

面向情绪感知的多功能对话系统的设计与实现

王明键, 梁浩纯, 高大拯, 冯 时

(东北大学计算机科学与工程学院, 辽宁 沈阳 110819)

摘 要:现如今, 自然语言处理技术不仅可以实现人机对话, 还能从对话中提取出用户所表达的情感。本文中所介绍的对话系统将以上两项技术结合起来, 使用Web开发相关技术构建了一个功能较为完备的多功能对话系统, 不仅具有查询天气、新闻、一对一闲聊等基础功能, 还可以将用户的实时情绪统计展示出来, 进而分析用户情绪变化趋势生成图表。本文对此系统整体设计和实现过程中用到的HTML、Django和数据库等相关技术进行了详细说明。

关键词: 对话系统; 情感分析; Django

中图分类号: TP311.1 **文献标识码:** A

Design and Implementation of Emotion-Recognition Multi-Functional Dialogue System

WANG Mingjian, LIANG Haochun, GAO Dazheng, FENG Shi

(School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China)

Abstract: Nowadays, natural language processing technology can not only realize man-machine dialogue but also analyze the emotions expressed in the dialogue process. The dialogue system introduced in this paper combines these two technologies together, building a multi-functional dialogue system with not only the basic functions of weather query, news and one-on-one chat, but also the function of displaying the user's real-time emotion statistics to analyze the user's emotional trend and to generate a chart. This paper elaborates on the related technologies such as HTML, Django and database used in the overall design and implementation of the system.

Keywords: dialogue system; sentiment analysis; Django

1 引言(Introduction)

聊天机器人的研究开始与1950年, 英国知名数学家图灵在《Mind》上发表了一篇经典论文——《计算机与智能》^[1]。在文中, 图灵第一次提出“机器智能”的概念。如今, 在深度神经网络技术的推动下, 自然语言处理学科日益完善, 计算机对人类使用自然语言描述的需求识别地越来越准确, 对简单问题和命令可以十分准确地做出回应^[2-5]。因此催生出了许多“语音助手”程序, 可以实现诸如查询天气、新闻等功能, 极大程度方便了人们的生活。然而现有的“语音助手”工具仅仅能够回应单一的指令, 无法捕捉和应用语句中的情感信息^[6]。

本文中实现的对话系统对现有“语音助手”进行了功能上的扩展, 不仅可以实现查天气等基本功能, 还可以分析用

户输入语句的情感, 将其以emoji表情的形式直观显示在对话界面上, 并且使用带有情感指定功能的聊天机器人后端生成契合当前用户情感的回复语句。用户还可以随时查询自己在一段时间内的情感变化情况, 以及用户总体的情感偏好。

2 需求分析(Requirements analysis)

本系统所要实现的目标为一个多功能对话系统, 拥有查天气、查新闻、讲笑话和闲聊四项基本功能。在用户与本系统交互时, 系统可以实时分析出用户每句话所蕴含的情感倾向信息, 在做出回复的同时将情感倾向以emoji表情的方式显示在每条消息上, 并统计用户感情变化趋势生成折线图。用户可选择匿名或者登录账号使用本系统。用户登录后可获得修改头像和保存聊天记录的功能。登录信息保存在Cookie和Session中, 以实现记住登录状态避免重复登录。

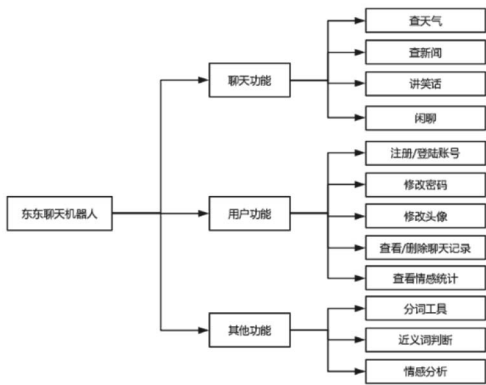


图1 系统功能模块图

Fig.1 System function model

3 系统设计和实现(System design and implementation)

3.1 总体设计

本系统基于Web技术开发，运行在网页浏览器上。前端采用Bootstrap和JQuery工具，以HTML和JavaScript语言编写，后端采用Django框架，以Python语言编写。前后端之间通过Ajax技术异步通信，数据全部以JSON格式包装。用户在网页上产生请求后，后端通过http协议将请求内容转发到相应的功能模块服务器上，各个模块产生的回应再通过网页后端服务器返回给用户。通过这样的设计，将系统本身与各个功能模块分离，结构清晰易于扩展，如图2所示。

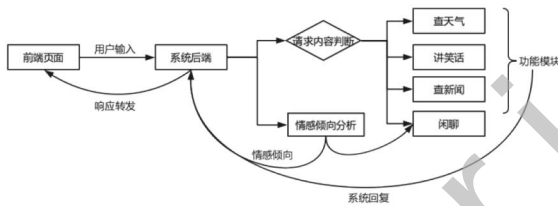


图2 系统整体架构图

Fig.2 System overall framework

3.2 数据库设计

经过分析，需要保存到数据库的仅有用户信息和聊天记录两项。使用sqlite作为数据库引擎，其结构通过Django的Model模型功能定义。由于Django自带了User数据表，其中包含用户名、密码等基本信息，因此不再进行重复设计，数据库结构如图3所示。

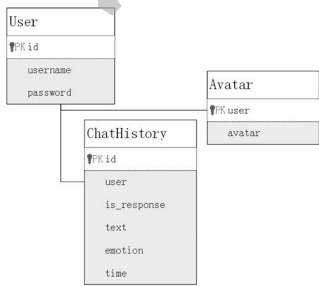


图3 数据库模型图

Fig.3 Database model

为了保存用户头像，新建了一个Avatar数据表，其中的user字段一对一关联到User表中的一个项目，相当于对User

表进行了扩展。上传的头像文件会自动以用户名重命名，保存到/media/avatar/中。Avatar表中avatar字段为Text类型，为头像文件的路径。该数据表使用了Django提供的信号功能添加了一个回调函数以实现当用户上传新头像时自动删除旧头像文件，释放存储空间。

ChatHistory表用于保存聊天记录，其中字段如下：user为外键，多对一关联到User表中的一个项目；is_response字段为Boolean型，用于区分该条记录是否为系统做出的回复；text字段为Text类型，保存聊天记录；emotion为Float类型，记录该条记录的情感倾向；time字段为该条记录产生的时间，设置其auto_now属性为True，在进行数据插入时该字段会自动被设置为当前时间。

3.3 后端设计

本系统后端使用Django框架制作。Django是一个基于Python编程语言的MVT Web应用框架。其提供了一个基于正则表达式的URL分发器，可以在特定URL被请求时自动执行相应的Python函数。本系统所有功能全部通过http请求的形式调用。可请求URL如表1所示。

表1 可请求URL表

Tab.1 URL request

URL	方法	功能
/	get	首页
login	get	登录页面
user	get	用户中心页
signup	post	登录功能
signin	post	注册功能
logout	post	退出登录
changepsw	post	修改密码
uploadavatar	post	上传头像
deleتهistory	post	删除历史记录
chat	post	聊天功能
separation	post	分词功能
sentiment	post	情感判断
synonym	post	近义词判断
pc-geetest/get	get	获取验证码
statistic	get	获取情感统计

(1)注册/登录/修改密码功能

注册与登录功能共用一个页面，修改密码功能在用户中心二级页面中。此三项功能的核心部分均为输入表单和验证码。当页面加载完成时，客户端会自动请求pc-geetest/get获取验证码，用户在表单中填写必要信息后点击验证码进行身份验证，此时会弹出滑动验证窗口，若用户成功通过验证，则在表单中自动添加challenge、validate和seccode三个隐藏的input元素。点击确定按钮时，以上三个元素与用户填写的信息被一同发往服务器。若表单中此三元素不存在或此三元素的值通过特定算法计算失败，则认为用户没有通过验证码测试，直接返回错误信息。验证通过后，对数据库进行增删改查，实现注册/登录/修改密码功能。

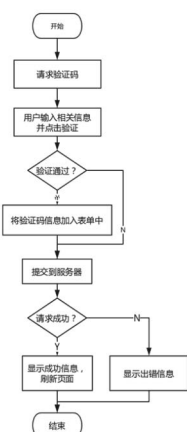


图4 注册、登录、修改密码流程图

Fig.4 Flow chart of sign-up,login and password modification

登录、注册、修改密码等需要传递敏感信息的场合，为防止跨站请求攻击，在生成的表单内部含有一个隐藏的input项目，其name为csrfmiddlewaretoken，value为服务器自动生成的验证代码，在提交时必须将csrf token与表单信息同时提交到服务器，否则服务器会拒绝此次请求。

用户登录后，系统自动将登录信息写入到Cookie和Session中。Cookie可以让浏览器记住登录状态。用户在进行请求时系统后端可以直接从Session中取得登录信息，从而对登录用户和匿名用户做出不同的响应。

(2)聊天功能

本系统后端与各种功能模块完全分离，由于功能模块可能分布在不同的服务器上，因此后端通过http请求的方式调用各种功能模块。当后端接收到用户在聊天界面输入的内容后，首先计算该段文字的情感倾向，再先检查其是否包含“天气”“新闻”或“笑话”字样，若包含，则调用相应的API进行处理。若不包含，则将对话内容和情感倾向值转发至聊天机器人后端，最后将各个模块的回复和情感值一同作为http响应返回给前端。

算法1 聊天功能

输入：用户输入text

输出：系统恢复response，输入句子的情感sentiment

if text is empty:

return

```
sentiment=calculate_sentiment(text) //计算情感倾向
```

if text contains ‘天气’:

```
response=weather(text) //查询天气需要从text中
```

提取城市

else if text contains ‘新闻’:

```
response=news() //调用查询新闻API
```

else if text contains ‘笑话’:

```
response=joke() //调用笑话API
```

else:

```
response=chat(text,sentiment) //闲聊功能，需要
```

指定输入内容和情感倾向。

```
return response,sentiment
```

在查询天气指令中,需要在对话内容中提取出要查询的城市信息,方法为对内容字符串正向扫描,首个与已知城市最大匹配的字串即为要查询的城市名。已经预先将已知城市列表构建成前缀树,用于加速匹配过程。在前缀树中,所有节点拥有final属性表示到该节点是否为一个完整的城市。树的节点与子节点通过字符串中的下一个字符进行状态转移。

算法2 前缀树匹配算法

输入：树根节点root，待匹配字符串str

c=root //c为在树上搜索的指针

```
for w in str:
```

```
if w in c: //子树包含待匹配串的下一个字符, 状态
```

$$c=c[w]$$

else:

break

```
return c['final']
```

当前节点是否为可匹配的完整字符串

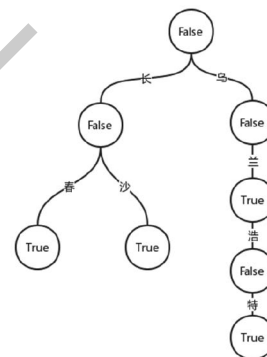


图5 匹配长春、长沙、乌兰、乌兰浩特的前缀树

Fig.5 Prefix tree for Changchun,Changsha.

Ulan and Ulanhot

若用户输入的内容带有html标签, 如<script>alert(1234);</script>, 若直接将其添加到页面上, 浏览器会运行其中代码, 攻击者可以利用此特性执行恶意代码, 造成潜在的安全漏洞。因此当用户进行输入时, 本系统使用JavaScript自动将输入内容中的符号转换对应的html实体, 如<script>将会被转换成<script>可以直接无歧义地显示在界面上, 消除了安全隐患^[7]。

(3)情感倾向分析

为了提取出句子中所蕴含的情感信息, 本系统基于长短程循环神经网络(LSTM)设计了一个情感分类器^[8]。其核心为两层拥有32个参数的LSTM神经元, 后再接两个32 16和16 1的全连接层。使用时将输入句子内容分词过后, 再利用word2vec^[9]把句子中的每个词转换为300维的向量表示。将向量序列放入本模型进行计算, 输出即为该句子的情感倾向值。本模型采用Chinese conversation sentiment^[10]提供的中文情感分类数据进行训练。经过训练后, 本模型在测试集上

的准确率达到91.3%，召回率为84%。

3.4 前端设计

Django作为MVT结构的Web应用框架，为开发者提供了一套功能完备的模板系统，本项目的前端页面中，用户头像、历史记录等信息均通过模板在服务端一次渲染而成，而不是在页面加载完成后再通过Ajax二次请求数据，减少了客户页面与服务器通信的次数。

本系统主要拥有主页、登录和注册、用户中心三个页面。三个页面遵循同样的设计风格，并且利用Django的模板继承功能将页面的公共部分，如header、加载js和css文件等代码提取出来作为父模板，避免出现重复代码，三个页面均从父模板继承而来。主页页面的核心部分为一个聊天窗口，用户与系统的对话以聊天气泡的形式展现。按照积极程度将对话分为四类，使用emoji表情按积极程度递减依次表示为：😊、😄、😁和😂。情绪标志、对话文字，以及用户头像缩略图包装在一个div中，通过设置css中的float参数实现聊天气泡居左或居右显示^[11]。



图6 聊天气泡

Fig.6 Chat bubble

当对话数量较多后，聊天对话框被拉长，影响使用体验，因此为聊天对话框设置最大高度后添加overflow-y:scroll;样式，可在内容超出最大高度后自动显示滚动条。在用户输入内容或系统对用户进行回复后，通过对聊天框的scrollTop参数进行设置实现滚动条自动滚动显示最新内容。聊天对话框主体下方为输入框和发送按钮，当页面加载完成时，为发送按钮设置键盘监听事件，以实现按回车键直接发送消息。

用户中心页面提供了修改密码、更换头像、查看情感统计和删除聊天记录功能。修改密码功能与注册登录完全相同，由表单、验证码、csrf token组成。更换头像功能需要用户上传一个图片文件，本系统使用bootstrap的file input插件实现该功能，指定上传的图片文件类型只能为jpg、png和gif中的一种，文件最大为1MB。由于上传图片也使用POST方式传输，因此必须在http数据报中手动加入csrf token信息，否则会被服务器拒绝服务^[7]。

当用户点击进入情感分析页面后，页面通过ajax异步向服务器请求该用户的历史情感信息。接收到信息后，本系统首先会把情感记录按日期分类，然后将每一天用户的情感随时间的变化情况绘制成折线图，方便用户查看情感变化趋势。另外，本系统还会将用户全部历史对话记录中属于四种情感分类的语句统计出来，以饼图的形式展示每种情感分类的比例，可以直观地看到该用户总体的情感偏好。

4 系统部署与测试(System deployment and testing)

Django应用程序可以使用uwsgi作为http后端，但是uwsgi功能较为简单，仅适用于开发测试之用。在生产环境中，选择高性能的nginx作为整个系统的前端服务，在nginx

中设置反向代理，将/路径映射到Django的uwsgi服务后端，并将static/和media/映射到专用的静态文件和媒体文件目录，使用Django提供的collectstatic命令将工程内的静态文件转移到专用目录中。如有需要，也可将以上两目录转移到专用的文件服务器上，以减轻主服务器访问压力^[12]。

系统运行结果如图7—图10所示。



图7 系统聊天界面

Fig.7 Dialogue system user interface



图8 用户中心页面

Fig.8 User center



图9 注册/登录页面

Fig.9 Sign up/login

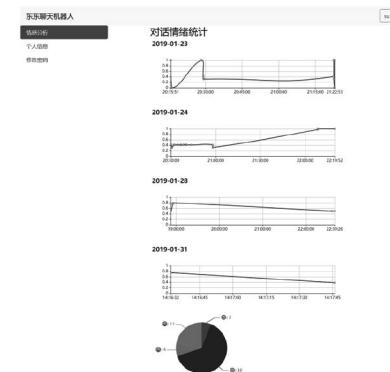


图10 情绪状态跟踪页面

Fig.10 Emotion state tracking

5 结论(Conclusion)

本文介绍了一个具有情感分析的多功能对话系统的实现过程。本系统基于Django框架，使用Python语言编写，代码较为轻量化，后端各个模块之间结构清晰，耦合度低，易于

修改和扩展。本系统在实现时也考虑到了数据安全的问题，使用了在敏感操作前加验证码、CSRF Token验证和HTML转义的机制，分别阻止了暴力破解密码，跨站请求攻击和XSS注入攻击。本对话系统可为用户提供天气、新闻、笑话和闲聊等日常服务，同时本系统能够感知用户的情绪状态，将其记录下来并以折线图和饼形图的方式展现给用户，方便用户了解自己的情感变化趋势和总体情感倾向。此外，本系统还提供了中文分词、情感判断、近义词判断三项工具服务，具有一定的实用价值。

参考文献(Reference)

- [1] Turing A M.Computing machinery and intelligence[M].Parsing the Turing Test.Springer,Dordrecht,2009:23-65.
- [2] Sutskever I,Vinyals O,Le Q V.Sequence to sequence learning with neural networks[C].Advances in neural information processing systems,2014:3104-3112.
- [3] Zhou H,Huang M,Zhang T,et al.Emotional chatting machine: Emotional conversation generation with internal and external memory[C].Thirty-Second AAAI Conference on Artificial Intelligence,2018.
- [4] 王浩畅,李斌.聊天机器人系统研究进展[J].计算机应用与软件,2018,35(12):1-6;89.
- [5] 高程.聊天机器人“小黄鸡”的语用原则和话语策略分析[J].语文学刊,2013(10):30-31;42.
- [6] Shawar B A,Atwell E.Different measurements metrics to evaluate a chatbot system[C].Proceedings of the workshop on bridging the gap: Academic and industrial research in dialog technologies. Association for Computational Linguistics,2007:89-96.
- [7] 韩可或.基于Django的XSS和CSRF防御系统的研究与实现[D].武汉邮电科学研究院,2018.
- [8] 五行素,陈锦回.基于LSTM深度神经网络的情感分析方法[J].上饶师范学院学报,2018,38(06):10-14.
- [9] Le Q,Mikolov T.Distributed representations of sentences and documents[C].International Conference on Machine Learning,2014:1188-1196.
- [10] Zhang L,Chen C.Sentiment classification with convolutional neural networks:an experimental study on a large-scale Chinese conversation corpus[C].Computational Intelligence and Security (CIS),2016 12th International Conference on.IEEE,2016:165-169.
- [11] 黄志刚.深入理解CSS3结构伪类选择器[J].软件工程,2018,21(3):34-36.
- [12] 苏翔宇,朱爱群.CentOS7下基于Nginx的反向代理及负载均衡研究与实现[J].现代计算机(专业版),2018(10):61-64.

作者简介:

王明键(1997-),男,本科生.研究领域:软件工程,机器学习.

梁浩纯(1998-),男,本科生.研究领域:自然语言处理.

高大拯(1997-),男,本科生.研究领域:深度学习,软件工程.

冯时(1981-),男,博士,副教授.研究领域:数据挖掘,自然语言处理.

(上接第46页)

参考文献(References)

- [1] Papagorgiou G.Bioenergetics of Photosynthesis[M].New York:Academic Press,1975:319-337.
- [2] K.Roháček.Chlorophyll Fluorescence Parameters:The Definitions,Photosynthetic Meaning,and Mutual Relationships[J].Photosynthetica,2002,40(1):13-29.
- [3] G H Krause,E Weis.Chlorophyll Fluorescence and Photosynthesis:The Basics[J].Annual Review of Plant Physiology and Plant Molecular Biology,1991,42(1):313-349.
- [4] 王子毅,张春海.基于ECharts的数据可视化分析组件设计实现[J].微型机与应用,2016,35(14):46-48;51.
- [5] 温国胜,田海涛,张明如,等.叶绿素荧光分析技术在林木培育中的应用[J].应用生态学报,2006,17(10):1973-1977.
- [6] 王莉敏,梁正和,段全锋.基于HTML5大文件断点续传的实现方案[J].计算机与现代化,2016(03):91-95.
- [7] 林立云.推荐一种读和写CSV文件的程序设计方法[C].Proceedings of 2011 National Teaching Seminar on Cryptography and Information Security(NTS-CIS 2011),2011,1(3):93-95.
- [8] 万文博,华灯鑫,乐静,等.基于激光诱导叶绿素荧光寿命成像技术的植物荧光特性研究[J].物理学报,2015(19):68-74.

作者简介:

程盛阳(1996-),男,本科生.研究领域:物联网技术.

郭亚(1977-),男,博士,教授.研究领域:物联网技术.

夏倩(1994-),女,硕士生.研究领域:物联网技术.