六门禁时间：00:00',
    '图书馆开放时间': '图书馆开放时间：8:30~22:00',
}

upload_data(question_dict)

while True:
    question = input('我:')
    answay = xiaoming(question)
    print('客服: ',answay)

```

本任务实战代码如下,位于/xml1/rw5.ipynb 同学们来运行一下吧

【任务拓展】

将聊天机器人应用到机器人或语音套件。

让机器能听、能看、能说、能感觉，是未来人机交互的发展方向，其中语音成为未来最被看好的人机交互方式，语音比其他的交互方式有更多的优势。当前，语音对话机器人、语音助手、互动工具等层出不穷，许多互联网公司纷纷投入人力、物力和财力展开此方面的研究和应用，目的是通过语音交互的新颖和便利模式迅速占领客户群。



思政聚焦

