Design and Implementation of a Music Recommendation Chatbot for Lonely People

Xianglin ZHAO

M.Sc. Practicum Report

Department of Computer Science

Hong Kong Baptist University

2021

Design and Implementation of a Music Recommendation Chatbot for Lonely People

Xianglin ZHAO

An M.Sc. practicum report submitted in partial fulfillment
of the requirements for the degree of
Master of Science

Principal Supervisor: Dr. Li CHEN

Department of Computer Science

Hong Kong Baptist University

May 2021

Acceptance of M.Sc. Practicum Report

Master of Science in Advanced Information Systems

Department of Computer Science

Hong Kong Baptist University

We hereby recommend that the M.Sc. Practicum Report submitted by Xianglin ZHAO entitled "Design and Implementation of a Music Recommendation Chatbot for Lonely People" be accepted in partial fulfillment of the requirements for the degree of Master of Science in Advanced Information Systems.

_____	_____
Dr.Li CHEN	Dr. Jacky CHAN
Supervisor	Co-supervisor
Date: _____	Date: _____

Declaration

I hereby declare that this M.Sc. Practicum Report represents my own work which has been done after registration for the degree of M.Sc. at the Hong Kong Baptist University, and has not been previosuly included in a project report, thesis or dissertation submitted to this or other institution for a degree, diploma or other qualification.

_____
Xianglin ZHAO

Date: _____

Abstract

This project uses the Rasa framework to build a chatbot, which can detect the user's emotions and recommend suitable music to help the user ease his emotions. Since there is no labeled data, I collected the comments of NetEase Cloud Music, and inferred the emotion of the music through the comments. I used the LSTM neural network to build an emotion classification model to recognize the user's emotions and recommend music based on two recommendation rules. Finally, this chatbot can talk to people smoothly, collect information from users, detect emotions, and recommend music.

Keywords: Chatbot，Emotion Detection，Music Recommendation

Acknowledgments

I would like to express my most sincere gratitude to my supervisor, Dr. li CHEN. The idea of this project was put forward by her. Due to my limited experience, she gave me a lot of professional guidance and patient help. I learned a lot through this project.

Dr. li CHEN invited professors Jiang Tonglin and Dr. Dai Yuwan of the Department of Psychology of Peking University to introduce the theories that can relieve loneliness through nostalgia, thank them for providing professional psychological knowledge and guidance for this project.

TABLE OF CONTENTS

Page

1. Introduction

1.1. Background

Since the beginning of 2020, the COVID-19 epidemic has lasted for nearly one and a half years. In order to prevent the spread of the epidemic, many places have taken measures to limit social distance, reduce people's daily contact, and even require people to stay at home and self-isolate. Studies have shown that these measures can make people feel lonely. A Harvard University survey showed that people's loneliness increased significantly during the epidemic. Among the 950 Americans surveyed, 36% have serious loneliness, especially young people and mothers with young children [1]. Such a common mental health problem should arouse people's attention. Loneliness is a negative emotion, people often feel empty, unwanted and alone. It may increase health risks or cause corresponding mental health problems, such as depression and anxiety[2]. Some studies have shown that listening to comfortable music can alleviate people's loneliness[3]. And increasing opportunities for communication may also help. Therefore, I want to design a chatbot to communicate with people who feel lonely in a more natural way, and recommend suitable music to him to solve his emotional problems.

1.2. Existing Research and Products

With the development of artificial intelligence, chatbot has become very popular in recent years. In the business field, many companies develop chatbots to replace the work of manual customer service and reduce labor costs. Corresponding technologies are also widely used in policy consultation, medical health servers and other fields. Some products have also entered our lives, such as Apple's Siri[4], Google Assistant[5], and Microsoft's XiaoIce [6].

In the field of mental health, there are some specialized chatbots, such as woebot[7], which is a relationship agent that can help adults monitor and manage symptoms such as stress, depression, and anxiety and provide emotional support. It is based on some treatment methods in psychology, such as Cognitive Behavior Therapy, Dialectical Behavior Therapy, Interpersonal

Psychotherapy. Another widely used chatbot is called wysa[8], which has similar functions to woebot. But in this app, users can choose to communicate with a human mental health professional.

These tools are mainly based on chat, using some preset dialogues or materials (such as music, meditation guidance) to solve mental health problems. It is not possible to recommend more personalized content based on the user's state, meanwhile, there is currently no chatbot specifically designed to solve the problem of loneliness. This is exactly what we want to achieve.

1.3. Main Works

In this project, my work mainly has the following three aspects:

- Use the rasa framework to build a usable chatbot. It interacts with users through the web, can more accurately understand the user's intentions, and achieve smooth conversations. The method of filling in psychological scale through dialogue and multi-round dialogue based on mixed mode are realized.
- The playlist, song and comment data of NetEase Cloud Music are collected through crawler technology, which is used to predict the emotion expressed by the song based on the song comment.
- A sentiment classification algorithm based on LSTM network is designed, which can identify the emotional state contained in the text. Two recommendation rules are constructed to realize the recommendation of appropriate music to users according to their emotions.

2. The Construction of Chatbot

2.1. Framework of chatbots

The core part of Chatbot is natural language understanding. At this stage, there are some mature frameworks and solutions for chatbot development. I investigated three more commonly used chatbot development frameworks:

- Juji[9] is a platform that can help users quickly create chatbots. It provides a large number of pre-made chat libraries. Users can quickly create chatbots through a graphical interface without writing code. It can also infer the user's personality and characteristics through dialogue. For developers, because it provides a limited API, the flexibility is low, and some complex functions are difficult to implement.
- Dialogflow[10] is a natural language understanding platform developed by Google. Thanks to Google's technology accumulation in this field, it provides models with high accuracy, good robustness, and support for multiple languages. This platform has clear documentation and easy-to-use functions. Many chatbot projects are developed based on dialogflow, including Google Assistant. However, the current ES version provided by google has limited functions，which not meet the needs of this project.
- Rasa[11] is an open source chat assistant development framework, based on many well-known machine learning frameworks such as spacy tensorflow. It is powerful, flexible and easy to use. Compared with other frameworks, it integrates natural language understanding and dialogue management. Developers can customize various actions by writing python code to complete the interaction with users and realize very complex functions.

After actual trials and comparisons, because RASA is an open source framework with strong flexibility and richer dialogue management functions, I finally chose RASA as the development framework for this project.

2.2. Concept and Configuration

To build a chatbot using rasa, we first need to understand some of these concepts. In the message entered by the user, what the user wants the chatbot to accomplish is called an intent (such as greeting, asking the weather), and some of the keywords are called entities (such as time, place, number). When the user's information is received, one step that the chatbot needs to complete is called an action. Actions usually consist of text replies and custom actions.

The core part of a chatbot is natural language understanding, which responds to user requests by identifying the intent and entity entered by the user and using specified actions according to predefined rules. In this project, users communicate with chatbots through Mandarin language. Therefore, in the configuration of the natural language understanding module, the spacy-based pre-training language model is mainly used to represent the text input by the user as a vector, and its tokenizer and entity extractor functions are also used. The intent classification uses the officially provided Dual Intent Entity Transformer (DIET) as the response selection, and the ResponseSelector is used to select response.

2.3. Dialogue management

For chatbots, the most common and easy mode to implement is F&Q, that is, the user asks questions to trigger the rules, and the chatbot generates a response. We hope that the conversation with chatbot can be more complex, such as being able to actively and continuously talk to the user instead of just relying on the user’s input to trigger; or reply to the message according to the user’s input in the previous rounds of conversations. which requires design Dialogue management function. To achieve these, we need to carefully design dialogue management.

In the rasa framework, the path of dialogue is specified by writing rules and stories. Rules are used to specify short conversations with clear paths. A story can be seen as a collection of a series of intentions and actions, used to define a business scenario that requires multiple rounds

of dialogue to complete. Rules and stories, as a dialogue management method provided by rasa, have great limitations in handling multiple rounds of dialogue. It cannot realize more complicated path configuration or multi-layer condition judgment. Therefore, I used python to write the dialog path selection rules. The core of it is to implement a decision tree, and respond to the user by selecting the corresponding action according to the user's state at different node. The user can click the button to trigger the corresponding action and select a specific path. Combining the rules and stories of rasa, a more complex dialogue logic can be realized.

I also designed other ways to trigger dialogue, such as the user opening the chat window to trigger a specific chatbot action to achieve active dialogue. For user messages whose intent cannot be effectively identified, I forward them to the small chat API provided by Tencent Cloud, and obtain appropriate responses from the interface. This situation is mainly used to reply to the user's small chat.

2.4. Form

In order to evaluate the user's emotional state more professionally, the UCLA Loneliness Scale[12] is used to measure user's loneliness. The conventional method is to let users fill out a web form. We changed the form of interaction and let users answer the questions of chatbot to complete the questionnaire. The original UCLA Loneliness Scale had 20 items. Due to too many problems, some shorter versions were later revised[13]. In order to balance the number of questions and accuracy, we chose a version with 9 questions.

Rasa provides a function of collecting information from users in the form of question and answer, called from. The main principle is to set some slots for the chatbot, and the slots can be regarded as the memory of the chatbot. When these slots are filled, it can be considered as a form submission. After that, the form data can be processed through custom actions.

2.5. Interface and Interaction

In order to facilitate the user's use, I use webpages to build the user's interface. The website is developed using the flask[14] framework and has basic user login and registration functions. The chat interface is implemented using a set of open source components called rasa-webchat[15] and interacts with rasa core through websocket. Through loads that is not visible to the user, the display of videos, pictures, and selection buttons can be supported. The front end of the website is developed using the bootstrap framework[16] and a template[17]. The page adopts a responsive layout, which is convenient for users to use on PC and various mobile devices.

3. Data and Model

To recognize the user's emotions from the user's conversation and recommend appropriate music to him, I encountered several challenges. First, I don't have any labeled data, second, how to identify the emotions contained in music, and finally how to construct recommendation rules. In order to solve these problems, I start with the music data and construct the required data set.

3.1. Data Collection and Preprocessing

NetEase Cloud Music is a very popular music platform in the Chinese community. This platform is rich in music and has accumulated a large number of music reviews. The platform has a very good community atmosphere, users like to share their stories and emotions in the comments. So I use this platform as the data source. Since the platform does not open APIs, I use the method of simulating client requests to obtain the corresponding data. Build a local service by node.js to forward the request and get the data response in json format.

There are three types of data we collected: playlists, songs, and song comments:

- Playlists are created by users and contains a list of songs on specific topics or preference. Each playlist has multiple tags marked by the creator. I collected 12 kinds of hot playlists under the emotional label (Nostalgia| Fresh| Romantic| Sad| Healing| Relaxing| Solitude| Moving| Excitement| Happy| Quiet| Missing) and 4 kinds of age label (1970s|1980s|1990s|2000s).
- The songs data includes song name, author, source url, and lyrics. All songs are obtained from the playlists collected above. Since these songs do not contain tags. We mark tags of the playlist on the song. For the same song from different playlists, we merge all the playlist tags as the song's tags. I collected a total of 15,325 songs. The labeled songs are shown in Table 3.1

Table 3.1 The Top10 Songs That Appears The Most in Playlists

Id	Song Name	Count	Tags
1	城南花已开	21	伤感,学习,钢琴,电子,浪漫,清新,思念,怀旧,治愈,放松,孤独,轻音乐,安静
2	夏の喚く	21	伤感,学习,钢琴,浪漫,旅行,清新,怀旧,治愈,放松,轻音乐,游戏,安静
3	海の形	20	伤感,世界音乐,清晨,学习,钢琴,浪漫,爵士,下午茶,清新,怀旧,治愈,放松,轻音乐,游戏,安静
4	夜、萤火虫和你	20	伤感,清晨,治愈,学习,流行,夜晚,浪漫,旅行,钢琴,清新,怀旧,放松,轻音乐,游戏,工作
5	The truth that you leave	19	伤感,学习,流行,钢琴,00 后,浪漫,旅行,工作,清新,思念,怀旧,治愈,放松,华语,孤独,轻音乐
6	Illusionary Daytime	18	伤感,学习,流行,说唱,钢琴,浪漫,欧美,清新,思念,影视原声,怀旧,治愈,放松,孤独,轻音乐,工作
7	星茶会	18	伤感,清晨,治愈,学习,浪漫,清新,怀旧,放松,轻音乐,游戏
8	寂静之空	17	伤感,清晨,治愈,学习,钢琴,旅行,清新,怀旧,放松,轻音乐
9	夏野与暗恋	17	伤感,学习,钢琴,浪漫,旅行,清新,怀旧,治愈,放松,轻音乐,安静
10	忆夏思乡	16	伤感,清晨,治愈,学习,钢琴,浪漫,旅行,清新,怀旧,放松,轻音乐

- The comment of songs includes the comment time, content, nickname of the commenter, and the number of comment likes, totaling 1,778,560.

The collected comment text includes a large amount of noise information such as network terms, emoji and emoji composed of special symbols(ヽ(●´ω｀●)/). Therefore, in the data preprocessing, only the Chinese, English, numbers and common punctuation marks in the word segmentation results are retained to obtain a usable comment dataset. The comments processed are shown in table 3.2.

Table 3.2 Most Popular Comments

Song	Comment	Liked Count
广东十年爱情故事	现在北京时间 7.28，我家傻丫头还在睡觉，听着她睡觉的呼吸声，特别安稳，很庆幸我能做到让她有安全感，我在努力，不让她受伤难过委屈，但是老惹她生气嘿嘿，不知道我还能活多久，胃癌晚期嘛，我爱她，真的很爱很爱，所以很珍惜一起相处的时光，她那么可爱，想想听不到她声音了就想哭	1826465
城南花已开	君安在，谢谢十二亩地，谢谢所有关心我的人，谢谢你们！早上因为治疗现在才看见，我会坚持的！	1507501
年少有为	假如我年少有为…	1379551
红色高跟鞋	“像被子里的舒服，又像风捉摸不住”不是屁是什么？[呆]	1147060
大鱼	真的觉得中国人很奇怪！[撇嘴]对自己国家的文化没有任何自信 为什么好不容易中国有了不错的作品 国人就开始诋毁 说什么抄袭千与千寻，但是大鱼的故事 还有人物的背景 和千与千寻千差万别 而且都是中国自己的东西 客家土楼 山海经 凤凰 龙 还有逍遥游 这都是满满的中国风 ！	998463
城南花已开	前段时间一位喜欢我音乐的朋友私信我，说他得了骨癌晚期还有半年的时间，希望我用他云音乐的 ID 写首曲子，他很喜欢他的 ID，然后发给我了很多医院的照片怕我不相信，我看到消息的时候正好我没有设备和乐器去创作，把运输的设备弄好第一时间去创作，终于完成了！送给城南花已开，希望你能听到，一切都好！	970760

3.2. Data analysis

Many comments describe the emotions of users after listening to the song, but they are not labeled, it is difficult to apply them in practice. I want to know what themes and emotions are covered by these comments, so I built an unsupervised machine learning model-the LDA

model[18]. LDA is also called a document topic generation model, which is used to identify potential topic word information in a document set. It can give the topics in the document set in the form of probability distributions. I tested the number of designated topics as 5, 10, and 15. For each topic, the model can find some words to describe it. From the results obtained(Fig3.1), it can be seen that in addition to the evaluation of the song itself, the user's comments also include themes describing “academic concerns”, “relationship issues”, “loneliness”, and “stress”. Therefore, it is possible to infer the mental state of the listener of the song from the comment, who is more likely to resonate with these music.

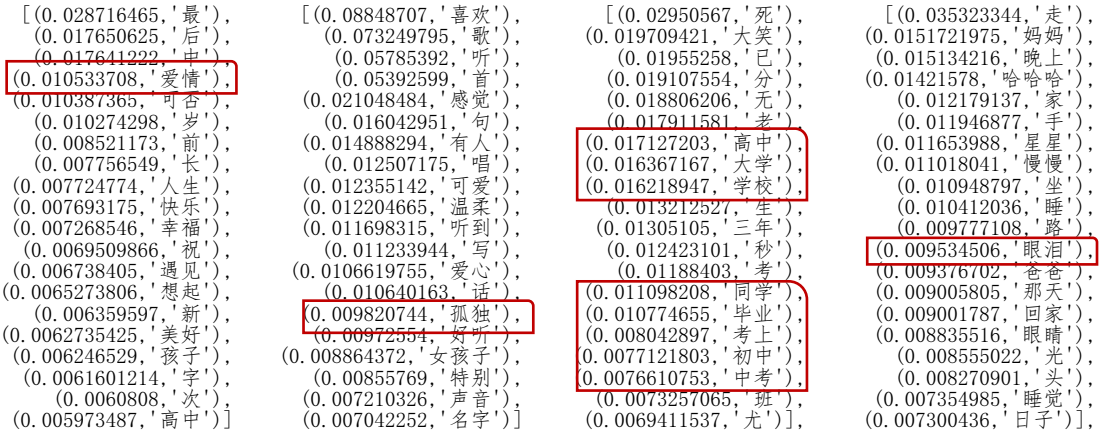

Fig 3.1 The Result of LDA about Relationship issues, Loneliness, Academic concerns, Stress

I also analyzed the time distribution of these comments, As shown in Fig3.2, we can clearly see that users prefer to post comments at night, and we can also speculate that some users have problems with insomnia.

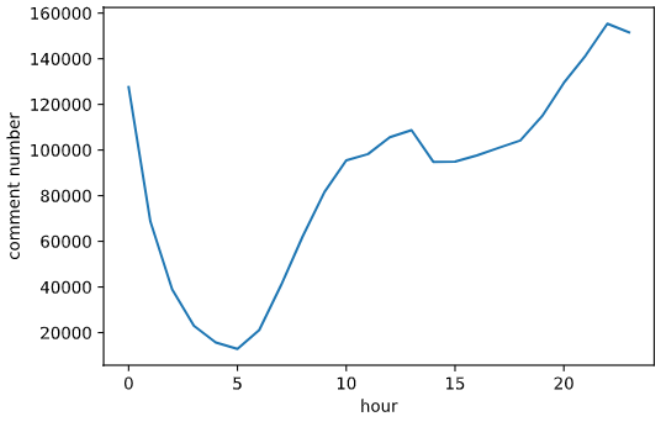

Fig 3.2 The time distribution of user comments

3.3. Classification model based on LSTM

3.3.1 Classification Dataset

Through the above data analysis, it can be seen that there is a certain connection between music and the user's mental state and psychological problems such as "insomnia", "academic concerns", "relationship issues", "loneliness", and "stress. Since these comments do not contain tags, it is difficult to directly extract which comments reflect the above-mentioned five dimensions of mental state. In order to classify these comments, I found an open source dataset—Emotional First Aid Dataset[19]. This data comes from the Q&A of users in the psychological community. Professional psychologists mark the psychological problem tags of these posts. The tags of this data contains the mentioned insomnia, stress, relationship issues, and academic concerns. With the labeled dataset, detecting the user's mental state and psychological problem in the text content becomes a classification problem. The original dataset contains the user's question description and other users' replies. I put the user's question and other people's replies together as an article. I built an SVM model to test the classification effect. The results obtained are shown in Figure 3.3.

Id	*precision*	*recall*	*f1-score*	*support*
0	0.38	0.06	0.11	141
1	0.57	0.83	0.67	2134
2	0.82	0.25	0.39	126
3	0.61	0.84	0.71	1483
4	0.00	0.00	0.00	50
5	0.62	0.07	0.12	266
6	0.00	0.00	0.00	124
7	0.55	0.47	0.51	609
8	0.00	0.00	0.00	62
9	0.67	0.07	0.12	266
10	0.55	0.24	0.34	74
11	0.61	0.36	0.45	224
12	0.31	0.05	0.08	109
13	0.00	0.00	0.00	229
14	0.00	0.00	0.00	32
15	0.00	0.00	0.00	10
16	0.00	0.00	0.00	19
17	0.00	0.00	0.00	17
18	0.00	0.00	0.00	25
accuracy			0.58	6000
macroavg	0.30	0.17	0.18	6000
weightedavg	0.53	0.58	0.51	6000

Fig3.3 The result of SVM

No dataset about loneliness was found in the public dataset in Chinese, so I used manual labeling. First, build a "lonely" synonym table[6], which comes from the synonym dictionary and Word2vec model. Using this synonym table as a keyword list(Table3.3), search in 200,000 comments, filter and de-duplicate the results, and get a total of 405 records. Finally, the dataset is shown in Table 3.4.

Table 3.3 List of synonyms for loneliness

孤单	孤独	孤寂	伶仃	寥寥	寂寂
只身	一身	独身	光杆儿	孤零零	
孤苦伶仃	伶仃孤苦	形影相吊	形单影只	孑然一身	孤家寡人
举目无亲	六亲无靠	无依无靠	单枪匹马	匹马单枪	离群索居
孤立无援	顾影自怜	单人独马	孤单单	孤儿寡母	孤僻
孤身					
孤身一人	孤孤单单	形影相对	孤苦伶丁		

Table 3.4 Overview of the Classification Dataset

Tag	**Items**
Insomnia	406
Stress	789
Academic Concerns	659
Relationship Issues	4968
Loneness	405

3.3.2 LSTM Model

When we have the labeled data set, the problem now becomes a classification problem. But the previous SVM model did not work well. In order to improve the accuracy of classification, I used a neural network model and made some optimizations.

The first is the representation of the text. In order to improve the accuracy, I used a word embedding model[19] with 8 million words provided by Tencent AI Lab. Using this as a dictionary for word segmentation can accurately extract many network terms, and then represent each word as a 200-dimensional vector.

Secondly, I used a variant of the LSTM model called GRU Network Model, which has better performance in short text classification. The whole network structure is shown in Figure3.3. Through optimization, the accuracy of the model in the verification set reaches 94%.

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 80, 200)	3943600
spatial_dropout1d (SpatialDr	(None, 80, 200)	0
gru (GRU)	(None, 80, 100)	90600
gru_1 (GRU)	(None, 100)	60600
dense (Dense)	(None, 1024)	103424
dropout (Dropout)	(None, 1024)	0
dense_1 (Dense)	(None, 1024)	1049600
dropout_1 (Dropout)	(None, 1024)	0
dense_2 (Dense)	(None, 6)	6150
activation (Activation)	(None, 6)	0

```
results = model.evaluate(xtrain, ytrain, batch_size=128)
print("test loss, test acc:", results)
```

```
49/49 [==============================] - 3s 52ms/step - loss: 0.1608 - acc: 0.9415
test loss, test acc: [0.16081570088863373, 0.941469669342041]
```

Fig 3.3 GRU Model and Accuracy

4. Recommendation Rules

The classification model mentioned above is applied to two parts, one is to detect the user's emotion from the conversation between the user and the chatbot. When it is detected that the user has the above five types of problems, the corresponding recommendation method is triggered. The other is to determine which psychological problems are reflected in the song's comments. Here I adopted a rule-based hybrid recommendation method.

At the same time, the comment time of the song is averaged, divided into 24 categories in hours, and converted into a component using the average comment hour/24. When the user describes his mental state to the chatbot, a vector representation of the user's mental state is obtained through the GRU model, and the chat time is converted into the time component. The component of the mental state detected by the model and confirmed by the user is set to the maximum value, the cosine similarity between the vector and all the song vectors(Table 4.1) is calculated, and the song recommendation list is obtained from high to low similarity.

Table 4.1 The emotion vector of a song

Song	None	Insomnia	pressure	academic	relationship issues	loneliness
吻得太逼真	0. 0063895797	0. 033616386	0. 23097299	0. 17825805	0. 44323808	0. 10752477

The above recommendation method can recommend affect-congruent music and provide a sense of empathic for users. However, in some cases counter-productive effect. For example, the above rules believe that Spring Festival Overture makes listeners feel stressed (Fig4.1), because there are many homesick comments under this song, and many people expressed their desire to return home in silken robes. But obviously this is not appropriate, so we changed the recommendation strategy.

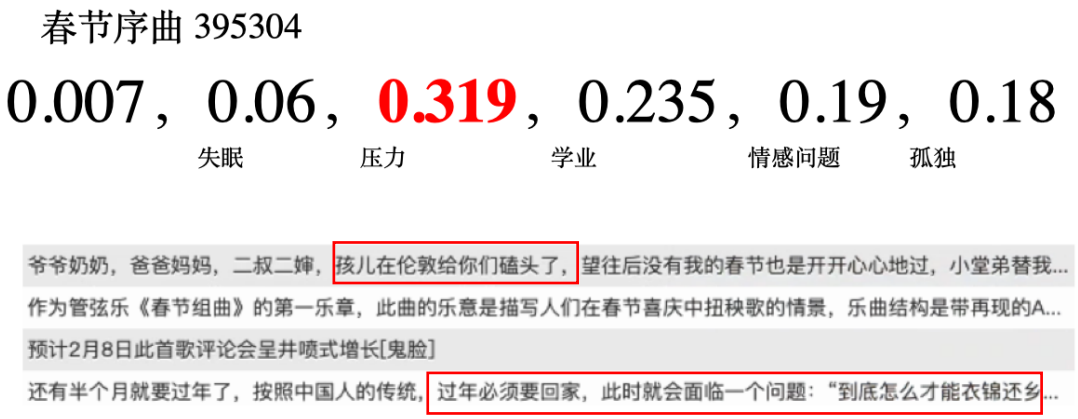

Fig 4.1 Emotion vector of Spring Festival Overture

Some psychological studies have shown that through nostalgic ways, such as listening to nostalgic music, loneliness can be alleviated[20]. Therefore, in order to solve the problem of loneliness, the age information obtained by the chatbot is matched with the age tag in the song tag to get the music list that meets the user's age, and the song list is recommended to the user according to the number of likes and from high to low.

Users can also search songs by talking to chatbot about the tags of songs they want to listen to. After extracting the tag of songs, the chatbot matches the song tags by searching, and recommends the playlist to users. In addition, users can also communicate with the chatbot to feed back the music they like and dislike, so as to match the user's music preferences to a greater extent.

5. System Design and Implementation

5.1. Overall Structure

The project is mainly divided into two parts, one part is the data collection and processing part, and the other part is the chatbot part. The overall structure is shown in Figure 5.1.

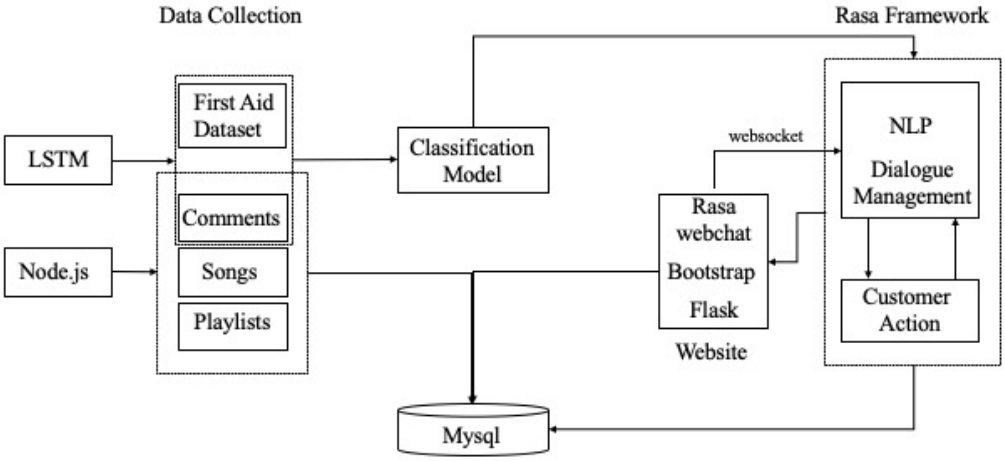

Fig 5.1 System Structure

The data analysis part is mainly the collected music data introduced in the previous article, and the model constructed from the data. The Chatbot part is the core of the project. It includes user pages, Natural Language understanding, custom actions, databases and other parts.

Let's take a user who feels lonely as an example. After the user logs in, the chatbot first asks the user how he feels today, and records the user's mood. There are 5 levels of mood, from happy to unhappy. If the user chooses to be unhappy, he will be asked what problem he has encountered. If the user chooses to be lonely, then enter the test of the loneliness scale. According to the test results, if the user is indeed in a lonely state, songs suitable for the user's state will be recommended to help the user relieve loneliness. The overall process is shown in Figure 5.2.

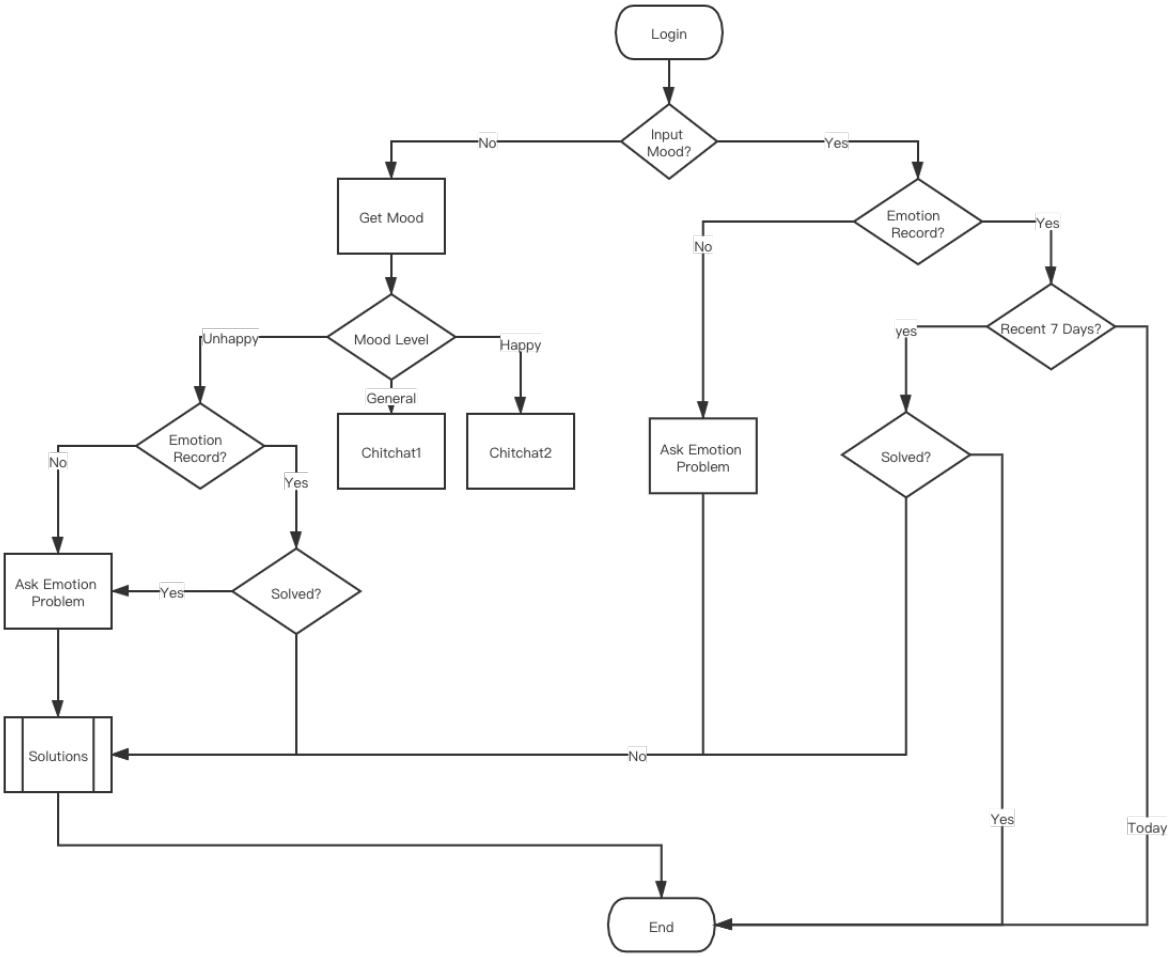

Fig 5.2. Dialogue Process

5.2. System Implementation

The main interface of the entire system is shown in Figure 5.3, click the button in the lower right corner to pop up the chat window.

Fig 5.3 Home Page

5.3. User Management

Users must first register an account to be able to use all functions. The user registration interface is shown in Figure 5.4. Users need to provide their own age for future song recommendations.

Fig 5.4 Registration Page

After registering an account, you can log in to the system to start a conversation with the chatbot. The login interface is shown in Figure 5.5.

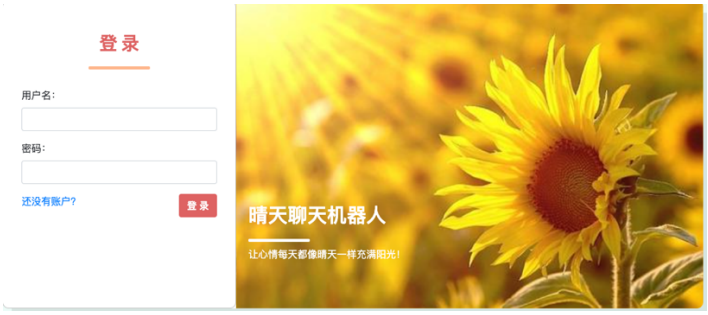

Fig 5.5 Login Page

5.4. Mood Tracking

If you talk to chatbot for the first time today, it will ask your mood, as shown in Figure 5.6.

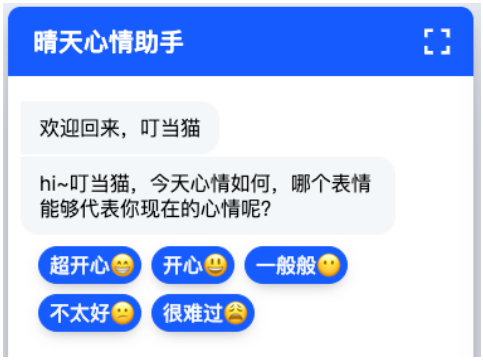

Fig 5.6 Mood Tracking

Users can choose their own mood, and the system will record it. Meanwhile, it will graph the user's mood changes in the recent period, as shown in Figure 5.7.

Fig 5.7 Mood Change Graph

If you choose unhappy, it will further ask you what problems you have encountered. As shown in Figure 5.8, you can choose the problem.

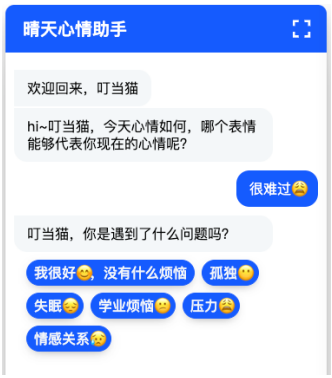

Fig 5.8 Emotion Problem

Suppose you choose loneliness, chatbot will enter the test of the loneliness scale, as shown in Figure 5.9.

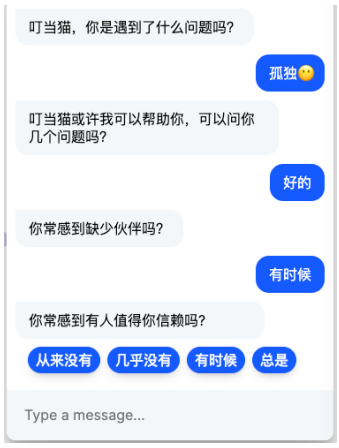

Fig 5.9 Loneliness Scale

When you answer all the questions, chatbot returns the test results, as shown in Figure 5.9. If your loneliness is high, chatbot will recommend some music for you(Fig 5.10).

Fig 5.10 Music Recommendation

If you choose other questions, chatbot will recommend music to you according to different rules.

5.5. Emotion Journal

Users can describe their experiences to chatbot and record their experiences like writing a diary. Chatbot will analyze this text and predict the problems the user encountered, as shown in Figure 5.11.

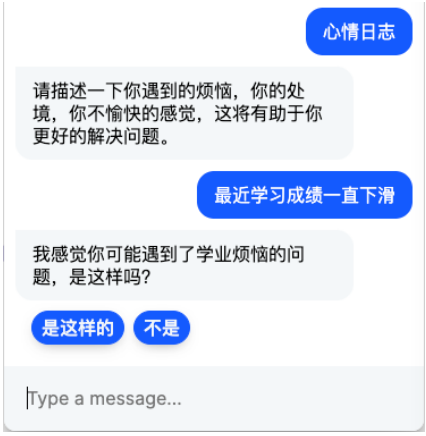

Fig 5.11 Emotion Journal

Users can view their own emotion journals on the webpage, as shown in Figure 5.5.

Fig 5.12 Emotion Journal View

5.6. Music Switch and Search

Users can talk to chatbot about the music they want to listen to search the music, you can say "Next" to switch songs, as shown in figture 5.13.

Fig 5.13 Music Switch

5.7. Chitchat

You can chat with chatbot, it will usually give you a suitable answer, as shown in Figure 5.13.

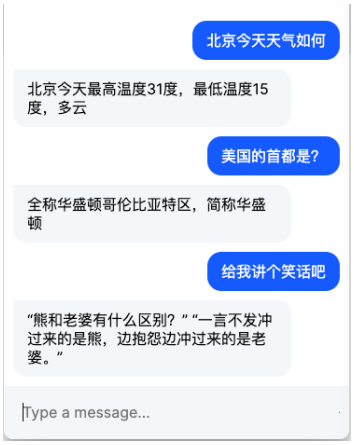

Fig 5.14 Chitchat

6. Conclusion and Future work

In this project, I mainly research and implement a chatbot based on natural language understanding. Through messages form the user to the chatbot to detect the user's emotions, and recommend appropriate music to help users improve mood. In order to solve the problem of music recommendation, through data collection and analysis, I determined the emotion categories to be classified, built and optimized a LSTM deep network model, and realized the emotion classification. Using the established emotional model, two kinds of recommendation rules are used to recommend music that can resonate to users. Due to the short time and my lack of in-depth understanding of relevant technologies, there are still many aspects to be improved in this project.

In the future, there are mainly the following areas worthy of improvement:

- The amount of data used in the construction of emotion classification model is less, and items in each category is imbalance, the generalization ability of the model is poor, so it is necessary to increase the data for training and establish a more perfect model.
- At present, the user's emotion is detected by one input of the user. In the future, we hope to use all the conversation messages of the user to more accurately identify the user's emotional state.
- Music recommendation method is relatively simple, so I should explore more complex and effective recommendation methods. And evaluate whether the recommended music meets the user's taste to help users improve their mood.
- There are still many limitations in the way of listening to music to improve mood. In the future, more effective psychological methods should be added.
- The interface of music playing is too simple, so a music playing component should be written to facilitate the user's music playing and switching.

Reference

[1] “Loneliness in America: How the Pandemic Has Deepened an Epidemic of Loneliness and What We Can Do About It,” *Making Caring Common*. https://mcc.gse.harvard.edu/reports/loneliness-in-america (accessed May 1, 2021).

[2] Cacioppo, John T., et al. "Loneliness and health: Potential mechanisms." Psychosomatic medicine 64.3 (2002): 407-417.

[3] K. Schäfer, S. Saarikallio, and T. Eerola, “Music May Reduce Loneliness and Act as Social Surrogate for a Friend: Evidence from an Experimental Listening Study,” Music & Science, vol. 3, p. 205920432093570, Jan. 2020, doi: 10.1177/2059204320935709.

[4] https://www.apple.com/siri/

[5] https://assistant.google.com/

[6] https://www.xiaoice.com/

[7] https://woebothealth.com/

[8] https://www.wysa.io/

[9] https://juji.io/

[10] https://dialogflow.cloud.google.com/

[11] https://rasa.com/

[12] Russell, Dan, Letitia Anne Peplau, and Mary Lund Ferguson. "Developing a measure of loneliness." Journal of personality assessment 42.3 (1978): 290-294.

[13] Russell, Daniel W. "UCLA Loneliness Scale (Version 3): Reliability, validity, and factor structure." Journal of personality assessment 66.1 (1996): 20-40.

[14] https://flask.palletsprojects.com/en/2.0.x/

[15] https://github.com/botfront/rasa-webchat

[16] https://getbootstrap.com/

[17] https://bootstrapmade.com/flexstart-bootstrap-startup-template/

[18] Mehrotra R, Sanner S, Buntine W, et al. Improving lda topic models for microblogs via tweet pooling and automatic labeling[C]//Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. 2013: 889-892.

[19] Yan Song, Shuming Shi, Jing Li, and Haisong Zhang. Directional Skip-Gram: Explicitly Distinguishing Left and Right Context for Word Embeddings. NAACL 2018 (Short Paper)

[20] Abeyta A A, Routledge C, Kaslon S. Combating Loneliness With Nostalgia: Nostalgic Feelings Attenuate Negative Thoughts and Motivations Associated With Loneliness[J]. Frontiers in psychology, 2020, 11: 1219.