



Movie Recommender Chatbot Based on Kansei Engineering

P.-C. Lin^{1*}, C.-A., Lai¹, Y.-H. Lin¹, M. Haris Haikal Othman²

¹Department of Information Engineering and Computer Science,
Feng Chia University, No. 100, Wenhwa Rd., Taichung, TAIWAN

²Fakulti Sains Komputer Dan Teknologi Maklumat,
Universiti Tun Hussein Onn Malaysia (UTHM), 86400 Parit Raja, Batu Pahat, Johor, MALAYSIA

*Corresponding Author

DOI: <https://doi.org/10.30880/jscdm.2020.01.01.005>

Received 20th December 2020; Accepted 10th February 2020; Available online 1st March 2020

Abstract This paper proposes a recommender system based on Kansei model to solve the trivial things around life which gives support and reference for people who can't decide things. The Kansei algorithm design is introduced in the developed system. Taking the recommended movie as an example, the user's preference is guessed by analyzing the relevant information of the movie and the relevant information of the user, thereby catering to the user's preference and achieving the recommendation purpose. A chat robot interface, cross-platform synchronization information, and easy-to-follow text communication are the main features of the system which allows people to communicate conveniently with the robot and assist them to solve the problem or decide a thing.

Keywords: Recommender System, Chatbot, Kansei Engineering

1. Introduction

The "machine" is regarded as a new star product that can replace human labor after the industrial revolution. The purpose is to replace the repetitive labor work and widely used in the textile or heavy industry [1-2]. After the information revolution, "robots" were born. Using computer program control, it has a simple thinking ability like human beings, and further replaces the work that requires a small amount of judgment [3]. For example, peanuts in canned milk peanuts are selected by robot detection. Such capability in robot is important as they start to develop decision making ability to finally assist human operation at some place.

With the rapid development of communication software, there is also a dialogue between the robot and the user on the communication software, that is, "chat robot". Simple operation or question and answer is applied in the human-robot interaction. 24 hours online customer service in the e-commerce system can be made by the assistance of the robot. The robot also can handle large and repetitive trivial problems in receiving delivery orders from fast food or beverage stores. It does greatly reduce the cost of hiring customer service personnel and making manpower arrangements more rational. The background of the chat robot is also program-controlled which makes the user able to communicate with the chat robot [4 - 5]. In traditional human-robot communication, the chat robot will only reply to the fixed text. Inputting the incorrect text will only get the error message, and the message fault tolerance rate is almost zero. However, in some situation, the user may not feel the feeling of chatting and quit using the chatbot after multiple messages. This means that the existing chatbot is lacking

*Corresponding author: peichunpclin@gmail.com

2020 UTHM Publisher. All rights reserved.

penerbit.uthm.edu.my/ojs/index.php/jscdm

with reasoning algorithm which enables the chatbot to make judgment or reasoning while communicating with the human. It makes other algorithms are needed to assist the chatbot in judging the user's answer.

At present, the application of the artificial intelligence approach has achieved success in many fields [6]. Chat robots have also been introduced with artificial intelligence technology through hybridization methods with machine learning with smart sensing technology, and so on. This is to improve the ability of chat robots who no longer only have a cool response. It also contributes to chat robots being a hot topic in key industries. Its application includes medical drugs, advertising, decision-making, e-commerce, financial transactions and more.

Today's chat bots can process data, but they can't load the data types derived from such huge data. Because large data is scientific in its view. Big data draws conclusions that are possible answers, not scientific ones as the correct answer. However, fuzzy statistics unexpectedly solve the problem that traditional precise logic theory cannot solve, or it is difficult to express conclusions. The challenge is to integrate and analyze scattered data is the application of fuzzy statistical techniques to big data. The design of the sensibility engineering we will use depends more on fuzzy statistical techniques for analytical research. This research topic will be based on the experimental design method of perceptual engineering. First, we will design the algorithm of perceptual engineering. Second, we bring this algorithm into the application of chatbot. Later, an independent database of individual users is established based on perceptual engineering. Finally, the user's needs and ideas are inferred from the results of the data analysis, and then a recommendation system for the user is made.

Due to the development of the Internet, people's dependence on search engines has increased. It is often seen on social platforms. Many users have entered keywords that are not accurate enough so that the search results are not as expected. Or the results of the search are not in line with user preferences. For example, a person who does not like pasta, searches for the food around Fengjia on the search engine. The search results are Tiger Ramen, Yilan Ramen, etc., which does not match the original user requirements. Therefore, in order to avoid such situation, and to create a system that can correctly cater to the user's preferences, the sensible engineering is introduced. This method use the user's data in various operations to calculate and guess the user's preferences. The preference is then recorded as reference and further activity such as calculates the user's interests and so on, in order to deliver the correct information and achieve the best results.

In the literature, Inagawa et al. [7] applied the Interactive Immune Algorithm (IIA) to search the database for relevant data about user preferences. Based on the data, a predictive judgment is made using the Markov model. The best answer is then returned to the user after the multi-party comparison. Fig.1 shows the flow of this algorithm. Hence, in this study, we apply the IIA method to the chat statement analysis of the chat robot. We will analyze the implementation of the chat robot. First, we will collect the user's preference data in the conversation between the user and the chatbot, or in the user's inquiry to the chat bot. Secondly, all collected data are analyzed by the sensible engineering system and stored in the user database. Finally, based on the information to predict the user's preferences, select the most likely answer, and then make an effective response system design.

This paper is arranged into five sections. Section 1 introduces the research and followed by literature review study in the Section 2. Section 3 describes the architecture of the movie recommender system. Section 4 demonstrates the result while Section 5 concludes the discussion.

2. Literature Review

2.1. Interactive Immunization Algorithm (IIA)

When the body resists foreign microorganisms, it generates a self-protection mechanism to prevent the damage caused by microorganisms. These microorganisms are called antigens. When the antigen invades the body, the immune system immediately develops an immune response, producing antibodies to bind to the antigen to achieve the effect of inhibiting the antigen. By this concept, de Castro [8] proposed an immune system-based architecture to solve complex evolutionary operations, namely the Immune Algorithm (IA). Inagawa and Hakamata [7] mentioned that the sensible engineering system uses IIA to learn user preferences [9]. The use of immune algorithms (IA) is illustrated by [7] to calculate fitness using the evaluation function (antibody-adapted antigen) to the extent that the IIA uses the user's own assessment, and not the evaluation function. The process of IIA evolution begins with a population of randomly generated antibodies. In the process of evolution, the fitness of each antibody is assessed by the user, and the evolution of this antibody is accomplished if the antibody is the optimal solution. Conversely, low fitness continues to evolve to produce the next generation, constantly mutating until the best solution is produced.

2.2. Decision of Kansei Field

In a perceptual engineering system, the decision in the perceptual field is to identify the stage in which a particular field is being investigated using a perceptual engineering methodology. Because the Kansei experience is unique with different products, a specific concentration area is necessary in the process of perceptual engineering research. Domain decisions can be segmented through the use of market analysis techniques or through targeted consumers [10]. In addition, the decision is necessary to deal with the Kansei engineering based on existing products or to start designing a new concept of a product from scratch. Different methods can be applied, and none can be classified as better than the other. As a method of experience alone, the field can be determined to improve the existing situation based on the existence of flexibility.

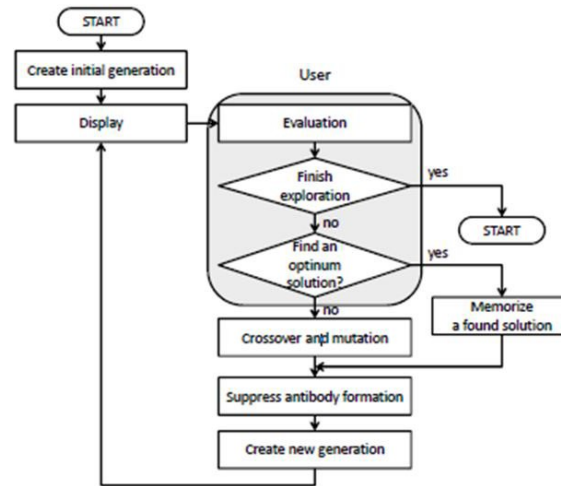


Fig. 1 - Algorithm of IIA [7]

(i) Identification of Kansei

First of all, Kansei's expression pattern is in the form of adjectives or nouns, known as Kansei words, which is a necessary condition. Usually the number of Kansei words originally prepared will be very large, and this reduction can be performed qualitatively or quantitatively.

(ii) Measurement by Kansei

Kansei's measurement is the process of capturing the consumer's Kansei. Because Kansei is subjective, vague and informally organized, it is impossible to measure it directly. Therefore, we need to come up with an indirect measurement method to use the alternative approach [11] Kansei measures the measures and psychological measures that are classified into physiology.

Physical measures aim to capture consumer behavior, response and physical expression patterns. This can be done by electroencephalography (EEG) analysis of EEG, Electromyography (EMG) muscle load measurement, used to measure Kansei's eye movements and other physiological ergonomic indicators, when a consumer uses or when looking at this product. Examples of such measurements can be used to find the degree of evoked in their hearts [12], and the response to robot feedback [13].

Psychological measures are coordinated with human mental states (such as consumer behavior, actions, and impressions). This can be done using a self-reporting system, such as different emotional scales (Desa) measurements, different scales of semantics or freely labeled systems. Such measures are popular in the implementation of its simple sensible engineering.

(iii) Analysis of Kansei

By the results of the first two identifications and measurements, we can calculate a relationship coefficient between each individual and other individual, through which the recommendation system can be operated. We usually use a factor analysis (Factor Analysis, abbreviated as FA) for inductive analysis. FA is a statistical data reduction technique that used to be used to observe correlation variability in random variables or to observe small changes in random variables relative to those. FA can usually reduce the data level of different attributes to some important degree. This reduction is necessary because the level of any attribute is affected by the results of other attributes. The FA is often used to explore the Kansei concept in the field of investigation and the psychological structure of Kansei. Such a result usually reinforces Kansei, which describes consumers in a field, and then decides on the new concept of Kansei products.

2.3. Recommender system

The recommendation system is a system for filtering information to predict the user's rating or preference for the item. There are two ways to generate a list of recommendations, content-based recommendations and collaborative filtering. Content-based recommendations use the characteristics of items to recommend items with similar characteristics, while collaborative filtering uses the past behavior of users to guess future behaviors. These two methods are often combined to achieve the purpose of improving recommendation accuracy[12].

With the rapid development of the Internet, people rely on the Internet to obtain all kinds of information, and then the amount of information is gradually increasing, and it has reached the level of information overloading, making it more and more difficult for users to get real on the Internet. The information you need. The recommendation system came into being. The filtering or feature retrieval method was used to retrieve a user's real information from a large amount of data, so that the user can easily obtain the required information[13].

The obvious problem in the movie recommendation system is that when a new user enters the system, the recommendation system is directly invalid because the system does not know the user's preference for the type of the movie. In order to solve this contradiction point, the movie will be randomly recommended to the user at the beginning, and then the user will be recommended according to the user's feedback on the movie, analyzing the feature points and the user's preference. It is obviously insufficient to randomly recommend a one-step movie. Usually, five to ten movies are recommended in succession, and a preliminary feature point is generated to further recommend the movie to the user. However, the previously randomly recommended movies are often not accepted by the user, which makes the feature points in the movie too long, so that the recommendation system cannot correctly recommend the user's favorite movies to the user. The same situation occurs in new movies. When a new movie is stored in the system, since the new movie has not been scored or referenced by other users, the system cannot produce valid recommendations, and most of them can only be made in a random manner. This movie can be used on the table [14].

2.4. Chatbot System

With the rapid development of communication software, there is also a dialogue between the robot and the user on the communication software, that is, "chat robot", which can be applied to simple operation or question and answer. With the e-commerce system, 24 hours online customer service can be made. Handling large and repetitive trivial problems can also be used to receive delivery orders from fast food or beverage stores, greatly reducing the cost of hiring customer service personnel and making manpower arrangements more rational. The background of the chat robot is also program-controlled, causing the user to communicate with the chat robot, as in the next command, the chat robot will only reply to the fixed text, and inputting the incorrect text will only get the error message, and the message fault tolerance rate is almost Zero, the user may not feel the feeling of chatting after a number of messages and give up using the chatbot, which means that other algorithms are needed to assist the chatbot in judging the user's answer.

3. Architecture of Movie Recommender System

Fig. 2 shows the Architecture of Movie Recommender System. Section 3.1 explains the decision-making process for the system.

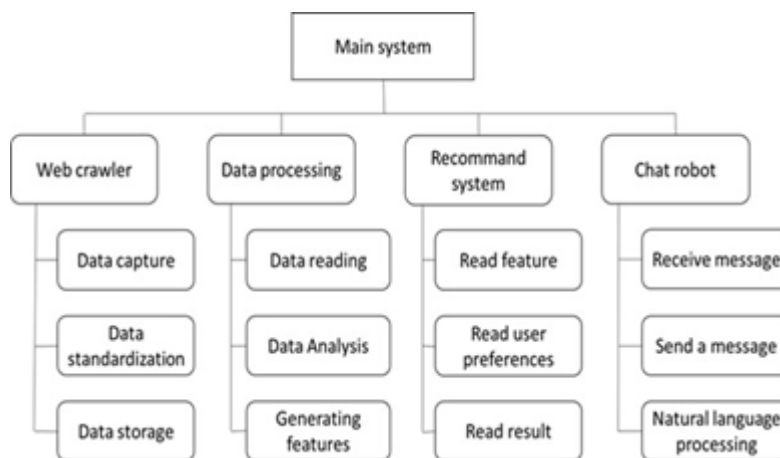


Fig. 2 - Movie Recommender System Architecture

3.1. Decision Making Process for Recommender System

We give expressions of the decision-making process for the recommender system in the following.

a) Web Crawlers

The recommendation system is a Chinese-based recommendation system. The major movie websites in China is examined, and to compare the information richness of each website, such as Chinese and English titles, director name, cast, release date, length, and foreign ratings. The Yahoo Movies page is used as a reference in this study due to the information needed. Secondly, the source code of the site is neat and formatted, which helps us to crawl the data using the automated program.

But the shortcomings are also obvious. Yahoo movies are not professional movie database websites. The movies displayed in the category are only a few recent ones. A crawler can climb up to two hundred. We started crawling this website every day from mid-September, filtering out about a thousand non-repeating Chinese film materials. For a recommendation system, such data volume can be said to be very rare, so we have targeted movies that have recently been released. That is, the current cinema film is recommended, and can be further recommended by combining relevant cinema materials. Fig. 3 Shows the Yahoo movie and its original code structure.

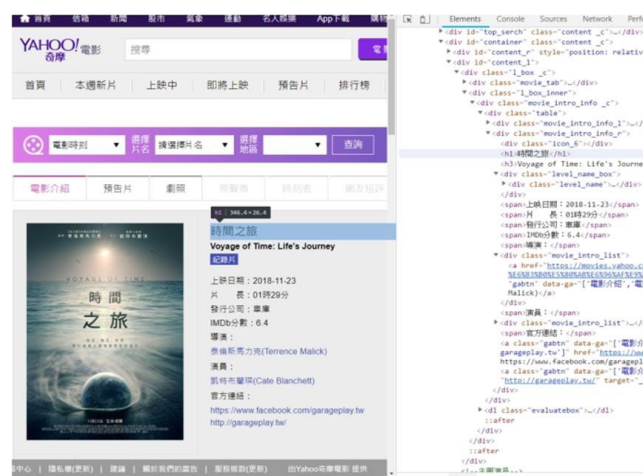


Fig. 3 - Yahoo movie and its original code structure

As mentioned in the previous section, the Yahoo movie's web page source structure is relatively simple, which is more conducive to the use of automated programs to crawl, because Yahoo does not provide the relevant Web API and other functions, so the use of such soil-making steel Ways to conduct data exploration. Fig. 4 shows part of the code of the crawler we wrote, using the request provided by the Python third-party library to send request to the website, get a response, and then put the resulting HTML into a BeautifulSoup that is also provided by a third party. BeautifulSoup provides a sophisticated analysis of the tag language HTML or XML, which is simple and quick to use. We use the methods provided to analyze the source code of the web page requested by the request, and then use the find all method provided by it to filter out the data we need. The code in the Fig.4 is the parsing of the movie page link obtained from the search result page.

```
def url_parser():
    parser_list = []
    last_found = ''
    for cate in ['intheaters', 'comingsoon']:
        parser_url = 'https://movies.yahoo.com.tw/movie_' + cate + '.html?page='
        for page in range(0, 10):
            print(c_time() + ' Now on ' + cate + ' page {}'.format(str(page + 1)))
            parser_rq = requests.get(parser_url + str(page + 1))
            parser_bs = BeautifulSoup(parser_rq.text, 'html.parser')
            for i in parser_bs.find_all('div', {'class': 'release_movie_name'}):
                for j in i.find_all('a', href=True):
                    if j['href'] != last_found:
                        parser_list.append(j['href'])
                        last_found = j['href']
            time.sleep(0.5)
    return parser_list
```

Fig. 4 - Crawler Code (Partial)

After the webpage is crawled, the BeautifulSoup analysis is used again to get several detailed parameters of the movie. With these parameters, some feature values can be generated, and the user can be recommended according to the feature values

We standardized the analyzed data into a format that we can easily read. Considering the convenience of data reading, we chose the lightest JSON as our data storage format. Fig. 5 below is a format reference for the actual storage of movie related material.

```

"師聲對決": {
  "chinese_title": "師聲對決",
  "english_title": "Le brio",
  "picture_link": "https://movies.yahoo.com.tw/i/o/production/movies/Octob",
  "category": [
    "劇情"
  ],
  "release_date": "2018-11-16",
  "length": "01時35分",
  "publisher": "鴻聯國際開發股份有限公司",
  "imdb": "6.9",
  "director": "伊凡艾塔(Yvan Attal)",
  "actor": [
    "丹尼爾奧圖(Daniel Auteuil)",
    "卡麥莉亞裘丹娜(Camélia Jordana)",
    "亞辛維夏(Yasin Houicha)"
  ]
},
    
```

Fig. 5 - Standardized data

b) Data Processing

Continuing Web crawler results, the HTML data is sorted into a pre-defined format. Some information is analyzed based on this data to assist judgment of the system in the future.

Few special forms is defined: movies in the last month, movies in the last three months, movies in the last six months, movies in the past year, movies of a specific type, movies developed by specific companies, A specific director's movie, a movie of a particular actor, and a movie matching the definition of these forms is filled into the form, so that the most relevant information can be found at a faster speed when searching for information. Take the famous American actor Morgan Ferryman as an example. We can create another film that is starring Morgan Freeman and write all the films of Morgan Freeman.

After the analysis step, a few of more specific tables are obtained, and then the other related features are generated for these tables. In the same way as Morgan Freeman, in the "Morgan Freeman starring movie" form, we found that most of the movies are comedy or action movies. We can mark Morgan Freeman separately "comedy" and "action" with feature tag. On the other hand, the feature tags are used when a new user first uses the system. Because there is not much analysis of the user data provided by the past.

Table 1 Morgan Freeman's Feature Table (Partial)

Actor	Morgan Freeman
Feature	Comedy, action, god, old man, humor
Starred in a movie	The Shawshank Redemption. Se7en. Bruce Almighty. LUCY ...etc

c) Building Recommender System

Previously, a number of special tables and features is generated, which are actually used in this function to recommend results. By combining the user's past history, or the results of past queries, this system can infer what kind of movie the user might like, or which actor has a specific preference. The user is also label with the tag, that is, the feature. By pairing the characteristics of the data end with the characteristics of the user side, plus the conditions of the user's current inquiry, etc., we can recommend a movie that the user may like. In the case of new users entering the system, we do not have enough user preference data to generate features or labels. In this case, we can only use the conditions currently entered by the user, and then based on the popularity of the movie score or release date. The user is randomly recommended to the movie, and the user is asked whether they like the movie or like the movie, thereby generating the user's characteristics.

4. Experiment Results

As shown in Fig. 6, with Morgan Freeman as an example, there is a user who has the characteristics of "like Morgan Ferryman" and "like comedy". When the user enters "I want to see the latest movie". At that time, because there was no clear time relationship, the system went to find the latest movie starring Morgan Freeman, and found the "Just Getting Started" to meet the conditions and recommended it to the user. When the user enter "Want to Watch Comedy" again, the user will also find the same movie. But since the previous result has already been recommended, it is speculated that the user may not like

"Going in Style ", so the system will recommend another one. When the user enters "Want to see Morgan Freeman's movie," the comedy is selected from all of Morgan Freeman's movies based on the user's characteristics. Then based on the film's IMDb score or within the system. Recommended times to recommend movies to users, here is recommended "Bruce Almighty."

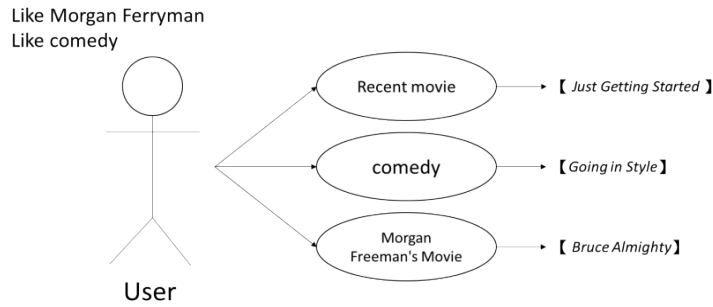


Fig. 6 - Result of Recommender System for Morgan Freeman

4.1 Chat Robot

We illustrate the processes of building chatbot in this subsection.

```
@app.route('/hook', methods=['POST'])
def webhook_handler():
    """Set route /hook with POST method will trigger this method."""
    if request.method == "POST":
        update = telegram.Update.de_json(request.get_json(force=True), bot)

        # Update dispatcher process that handler to process this message
        dispatcher.process_update(update)
    return 'ok'

def reply_handler(bot, update):
    ans = pr.process(update.message) # 將訊息直接拋入處理程序中
    update.message.reply_text(ans)
```

Fig. 7 - Processing of the Telegram Message API

4.2.1 Receiving and Sending Messages

For our implementation of the recommendation system, the chat robot interface only needs to provide the receiving and sending of messages. Fig. 7 shows the basic messaging function. After the reply handler receives the incoming message from the user, it will trigger this function as update. Telegram wraps the update in JSON format, so the update contains all the information related to this message, but we don't need that much, we only need user_id, message_time, message_type, etc. in update.message, update.message. Dropped into the other function we wrote, handed over to the recommendation system of 4.5, and then the returned result is returned through the update of the provided return message method to achieve the ultimate goal of communication with the user.

4.2.2 Natural language processing

Natural Language Processing (NLP) is a problem that human-to-machine communication will face. Machines are different from humans in understanding multi-level enrichment and methods. Machines can only understand rigid machine language, and most of these languages It is imperative and very unintuitive to human beings. It can be said that it is harder to understand for human beings. Therefore, trying to make the machine understand the human language is bound to be the future trend. Compared with English, it is more difficult to implement NLP. Because there is no gap between words and words in Chinese sentences, it is necessary to think about the problem of Chinese word breaking in a set of methods. In our topic, the solution is Python's third-party suite, Jieba, which is designed for Chinese word- breaking, with fast and lightweight features, and is compatible with the Jieba-specific entries developed by Academia Sinica for Chinese. It has been possible to accurately segment 90% of the sentences, and the less than 10% can be expanded into the package in the form of custom terms, so that the accuracy rate will rise again. Another problem with Chinese word-breaking is that Chinese habitually omits pronouns, although English Pronouns are also omitted, but basic pronouns can be omitted from the context to understand the original statement. However, Chinese is not the case. Chinese is similar or even the same statement because of the sentence or province. The position of the syllabic words is different, which leads to a completely different semantic situation. E.g:

Throw the bones to the puppy because (dog) is hungry, or
 Throw the bones to the puppy because the bones are useless.

Two similar statements are given above, omitting two different pronouns. These two pronouns use it as a pronoun in English, but there are differences between "it" and "it" in Chinese. The program is difficult to discriminate.

In our topic, in order to avoid the above problems, several groups of keywords is defined in advance that will appear when communicating with the system. After using Jieba Chinese word breaks, look for keywords in the word break to achieve the effect of judgment and meaning. Although the accuracy of this method is quite low, the application can handle most situations. Because there are no other conditions, in the case of directly communicating with the system "I want to watch a movie!", the system will only randomly recommend movies according to the characteristics of the users only. If the users have no features, they are all randomly recommended.



Fig. 8 - (left) Entering the continuous result of "I want to watch a movie" and (right) Enter the continuous result of "I like to watch Morgan Ferryman"

User can also enter their own preferences directly, for example entering a movie like ‘Morgan Freeman’ and directly asking for the results of a particular type of video does not always ask for preferences.

Directly ask the robot to recommend a recent good-looking movie, and the robot will reply in batches with reference to the leaderboard. Following the result from Fig.9 (Right), the direct input of the number is equivalent to the full movie name, and the robot will provide the cinema reservation information. The result for multiple conditions is shown in Fig. 10 (Right). The error message is returned for the result of not finding the relevant condition.

Directly ask the robot to recommend a recent good movie, and the robot will reply in batches with reference to the leader board. Following the result from Fig. 9 (Right) of previous picture, the direct input of the number is equivalent to the full movie name and the robot will provide the cinema reservation information.



Fig. 9 - (Left) Enter the result of "Is there a good movie recently?" and (Right) Enter the result of "I want to see a love movie"



Fig. 10 - (Left) Direct input result number and (Right) Response results of multiple conditions



Fig. 11 - (Left) Error results for condition not found and (Right) Unable to determine the wrong result of semantics

The result for multiple conditions shown in Fig.10 (Right). The error message is returned for the result of not finding the relevant condition. Fig.11 (Right) show the situation when the chatbot is unable to judge the wrong result of semantics.

5. Conclusions and Future Works

The potential of the recommendation system is endless. A prototype Chatbot of movie recommender system in this paper is presented with some explanation on its implementation. We hope to deepen the extension in the future and promote it into a recommendation system for music and even online programs. In the function of the Telegram, API is much more than that we only use the basic text transcribing function. In the future, we may be able to import the positioning function, directly open the navigation to the nearest studio, or directly connect with the cinema booking system.

Acknowledgement

The authors express their appreciation to the Feng Chia University. This research work also supported by Ministry of Education, R.O.C., under the grants of TEEP@AsiaPlus.

References

- [1] Brynjolfsson, E., & McAfee, A. (2012). Race against the machine: How the digital revolution is accelerating innovation, driving productivity, and irreversibly transforming employment and the economy. Brynjolfsson and McAfee.
- [2] Vysocky, A., & Novak, P. (2016). Human-Robot collaboration in industry. MM Science Journal, 9(2), 903-906.
- [3] McBride, N., & Hoffman, R. R. (2016). Bridging the ethical gap: from human principles to robot instructions. IEEE Intelligent Systems, 31(5), 76-82.
- [4] Oh, J., Howard, T. M., Walter, M. R., Barber, D., Zhu, M., Park, S., ... & Romero, O. (2016, October). Integrated intelligence for human-robot teams. In International Symposium on Experimental Robotics (pp. 309-322). Springer, Cham.
- [5] Schiffer, S. (2016). Integrating qualitative reasoning and human-robot interaction in domestic service robotics. KI-Künstliche Intelligenz, 30(3-4), 257-265.
- [6] Sheridan, T. B. (2016). Human-robot interaction: status and challenges. Human factors, 58(4), 525-532.

- [7] Yuma Inagawa and Junki Hakamata, "A framework of recommender system considering the variety of Kansei", SCIS-ISIS 2012, Kobe, Japan, November 20-24, 2012.
- [8] L. N. de Castro and J. Timmis, "Artificial Immune Systems: A New Computational Intelligence Approach." London, U.K.: Springer-Verlag, 1996.
- [9] H.Nishino, T.Sueyoshi, T.Kagawa, K.Utsumiya, "An Interactive Graphics Rendering Optimizer Based on Immune Algorithm," 2007 EvoWorkshops, pp.459-469, 2007.
- [10] Anitawati Mohd Lokman, "Design & Emotion: The Kansei Engineering Methodology", Vol.1, Issue 1, Malaysia, 2010.
- [11] Ishihara, I., Nishino, T., Matsubara, Y., Tsuchiya, T., Kanda, F., Inoue, K. (2005). Kansei And Product Development (In Japanese), Ed. M. Nagamachi. Vol. 1. Tokyo: Kaibundo.
- [12] Nishikawa, K., Hirakawa, Y., Nagamachi, M. (1997). An impact of heating environment influencing heart rate. Jap. J. of Ergonomics, 33 (2) 105-112.
- [13] Ishihara, K., Harada, M., Ishihara, S. (2007b). Kansei to Movement of Autonomous Robots. The 10th International Conference on Quality Management and Operation Development (QMOD '07), Sweden: Linköping University Electronic Press, ISSN 1650-3740.