

**ARTICLE**

# Design of Artificial Intelligence Companion Chatbot

**Xiaoying Chen<sup>1,\*</sup>, Jie Kang<sup>1</sup> and Cong Hu<sup>2</sup>**<sup>1</sup>College of Mechanical & Electrical Engineering, Sanjiang University, Nanjing, 210012, China<sup>2</sup>Guangxi Key Laboratory of Automatic Detecting Technology and Instruments, Guilin University of Electronic Technology, Guilin, 541004, China

\*Corresponding Author: Xiaoying Chen. Email: chenxiaoyingseu@163.com

Received: 08 September 2023 Accepted: 08 February 2024 Published: 28 March 2024

**ABSTRACT**

With the development of cities and the prevalence of networks, interpersonal relationships have become increasingly distant. When people crave communication, they hope to find someone to confide in. With the rapid advancement of deep learning and big data technologies, an enabling environment has been established for the development of intelligent chatbot systems. By effectively combining cutting-edge technologies with human-centered design principles, chatbots hold the potential to revolutionize our lives and alleviate feelings of loneliness. A multi-topic chat companion robot based on a state machine has been proposed, which can engage in fluent dialogue with humans and meet different functional requirements. It can chat with users about movies, music, and other related topics, and recommend movies and music that may interest them to alleviate their loneliness and provide companionship. The interaction platform of the companion robot is realized through the QQ communication platform, with two chat modes: Conversation mode and recommendation mode. First, the KdConv open-source corpus was selected, and Python was used to crawl information on movies and music from Douban and QQ Music to establish and pre-process the dataset. Then, the dialogue function was implemented using generative language models and retrieval systems, while the recommendation function was achieved using user profiling and collaborative filtering. Finally, a state machine algorithm was used to achieve real-time switching between the two chat modes of the companion robot. In conclusion, test participants gave high ratings for the accuracy of the companion robot's responses and the satisfaction with its content recommendations. Compared to traditional large-scale integrated models, this robot employs a state-machine framework to achieve diverse functions through seamless state transitions, thereby enhancing computational speed and precision. Additionally, the robot can recommend movies and music, providing companionship and alleviating loneliness for users, which is of great significance in modern society where interpersonal relationships are increasingly alienated.

**KEYWORDS**

Chatbot; natural language processing; information retrieval; recommendation system; generative dialogue

## 1 Introduction

The advancement of artificial intelligence (AI) and chatbots has presented fresh opportunities for enhancing human life. As these fields continue to make progress, it has become feasible to create



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

machines that can engage in natural conversation with people, comprehend their needs, and offer effective solutions. The concept of machines communicating with people can be traced back to the earliest days of computing. However, it was only with the advent of natural language processing (NLP) and artificial intelligence that chatbots started to become a reality.

The rise of urbanization and technology has resulted in people spending more time alone and feeling disconnected from others, which can lead to feelings of loneliness and isolation that may negatively impact their mental well-being. As a branch of chatbots, companion robots have the potential to alleviate people's feelings of loneliness and isolation. They can initiate conversations with individuals, enabling them to share their emotions and thoughts. The ability of chatbots to provide personalized responses based on user preferences and interests can foster a sense of companionship [1].

To design an effective chatbot that can alleviate loneliness, several key factors must be taken into account: Natural language processing (NLP) [2], Personalization, Emotional intelligence, Diversity of topics, and Usability and accessibility.

With the rapid advancement of deep learning and big data technologies, an enabling environment has been established for the development of intelligent chatbot systems. Initially, the proliferation of the internet and mobile internet has facilitated the accumulation of diverse conversational datasets from online platforms. Secondly, significant breakthroughs in deep learning (DL) technology, particularly in image processing and text translation, have contributed to extensive research on human-machine dialogue systems. By effectively combining cutting-edge technologies with human-centered design principles, chatbots hold the potential to revolutionize our lives and alleviate feelings of loneliness. However, current conversational robots still face limitations in language comprehension, context understanding, personalized and emotional understanding, data privacy and security, and multi-language support. Therefore, future research efforts are required to conduct in-depth exploration and innovation to address these challenges and enhance the performance and user experience of conversational robots.

A multi-topic chat companion robot based on a state machine has been proposed, which can engage in fluent dialogue with humans and meet different functional requirements. Compared to traditional large-scale integrated models, this robot employs a state-machine framework to achieve diverse functions through seamless state transitions, thereby enhancing computational speed and precision. Additionally, the robot can recommend movies and music, providing companionship and alleviating loneliness for users, which is of great significance in modern society where interpersonal relationships are increasingly alienated.

## 2 Related Work

Companion robots communicate with users through chat, attracting interaction between the user and the robot. In the early days, rule-based methods were used to output the chat content of companion robots. However, with the rapid development of machine learning and deep learning, two new methods have been developed: Generative models and retrieval-based systems. As these two methods gradually developed, they began to be used together.

In 1966, computer scientist Joseph Weizenbaum designed the ELIZA chatbot [3], which was first used in medical consultations. ELIZA collected information from user inputs and generated responses based on rule-based methods. Rule-based generation requires a lot of manual work to organize rules and is limited in terms of the content of the chat. With the rapid development of deep learning, the use of deep learning to find rule features reduces the need for a large amount of human resources and increases the flexibility and variability of chat topics, improving the accuracy of answering questions.

Vinyals et al. [4] used a generative method to produce chatbot dialogue content. The model generates the response that needs to be given to the user based on the input sequence using the Sequence-to-Sequence (Seq2Seq) model. However, due to the need to maintain response accuracy, generative content tends to be conservative, simple, and may lack coherence and logic.

Lu et al. [5] used a retrieval-based method to produce chatbot dialogue content, avoiding the need to generate new text by selecting appropriate responses from a pre-designed set of candidate responses. Retrieval-based methods are more accurate in their language, logic, and syntax than generative methods. However, when different users input the same question, the chatbot produces the same answer, limiting the diversity of chat topics and reducing interest among some users.

Qiu et al. [6] combined generative models with the retrieval-based method, processing the retrieved content using the generative model to produce more accurate, specific, and relevant chatbot responses. Liu [7] also used a combination of generative and retrieval-based methods to produce chatbot dialogue content, but their approach was to compare and score the outputs of both methods before replying to the user. However, as a companion robot, it not only needs to provide accurate responses but also needs to have a function that attracts users. Therefore, a recommendation system is needed to recommend content that may interest the user.

Lappromrattana et al. [8] presented how to design and implement a quick prototype of companion bots for elderly people. It is able to send emails, open YouTube to provide entertainment and remember the times an elderly person must take medicine and remind them is designed and implemented as a quick prototype.

At present, there are three difficult problems that need to be solved in the design of companion robots in China. One is that the content reply of the companion robot is single, only able to reply to a topic in one field, can only provide simple answers to multiple topic fields, and cannot answer related questions in depth; The second issue is that the accuracy and relevance of content replies to companion robots are not high, and the questions asked and answered by users are not very accurate. The third issue is that companion robots are unable to answer content that users are interested in, and can only answer related popular or general content. This article implements the companion robot to communicate with users on topics such as movies and music through the QQ communication platform and recommends music and movies that users are interested in.

### 3 Methodology

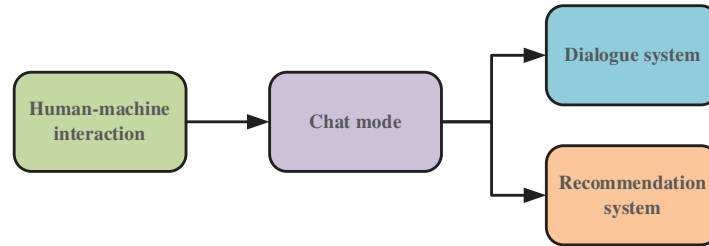
The designed chatbot is required to have strong entertainment capabilities and companionship features, targeting young users with a focus on entertainment-related topics. It should be able to engage in conversations with users about movies and music, retrieve relevant information, and provide recommendations on related content.

#### 3.1 System Scheme Design

Fig. 1 shows the four components of the system: Human-machine interaction, chat mode, dialogue system, and recommendation system.

**Human-machine interaction:** To meet the user's needs for an always-available companion robot, the human-machine interaction platform [9] for the companion robot is designed through the QQ chat platform, which allows sending text, images, and shared links. Tencent QQ is an instant messaging software used by over half of China's population. Integrating the companion robot with the QQ platform avoids the need for users to download other software and is convenient, making it easy for

initial dissemination. The simpler-robot open-source framework is used to log in to QQ with Java and obtain user messages and send messages actively. The companion robot runs on a personally built server, allowing users to chat with it anytime, anywhere via networked devices such as phones or computers by using the QQ communication software. Users can add the companion robot friend directly by QQ number, and after the friend request is accepted, they can chat with the companion robot.



**Figure 1:** System architecture of the companion chatbot

**Chat mode:** A fixed chat state [10] is unable to meet the diverse needs of companion robots, so this paper adopts multiple chat states to satisfy the various requirements of the chatbot. The artificially intelligent companion robot comprises two chat modes: Dialogue mode and recommendation mode.

The dialogue mode is based on the XLNet generative question-answering language model and a retrieval system. Preprocess the user input data and retrieve relevant content. Input the retrieved content into the generative model along with the user input question to obtain the generated content, which is then returned to the user.

The recommendation mode is implemented based on user profiles and collaborative filtering. Analyze the user's historical chat records, draw a user profile, generate recommended content through collaborative filtering, return it to the user, and obtain user feedback.

As there are two chat modes, the specific mode used for chatting with the user is maintained by a state machine, which decides based on the user's previous messages and the information they have sent. To maintain smooth transitions between these states, a state machine is used to implement topic flow switching. The state machine reads user messages in real-time, determining whether to enter a new mode or switch from one mode to another while already in a mode.

**Dialogue system:** The dialogue system needs to understand user input and select appropriate responses. This paper chooses XLNet [11], a widely-used Question & Answering generative natural language model. Due to the large amount of pre-trained data available for XLNet, this paper uses the Kdconv [12] open-source corpus as the fine-tuning dataset for XLNet. However, as the movie and music information in the Kdconv corpus is outdated, additional data is crawled from the Douban Movie website and QQ Music website using Python to supplement the response content of the companion robot. As a generative natural language model, XLNet requires not only the question but also related content for finding answers [13]. A retrieval system based on an Elasticsearch retrieval engine is used to provide relevant content for XLNet's questions. The search system is responsible for searching for information such as names of people, movies, songs, and brief introductions about them.

**Recommendation system:** To achieve better companionship, this paper also introduces a recommendation system [14] that can recommend movies and songs users may be interested in. The relevant data for the recommendation system is also crawled from the Douban Movie website and QQ Music website using Python to ensure the timeliness of the recommended content. During the initial chat

phase when the user is not yet understood, default hot content is recommended. As the chat history increases, the user profile [15] is gradually built. When users need recommendations, collaborative filtering [16] is used to recommend music and movies based on their profiles.

**System evaluation:** For evaluating the accuracy of the companion robot’s response content [17] and the reasonableness of its recommendations [18], this paper employs three test subjects to conduct a comparative test of the companion robot, Xiao Ai, and human communication. In terms of Q&A, the quality of the information provided in the feedback generated by the three systems is compared and evaluated. The information provided by the system is compared with the actual answers to assess the similarity, with high similarity indicating a correct response and low similarity indicating an incorrect response. For recommendations, test subjects choose their own interests as topics to test whether the returned content from the three systems is reasonable.

### 3.2 Database Construction and Data Preprocessing

Currently, there are three main open-source Chinese dialogue datasets: The Douban multi-turn dialogue dataset, which covers non-fixed chat content and belongs to casual conversation type; KdConv, a Chinese multi-domain knowledge-driven dialogue dataset containing movie, music, and travel domains; and Taobao customer service dialogue dataset, which is an e-commerce conversation corpus mainly related to product content. As the companion robot in this paper mainly chats about movies and music, KdConv is chosen as the primary dataset.

#### 3.2.1 Selection and Acquisition of Databases

The KdConv open-source corpus is chosen as the dataset for this paper. To make the robot’s message responses more diverse and real-time, content is crawled from Douban and QQ Music using Python crawlers.

KdConv is a Chinese multi-domain knowledge driven dialogue dataset, which includes three topics: Movies, music, and travel. There are a total of 4.5 K dialogues and 86 K sentences, with an average conversation round of 19.0. Because the companion robot mainly discusses topics related to movies and music, only the data on movies and music in KdConv was selected.

The KdConv dataset format is Json format, which can be parsed layer by layer to obtain dialogue content and related data. The following shows the conversation content message and the information attrs required to answer the user in the KdConv dataset. The information sent by the user without the attrs field, and the content answered by the user with the attrs field. In attrs, there are triples of attrname, attrvalue, and name. The content of the name field is the person name, movie name, song name, etc. The attrvalue field is the attribute value of name, and the attrname field mainly represents the relationship between the name field and the attrvalue field.

```
{
  "message": "Have you ever heard of Mervyn Leroy?"
},
{
  "attrs": [
    {
      "attrname": "birthplace",
```

```

    "attrvalue": " America",
    "name": "Mervyn Leroy"
  }
],
"message": "Yes, he is an American."
}

```

### 3.2.2 Data Preprocessing

The pre-filtering and noise reduction rules used in this paper are as follows:

- (1) Remove a large number of special symbols that are not Chinese, English, or numbers: Special symbols such as “-”, “\*” and “.” are removed.
- (2) Remove continuously enclosed whitespace characters: Meaningless whitespace text consumes a lot of performance, so a large number of whitespace characters in the dataset are removed.
- (3) Uniform file format: When inputting data into the model, various files need to be in a uniform format. The crawled information and KdConv dataset are unified in JSON format.
- (4) Convert English uppercase and lowercase letters: To make English expressions in the dataset uniform.

The Chinese word segmentation tool jieba is used for data set Chinese word segmentation. To avoid splitting certain movie and song names, a dictionary is built based on the data crawled in the second part; the named entity recognition function provided by the language technology platform [19] (LTP) is used; and the Chinese relation extraction open-source tool DeepK [20], developed by the Knowledge Engine Laboratory of Zhejiang University, is used to parse text.

### 3.2.3 Preparation of Database for Recommendation Mode

To ensure the companion robot’s recommended content is timely, all content was obtained through web crawling. The study used a web crawler to extract content from Douban and QQ Music platforms in preparation for the recommendation system.

## 3.3 Design of QA Function Based on XLNet Model and Retrieval System

This study used the XLNet generative language model to generate responses for the companion robot. Since XLNet is a generative model that requires both questions and answers to be input simultaneously, a retrieval system was used to search for answer-related content based on the user’s question. The pre-trained XLNet model was defined as a QA model and fine-tuned using the KdConv dataset processed in [Section 3.2](#), after which it was compared with other models.

An appropriate retrieval system was selected, and the data obtained in the [Section 3.2](#) was stored in the retrieval system. After preprocessing user input, relevant content was retrieved to provide answer choices for the XLNet model.

The XLNet generative natural language model is used to generate responses for the companion robot. As XLNet is a generative model that requires both questions and related content to generate answers, a retrieval system is used to search for answer-related content based on user input questions.

### 3.3.1 XLNet Model Design

The XLNet pre-trained model can be considered an optimized version of BERT, with a significantly increased training data scale during the pre-training phase. BERT [21] was trained on a 13G-scale dataset, while XLNet [22] added 19G-size data from ClueWeb and 78G-size data from [commoncrawl.org](https://commoncrawl.org).

In this paper, we use the `chinesexlnetbaseL-12H-768_A-12` model as the pre-trained model, which needs to be fine-tuned with our own data. The data source for the answer selection task is text pair data, i.e., the KdConv dataset prepared in [Section 3.2](#). During fine-tuning, only the task-specific input/output format needs to be sent to the XLNet model, and all parameters in the model are fine-tuned end-to-end, converting questions and related information into the input format of the XLNet model.

We train the model by inputting the training data into the language model, allowing the vector output of each position to capture semantic information in the sentence. The XLNet model captures deeper information hidden within words based on the vector of the word's position in the sentence.

In using the XLNet code provided by the HIT-IFLY Joint Laboratory, we fine-tuned the pre-trained model using our own prepared dataset and adjusted some parameters. First, the XLNet model is created using the code, and the pre-trained model provided by the HIT-IFLY Joint Laboratory is downloaded. The model is imported using the `load_trained_model_from_checkpoint` function of the `keras_bert` module, defined as a Q&A model, and input/output is defined, with optimization parameters set.

To test the model's ability to automatically answer questions, we divided the QA pairs from the dataset constructed in [Section 3](#) into three categories: Development set, test set, and challenge set (with longer dialogue lengths). We tested on five pre-trained models: BERT, BERT-wwm, BERT-wwm-ext, XLNet-base, and XLNet-mid. The results are shown in [Table 1](#), with evaluation metrics of EM/F1, which are important for evaluating various aspects of the model.

**Table 1:** Results of test

Model	Development set	Test set	Challenge set
BERT	64.5/79.1	70.0/87.0	18.6/43.3
BERT-wwm	66.2/80.3	70.5/87.4	21.0/47.0
BERT-wwm-ext	66.1/82.8	71.4/87.7	24.0/47.3
XLNet-base	65.2/84.5	67.0/87.2	25.0/51.3
XLNet-mid	66.8/88.4	69.3/89.2	29.1/55.8

From the results, it can be seen that XLNet has certain advantages over other BERT models in the Q&A field. Especially in terms of the F1 evaluation metric, it shows higher accuracy compared to other models. In the evaluation of the challenge set, the accuracy also sets a gap with other models.

### 3.3.2 Retrieval System Design

The database from [Section 3.2](#) and the crawled movie and music information are stored in Elasticsearch, an open-source retrieval database, to retrieve relevant content for answer selection in [Section 4](#). As the XLNet generative natural language model requires input from both questions and related content, analysis of the question is required to determine the relevant content. Preprocessing

methods, such as word segmentation, entity naming, and relationship extraction, are used to analyze the user's query. For example, if a user asks "Do you know what songs Xu Song sings?", after preprocessing, "Xu Song" is recognized as a name, and "What Can't Be?" is recognized as a song title. To provide relevant information for this query, we need to search for "Xu Song" as a name and "What Can't Be?" as a song title.

### *Retrieval System Architecture*

The mainstream open-source search engines are Elasticsearch [23] and Solr, with the biggest difference being that Elasticsearch supports distribution and has a much faster search speed when uploading data while searching than Solr. To maintain search capability while uploading data, Elasticsearch was chosen as the retrieval engine in this paper.

The data source for this system mainly comes from crawling relevant content from Douban and QQ Music websites. Fig. 2 shows that after preprocessing the data, the extracted data is packaged into JSON format to meet the data storage requirements of Elasticsearch. Then, the IK Chinese word segmentation tool provided by Elasticsearch is used to parse and segment the JSON data documents. An inverted index is built based on the segmentation results, and a global index database is built.



**Figure 2:** Framework of retrieval system process

### *Implementation of Retrieval Index*

The indexes are mainly divided into movies and songs. The movie index includes detailed information about the movie and its cast, while the song index includes detailed information about the singer and the song.

For example, the index data for the movie "Let the Bullets Fly" includes the unique ID from Douban, the director's name, the movie name, the genre, country/region, duration, alias, rating, release date, screenwriter, language, actors, synopsis, and number of movie ratings. Many data points can be used to search for a movie, but the primary search is by exact movie name. If there is no exact match for the movie name, a fuzzy search can be conducted.

```

{
  "id": 3742360,
  "director": " Jiang Wen",
  "title": "Let the Bullets Fly",
  "type": "Drama, Comedy, Action, Western",
  "country Of Production": "Mainland China, Hong Kong",
  "aka": "Let the Bullets Fly for a While, Piao Shao Yun, Let The Bullets Fly" ),
  "length": "132 minutes",
  "rate": 8.9,
  "release Date": "2010-12-16(Mainland China)",

```



```

    "screen writer": "Zhu Sujin, Shu Ping, Jiang Wen, Guo Junli, Wei Xiao, Li Bukong, Ma Shitu",
    "spoken Language": "Mandarin, Sichuan dialect, Shanxi dialect",
    "starring": "Jiang Wen, Ge You, Chow Yun-fat, Carina Lau, Chen Kun, Zhang Mo, Jiang Wu,
    Zhou Yun-Ming",
    "summary": "During the Republic of China era, Ma Bangde (Ge You), who became a county
    magistrate by donating money, travels with his wife...",
    "the Number Of Evaluations" : 1489514,
}

```

The above describes the creation of indexes for movies, which serve as the search entry point for the search system. The data listed above is used to create indexes that enable efficient searching of movie-related information.

### *Implementation of Retrieval System Search*

By storing movie and song related information in Elasticsearch, and after segmentation and indexing, the indexes can be searched using Elasticsearch's related code. For example, if a user sends a message such as "Are there any Chinese-style songs with a strong flavor?", named entity recognition selects "strong flavor" and "Chinese style". The system then searches for relevant results in Elasticsearch, and returns the song "Thousand Mountains and Rivers". This search result is inputted into the XLNet generative language model, which outputs the response, "You can try listening to the song 'Thousand Mountains and Rivers.'"

For song information, a GET request can be used to retrieve the QQMusicSongDetail index, and the brief Introduction section of the song information can be matched to the relevant information about "strong Chinese style." Using the Elasticsearch search system, the relevant song information for "strong Chinese-style songs" was found, providing effective data for XLNet's answer selection.

### **3.4 Design of Recommendation System Based on User Profile and Collaborative Filtering**

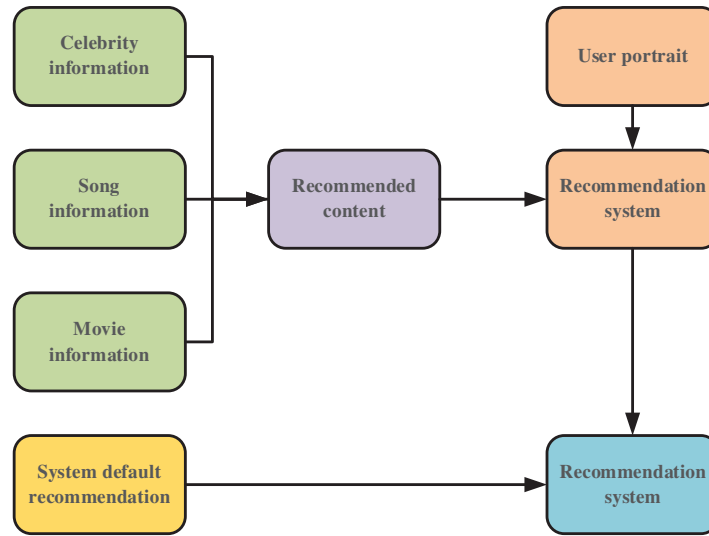
In this chapter, user preferences were first roughly outlined using a user profile, and then specific movies or songs were recommended to the user based on collaborative filtering. User profiles and collaborative filtering were used to recommend content that best meets the user's needs, increasing the frequency with which users interact with the chatbot.

#### *3.4.1 Architecture of Recommendation System*

**Fig. 3** illustrates the operating process of the recommendation system. For new users, the system will proactively recommend content using a default recommendation approach, which recommends recent hot topics. When recommending related movies, songs, or celebrities, the system reads the user profile and applies collaborative filtering to generate recommended content.

#### *3.4.2 User Portrait*

To accurately recommend content that users are interested in, user preferences are recorded through various means and forms. This includes which types of movies the user likes, which types of music they enjoy listening to, and which actors or singers they prefer. These records are related to movies and songs.



**Figure 3:** Workflow of the recommendation system

The user profile in this paper mainly records the user’s preferred movie genres, listened-to music genres, and favorite actors/directors/singers, as user preferences are continuously gathered during chat sessions with the companion robot for subsequent recommendation.

Table 2 shows the user profile of User A. The methods for building the user profile include requests, queries, and active recommendations. Request method means that the user does not request specific name information but requires certain types; query method means that the user requests specific name information, and active recommendation is when the robot actively recommends content to the user. To reduce table width, person names, song titles, and movie titles are combined as specific names. 1 means that the user declined the reply from the companion robot, and 0 means that the user did not decline. User A requested classic movies 5 times, requested American movies once, queried movie ratings 4 times, queried actors 3 times, and queried “Leslie Cheung” movies 3 times, with the most frequent topic being related to “Farewell My Concubine,” a total of 6 times. No music-related topics were asked, and the active recommendation of the popular song “Brave Warrior” was declined. These are all instances of User A’s chat history with the companion robot, and their user profile will be recorded for the next recommendation.

**Table 2:** User profile of User A

Number	Method	Specific ame	Type	Topic	Declined
5	Request		Classic	Movie	0
1	Request		American	Movie	0
4	Query		Ratings	Movie	0
3	Query		Actors	Movie	0
3	Query	Leslie heung		Movie	0
6	Query	Farewell My Concubine		Movie	0
1	Active Recommendation	Brave Warrior	Popular	Song	1

### 3.4.3 Collaborative Filtering

After data crawling and user profiling, concrete recommendation information needs to be generated. The common and effective collaborative filtering algorithm is used in this paper to generate recommended content.

Collaborative filtering is widely used in recommendation systems. This technique analyzes the correlation between movies/music or users to predict movies and music that users may be interested in, and recommends these movies and music to them.

Mahout [24] is used to implement the collaborative filtering algorithm. Mahout is a scalable and high-performance recommendation engine that implements basic user-based recommendation algorithms and meets the requirements of scalability, performance, and flexibility as a recommendation system.

The core program of collaborative filtering first reads ratings.csv, as shown in Table 3, which includes user ID, movie ID, user rating, and timestamp. Then, it calculates content similarity, constructs a recommendation engine, and passes in the user ID to be recommended to obtain the recommended movie IDs.

**Table 3:** Example content of ratings.csv

UserId	MovieId	Rating	Timestamp
1	296	5	1147880044
1	306	3.5	1147868817
1	307	5	1147868828
1	665	5	1147878820
1	899	3.5	1147868510
1	1088	4	1147868495
1	1175	3.5	1147868826
1	1217	3.5	1147878326
1	1237	5	1147868839
1	1250	4	1147868414
1	1260	3.5	1147877857
1	1653	4	1147868097

When recommending content to User A, the user profile that has been recorded is used to generate a CSV file. Using collaborative filtering, the recommended content is calculated: The movies “To Live” and “Ashes of Time Redux.” The movie “To Live” was recommended because users who like classic movies have given it high ratings. The movie “Ashes of Time Redux” was recommended because users who like Leslie Cheung as an actor have given it high ratings.

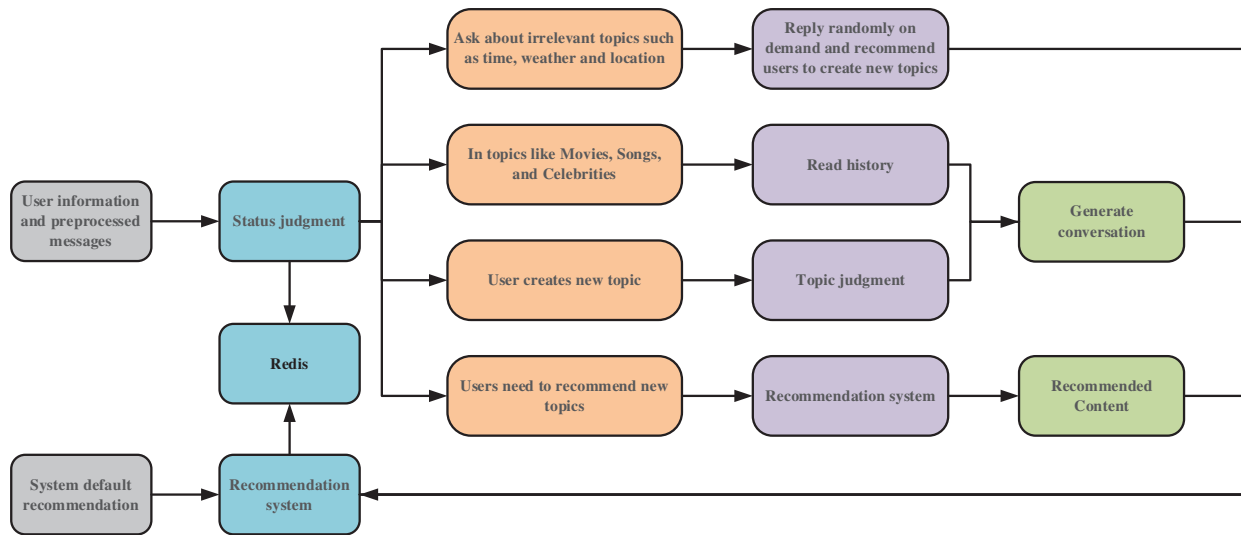
## 4 Results

### 4.1 State Machine Framework

To create a companion robot capable of holding fluent conversations with humans, implementing a large and comprehensive model that fulfills all functionalities is challenging. As more functionalities are added, the computational speed and accuracy of the model decrease. An alternative approach

is to meet different functional requirements through continuous state transitions. Deciding when to transition states and what state to transition is crucial. This paper utilizes Redis to store user communication states, with each user having their own distinct state. Redis, a key-value storage system, utilizes the user ID as the key and the state as the corresponding value. Redis offers fast read and write speeds, and can be remotely deployed over the network for data transmission and reception. However, compared to local storage, there is a risk of downtime, as the robot may fail to reply to user queries due to incorrect conversation states, significantly impacting user experience.

The state machine is divided into four states: Unrelated topics such as asking time, weather, and location; discussing movie, song, and celebrity topics; creating new topics and requesting recommendations for new topics from users. Fig. 4 shows the state machine process, which first determines the next step based on user information, pre-processed information, and the current state read. The chosen state generates dialogue content, sends the message, and records the state.



**Figure 4:** State machine process

#### 4.2 State Machine Process Display

Determining the state requires consideration of three aspects: User information, preprocessed messages, and content read from Redis. When it is identified that the user is asking irrelevant topics such as time, weather, and location, respond accordingly with random replies if necessary and recommend that the user create a new topic. When the user sends “What time is it now?”, the chatbot responds with “March 12, 2022” according to the current time and guides the user to continue communicating with “If you have any favorite movies, please let me know”.

When it is identified that the user is creating a new topic that is not related to the previous conversation, record the new topic in Redis. When the user sends “Do you know the movie Farewell My Concubine?”, the chatbot responds with “The Douban rating of ‘Farewell My Concubine’ is 9.6, and there are currently 1,913,376 people participating in the review”, and provides a link to the movie’s Douban page for the user to view details.

If the message is identified as being related to the previous conversation, the chatbot will continue discussing the topic if it is related to movies, songs, celebrities, and so on. When the chatbot reads that

the user is still discussing the movie “Farewell My Concubine,” and the user sends “Do you know the release date of this movie?”, the chatbot will query the release date of “Farewell My Concubine: July 26, 1993 (Chinese Mainland), January 01, 1993 (Hong Kong).”

When it is identified that the user needs a recommendation, the chatbot enters recommendation mode. Based on the requirements in the user’s message, the chatbot makes recommendations. When the user requests songs related to “Chinese”, the chatbot queries related “Chinese” songs and selects the appropriate content to reply to the user based on their user profile and collaborative filtering, “It is recommended that you listen to the album ‘Knock My Head’, sung by Peng Jiahui; released on December 05, 1998; song type: Rock; song language: Mandarin”. When the user continues to ask for information with “Which songs did Peng Jiahui sing?”, the chatbot switches to conversation mode and answers the questions asked by the user, “Peng Jiahui has sung: Hate for meeting too late, fire in the heart, liking two people, hearing that love has returned, Zhang San’s song, how did you let me feel sad, walking on the red carpet that day, aftertaste, Lugang Town”.

The chatbot will choose an appropriate time to proactively recommend songs and movies that the user is interested in and ask for feedback. For example, when the user had previously discussed topics related to “Jay Chou” with the chatbot, the chatbot proactively recommended a song by Jay Chou later on.

The above describes the decision-making process of the multi-state machine, which is used to implement the various functions and long conversation capabilities of the chatbot.

### ***4.3 System Evaluation***

To evaluate the rationality of the chatbot’s response content, a control group test was conducted. Twenty test participants were asked to compare the responses of the chatbot, Xiao Ai and human communication. The first-grade indexes for scoring were functionality, interactivity, scalability, and user experience.

Functionality refers to whether chatbots can complete the tasks required by users. This project adopts two-second index, movie topic, and music topic, to assess the accuracy, timeliness, and efficiency of completing tasks.

Interactivity refers to the interaction ability and naturalness of chatbots, including language processing and generation, sentiment analysis, context understanding, and other aspects. This project adopts a second index of user dialogue fluency and coherence.

Scalability refers to its ability to adapt to different scenarios and user needs, as well as its ability to quickly adjust and optimize with changes in user scale and business needs. This project adopts a recommendation system as a second index.

User experience is an important indicator for evaluating chatbots, including interface usability, readability, and aesthetics. This project adopts the usability and aesthetics of the interface as a second index.

And score them on a scale of 1–10 in terms of the secondary indicator. [Table 4](#) shows the average scores given by the test participants.

In terms of recommendation system testing, Xiao Ai performs poorly as all recommended content is fixed and not based on the user’s interests. Human interaction provides more specific and detailed reasons for recommendations compared to the companion robot.

**Table 4:** Average scores given by test participants

First-grade indexes	Functionality		Interactivity		Scalability	User experience	
Second index	Movie topics	Music topics	Fluency	Coherence	Recommendation system	Usability	Aesthetics
Chatbot	7	8	7	8	8	8	7
Xiao Ai	6	7	8	7	5	7	8
Human Communication	9	8	6	9	9		

In movie topic chat testing, Xiao Ai can engage in simple Q&A but cannot maintain a continuous conversation, while the companion robot has less information obtained from human interaction, making it harder to discuss plot details. Therefore, the companion robot scored lower in this project.

In music topic chat testing, Xiao Ai performs relatively well but lacks depth in content replies. As music-related conversations involve a lot of content, human response time may be slower, while the companion robot can reply relatively quickly.

## 5 Conclusion

As the distance between people continues to grow, communication becomes less frequent, leading to an increase in feelings of loneliness. The demand for companion robots is increasing every year, and their application is vast, creating a significant market opportunity.

This article explores the use of accompanying robots that engage users in discussions about movies and music via the QQ communication platform. This robot has the ability to recommend music and movies that align with users' interests. Here is a breakdown of the article's key points:

(1) This article selects the KdConv multi-topic dataset and enriches it by scraping data from the Douban Movie and QQ Music websites. It employs state machine techniques to switch between different topics, enabling companion robots to queries related to movies and music, and provide personalized recommendations for content like movie soundtracks.

(2) To generate dialogue content for companion robots using XLNet generative natural language model and retrieval system, we first analyze the user's input content. Then we retrieve relevant information through the retrieval system and combine it with the user's input. This combined information is fed into the XLNet model to generate reply content. After fine-tuning the retrieval system and training the XLNet model, the generated reply content is highly relevant and accurate to the user's questions.

(3) Adopt user personas and collaborative filtering algorithms to generate content that users are interested in. Read the chat of users to depict their persona features, and then use the features generated by user personas to enter the collaborative filtering algorithm, to generate recommended content.

The companion robot designed in this article only focuses on providing a Q&A on topics related to movie music, with a slightly narrower scope of topics. To enhance the adaptability of companion robots in various chat environments, it is essential to expand the range of chat topics, thereby broadening their usage. The training time of generative natural language models is extensive, leading to a high computational load. To facilitate the widespread use of robots in the future, the model requires optimization for improved efficiency. The recommendation system lacks comprehensive user information which prevents accurate profiling and satisfactory recommendations in the short term.

**Acknowledgement:** We would like to thank W.J. Wang for her technical support during the preparation of this manuscript.

**Funding Statement:** This work is supported by the Guangxi Key Laboratory of Automatic Detecting Technology and Instruments (YQ21207) and the Qinglan Project of Jiangsu Province.

**Author Contributions:** The authors confirm their contribution to the paper as follows: Study conception and design: Xiaoying Chen, Jie Kang; data collection: Cong Hu; analysis and interpretation of results: Xiaoying Chen; draft manuscript preparation: Xiaoying Chen, Jie Kang. All authors reviewed the results and approved the final version of the manuscript.

**Availability of Data and Materials:** Data sharing is not applicable to this article as no new data were created or analyzed in this study.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] Q. Zhai, L. Feng, G. F. Zhang, M. Liu, and J. J. Wang, "The application of artificial intelligence in the field of mental and psychological health," *Zhejiang Med. J.*, vol. 42, no. 10, pp. 1078–1084+1091, 2020 (In Chinese).
- [2] V. Gupta, "A survey of natural language processing techniques," *Vishal Gupta Int. J. Comput. Sci. Eng. Technol. (IJCSET)*, vol. 5, no. 1, pp. 14–16, 2014.
- [3] H. Chen, X. Liu, D. Yin, and J. Tang, "A survey on dialogue systems: Recent advances and new frontiers," *SIGKDD Explor.*, vol. 19, no. 2, pp. 25–35, 2017. doi: [10.1145/3166054.3166058](https://doi.org/10.1145/3166054.3166058).
- [4] O. Vinyals and Q. Le, "A neural conversational model," in *Proc. 31st Int. Conf. Mach. Learn.*, Lille, France, 2015, vol. 37.
- [5] Z. Lu and H. Li, "A deep architecture for matching short texts," in *Proc. 26th Int. Conf. Neural Inf. Process. Syst.*, Red Hook, NY, USA, 2013, pp. 1367–1375.
- [6] M. Qiu, F. L. Li, S. Wang, G. Xing, and C. Wei, "AliMe chat: A sequence to sequence and Rerank based chatbot engine," in *Proc. 55th Annu. Meet Assoc. Comput. Linguist.*, Vancouver, Canada, 2017, pp. 498–503.
- [7] B. L. Liu, "Research and implementation of chat robots based on search and generation," M.S. dissertation, Shenyang Normal University, China, 2021.
- [8] T. Lappromrattana and P. Sooraksa, "Quick prototyping of companion bots for elderly people," *Sensor Mater.*, vol. 35, no. 4, pp. 1487–1495, 2023. doi: [10.18494/SAM4268](https://doi.org/10.18494/SAM4268).
- [9] X. Dai, "Research on usability design of human-computer interaction interface for children accompanying robot," M.S. dissertation, Tianjin Polytechnic University, China, 2019.
- [10] F. Y. Shen, "Research and implementation of customer service robots facing the financial field," M.S. dissertation, University of Electronic Science and Technology of China, China, 2021.
- [11] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov and Q. V. Le, "XLNet: Generalized autoregressive pretraining for language understanding," in *33rd Conf. Neural Inf. Process. Syst. (NeurIPS 2019)*, Vancouver, Canada, 2019, pp. 5754–5764.
- [12] H. Zhou, C. Zheng, K. Huang, M. Huang, and X. Zhu, "KdConv: A Chinese multi-domain dialogue dataset towards multi-turn knowledge-driven conversation," arXiv preprint arXiv:2004.04100, 2020.
- [13] X. S. Li, "The design and implementation of question answering system based on retrieval and answer generation hybrid," M.S. dissertation, Zhejiang University, China, 2019.
- [14] P. Fei, "Research on user profiling construction," M.S. dissertation, Dalian University of Technology, China, 2017.

- [15] X. S. Li, “Design and implementation of movie recommender system based on dialogue system,” M.S. dissertation, Beijing University of Posts and Telecommunications, China, 2020.
- [16] T. Yao, “Research on personalized recommendation based on collaborative filtering algorithm,” M.S. dissertation, Beijing Institute of Technology, China, 2015.
- [17] Z. P. Deng, “Study on multi-turn response selection for retrieval chatbots based on deep learning,” M.S. dissertation, Chongqing University, China, 2020.
- [18] L. L. Lu, “Research on recommendation algorithm based on score correction and topic community detection,” M.S. dissertation, Shanghai Normal University, China, 2021.
- [19] W. X. Che, Z. H. Li, and T. Liu, “LTP: A Chinese language technology platform,” in *23rd Int. Conf. Comput. Linguist.*, Beijing, China, 2010, pp. 13–16.
- [20] N. Zhang *et al.*, “DeepKE: A deep learning based knowledge extraction toolkit for knowledge base population,” in *Conf. Empir. Methods Nat. Lang. Process.*, Abu Dhabi, United Arab Emirates, 2022.
- [21] M. J. Chen, X. Wei, Y. Zheng, and S. Y. Zhou, “Intelligent question answering system based on BERT,” *Digit. Technol. Appl.*, vol. 40, no. 1, pp. 161–163, 2022.
- [22] S. R. Liang, “Research on short text sentiment analysis based on XLNet pre-trained language model,” M.S. dissertation, Guilin University of Technology, China, 2021.
- [23] J. F. Zhang, “Design and implementation of distributed music vertical search engine based on elasticsearch,” M.S. dissertation, Jilin University, China, 2019.
- [24] J. Chang, “Research and implementation of recommendation algorithm based on apache mahout,” M.S. dissertation, University of Electronic Science and Technology, China, 2013.