Enhance Human-Computer Interaction Using External Robot Agent in Self-healthcare System

Yihua Zhang
44171022-4

Master of Engineering

Supervisor: Prof. Jiro TANAKA
The Graduate School of Information, Production and Systems
Waseda University

February 2019

Abstract

When we feel sick, we have two choices: see a doctor or do self-healthcare. In the case of seeing a doctor, doctor plays a key role in helping patient access the medical information. However, in self-healthcare system using PC, it is often difficult to find the essential medical information. There are self-health care systems with internal agent to help user access medical information smoothly. User interacts with internal agent using voice, and agent controls the computer to show information to user. But it still has disadvantages: Firstly, the whole process is not so attractive. Secondly, operating internal agent may be inconvenient for novice computer user. Thirdly, the way of interaction is limited.

The goal of our research is to explore the feasibility of using an external robot agent to build a robot-human-computer relation in self-healthcare system.

The system includes two parts. One is Providing Health Information. Robot agent collects user's health condition by asking question based on user's symptoms and help user access health-care information on computer. The other part is Showing Medicine Instruction. System can give instruction of a medicine by recognizing medicine package shown by user.

We have used an external agent to build a robot-human-computer relation, which is similar to a doctor-patient-computer relation in the real world, to make the self-healthcare system more scalable and convenient. Besides, we have integrated motion interaction, voice action, and image interaction to make the whole process more interactive and interesting.

Keywords: health care, conversational search, intelligent assistants

Acknowledgements

I would like to express my sincere gratitude to Professor Jiro TANAKA. In the process of completing the research and writing the thesis, he gave me great advice and help. During the two years of study in Waseda University, Professor Jiro TANAKA corrected my academic attitude and guided me on how to conduct research. Professor Jiro TANAKA's rigorous academic attitude and patience with students will have a positive impact on my future life. In addition, I would like to thank the IPLab members for their advice and help in my research and experiments.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

With the aging of the world's population and the shortage of medical resources, self-health care has become a hot area coping with rising medical costs. Traditional self-health care includes getting information on health and human body systems, advice from clinical social workers, and self-monitoring devices and assistive technology. In the recently years, online self-health care has appeared as a new method of self-health care. People who want to receive support and information from Internet use computer or smart device to get information in order to improve their health condition.

In the modern hospital health care, doctors use computer in order to provide more efficient health care for their patients. When patient sees a doctor, the doctor asks patient for his symptom and operates the computer to get medical information. Then, doctor tells diagnosis and medical advice to patient. Besides, nurse tells patient the instruction of medicine when patient shows the package of medicine.

So in the real world, we find that doctor or nurse plays a key role in helping patient to access the medical information. The patient talks with doctor, doctor or nurse understands user's health condition and use computer to access the medical information for user, as shown in Fig 1.1.

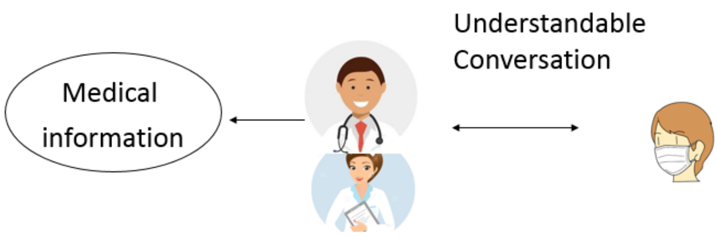

Fig. 1.1 Patient-doctor-computer relation in real world

In a Self-health care system using PC, user operates the PC and PC shows health care information to user, as shown in Fig 1.2.

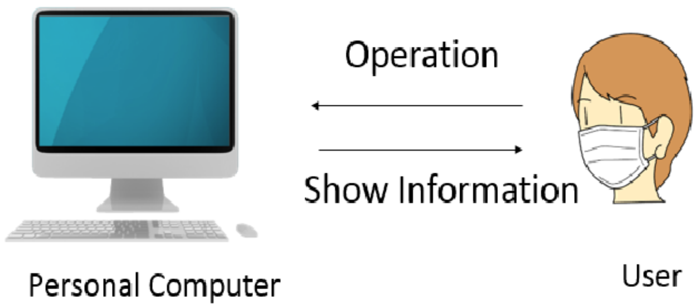

Fig. 1.2 Self-health care system using PC

However, for users who are not familiar with computers such as the senior people or some disabled people, it is often difficult to find the essential health-care information. These users prefer to talk to people to get information than to use keyboard and mouse to do input. When they search information on PC, they are not motivated to use PC.

There have been self-health care systems with internal agent to help user access medical information. In a self-healthcare system with internal agent like Siri, user talks with internal agent and agent searches for the related self-health care information from Internet. Then, it will control computer to show health care information to user, as shown in Fig 1.3.

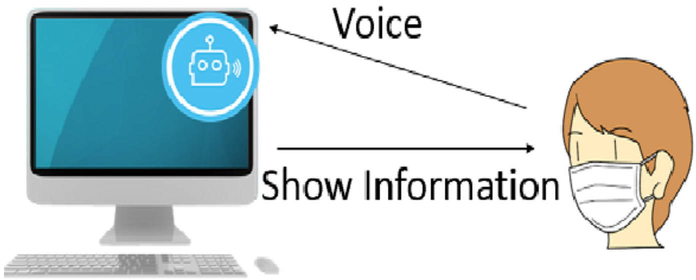

Fig. 1.3 Self-healthcare system with internal agent

However, this kind of system still has disadvantages. Firstly, although the traditional internal agent can speak out, it only converts the text information into voice. When the user using the computer, he still has the feeling that he is talking to the computer, and the whole process will be not so attractive. Secondly, for users who are not familiar with computers, in many cases, operating internal agent itself needs to have some knowledge of computer, and they may be not motivated to use PC. Thirdly, for systems with internal agents, the interaction between the user and the agent is done on the computer screen. That makes the way of interaction be limited.

Therefore, we use an external agent to make health-care system more attractive and enhance the user experience. The external robot agent is humanized and like a doctor who gives user health-care information. Agent can do motion performance when speaking out just like a human does and support diverse ways of interaction.

Chapter 2

Research Goal and Approach

2.1 Research Goal

The goal of our research is to use an external robot agent to build a robot-human-computer relation in self-health care system, as shown in Fig 2.1. Through this triangle relation, we can make robot agent like a doctor who gives user medical information. It can show and explain the health-care information on the computer to user. At the same time, it can tell the computer what does user want computer to do and what kind of health-care information is needed according to the interaction with user. So that, we can make the process of searching health-care information more interactive and simple.

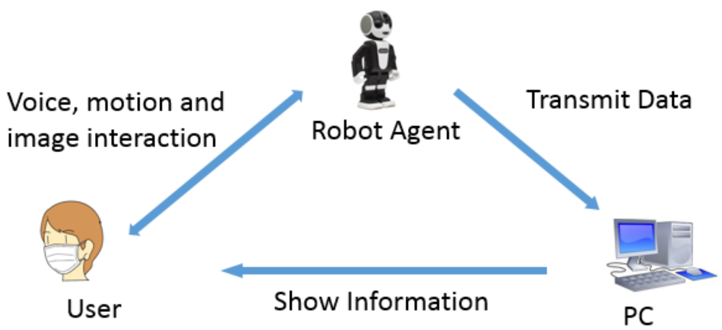

Fig. 2.1 Triangle relation of robot-human-computer

2.2 Research Approach

We formalize self-health care system as the triangle robot-human-computer relation using external agent. This relationship is similar to a doctor-patient-computer relation in the real world. In this triangle relation, user interacts with robot agent through voice, motion and image interaction. For the voice interaction, agent asks question to user and user answers question. For the voice interaction, user shows real object to robot agent. For the motion interaction, agent can do action while speaking out. After that, robot agent transmits user's data to computer, and computer will show health-care information to user.

Prevention and treatment play an important role in self-healthcare. When we feel uncomfortable, we often do some self-regulation to preventing health condition from getting worse. And if our health condition is getting worse, we will take some medicine. However, these two parts usually require people to know some professional knowledge or take much effort to search it on Internet. Thus, we propose two functions for our system, involving the two parts of self-health care: prevention and treatment. For prevention, we propose function of Providing Health Information. User can get health information to do self-regulation to improve health condition. And for treatment, we propose function of Showing Medicine Instruction. User can get instructions of the medicine when he feels ill and needs to take medicine. In Providing Health Information, robot agent collects user's health condition by asking question based on user's symptoms and help user to access health-care information on computer. In Showing Medicine Instruction, agent shows instruction of a medicine by recognizing medicine package shown by user. Both of which are typical cases of health-care and are closely related to human-computer interaction. To realize them, we introduce several kinds of interactions among user, robot and computer:

1. User inputs his situation by making conversation with robot agent or showing real objects.

2. Robot agent interacts with users using motion performance to make the process of acquiring user information more relaxed and fun.

3. Robot agent gets health-care information from Internet for user, including medicine instruction, article, video, and social networking service.

4. User selects and decides the information they want to see via voice control.

2.3 Advantages compared to system with internal agent

Compared to health-care system with internal agent, health-care system with external agent has the following advantages.

Firstly, we have introduced motion interactions. Compared to internal agent, not only can external robot agent support voice interaction, but also it can do motion performance when speaking out. Although the traditional internal agent can speak out, it only converts the text information into voice. When the user using the computer, he still has the feeling that he is talking to the computer, and the whole process will be not so attractive. In comparison, the robot can make an action based on the content of the speech. For example, robot nods when it says OK, and shakes its head when it says, "I can’t hear what you are saying." Moreover, it can bow like a human when it let user to wait. Thus, when the user communicates with the robot, he can get an experience like talking to people. Such a robot agent is more humanized, attractive and can enhance the user experience.

Secondly, system with external agent can be used for users who are not familiar with computers, such as the senior people or some disabled people. These users prefer to talk to people to get information than to use keyboard and mouse to do input. When they search information on PC, they are not motivated to use PC. Although we have system with internal agent, in many cases, operating internal agent itself needs to have some knowledge of computer. Therefore, we let the user do the input through an external robot: The robot asks the user question, then the user answer the question. After that, robot sends them to PC and PC displays information to user. In this case, user does not need to operate the computer when he inputs his situation. Through this system, we make it easy for users who are not familiar with the computer to get information.

Thirdly, system with external agent has more diverse ways of interaction. For systems with internal agents, the interaction between the user and the agent is done on the computer screen. That makes the way of interaction be limited. But in the case of external agent, we used an individual agent to replace the built-in agent. This allows not only the spatial extent of the interaction, but also the system can support more kinds of interactions. Users can take the agent away and interact with it in a comfortable way. For example, user can lay on the bed and talk to robot agent, and robot agent controls the computer on the desk to show health-care information. Besides, user can interact with the robot using real objects by showing something. Through external robot agent, we increase the scope of interaction, so that interaction of system is more vivid.

2.4 Advantages compared to stand-alone robot

Compared to stand-alone robot, health-care system with external agent has the following advantages.

Firstly, health-care system with external agent can take the advantage of system using PC, When we use stand-alone robot to search for information, it is the robot that is responsible for searching for information. By contrast, when we use external robots to search for information, it is the computer that is responsible for searching for information. The modules of the current stand-alone robot are all customized. When we want to use it to search for medical information, we need to spend a lot of effort to build a framework for it. Compared to stand-alone robot, system with external robot agent has advantage of the flexibility and versatility of the computer to search for information on the Internet. The robot itself is only responsible for acting as a bridge between the user and the computer. The robot agent is like an extension of a computer, and it is a mixture of the human-like stand-alone robot and system using PC

Secondly, for the stand-alone robot, the agent and the computer main processor is integrated. For the triangle relation of external robot agent-computer-user, all the parts are independent. Thus, the external agent has more expandability and flexibility than stand-alone robot. If it is needed, we can easily extend interactions or improve one of them. For example, we can add sensor on the agent or on the computer to support more kinds of interaction. This makes the system more scalable. In comparison, stand-alone robot is highly integrated from the beginning, and adding interaction or improving one of them is inconvenient or even difficult to implement.

Thirdly, system with external agent has more diverse ways of interaction. When we talk to stand-alone robot, the information is shown on the screen placed on robot's body. We can find that the interaction is two directional. In our system, we formalize the system as the triangle relation of robot-computer-user. The agent is separated from computer, which allows not only the spatial extent of the interaction, but also the system can support more kinds of interactions. Users can take the agent away and interact with it in a comfortable way such as searching for information in a distance. Through external robots, we can increase the scope of interaction, so that the interaction of system is more vivid.

Chapter 3

Related Work

3.1 Related Works on Hospital Health Care System

Many researchers [1][2][3][4][5] try to apply user interface using virtual agent in hospital health care.

Timothy W. Bickmore's research [2] presented a system with animated and empathetic virtual nurse interface for educating hospital patients. In this paper, they said many patients have inadequate health literacy, resulting in a reduced ability to read and follow what doctor or nurse says. They described an animated, empathic virtual nurse interface for educating and counseling hospital patients with inadequate health literacy. The development methodology, design rationale, and result of user testing are described. Results indicate that hospital patients with low health literacy found the system easy to use, reported high levels of satisfaction. Patients also expressed appreciation for the time and attention provided by the virtual nurse and felt that it provided a useful source for their medical information. They showed that the virtual agent provided can help user to get medical information.

Jonathan Gratch's research [1] show that use of virtual humans can make users feel gladder to engage. In their research, participants received a health-care interview by interacting with a virtual agent. The results of the experiment showed that when using system with virtual agent, participants felt high system usability and did not hesitated to give their information. The study showed that in the case of considering the user experience and encouraging users to provide their information actively, the virtual person can help medical workers to collect the patient's information.

Zeljko Obrenovic's research [3] presented a system built in a tele-medical environment using a mobile agent. They designed some interaction for the agent, supporting different kinds of user interface components in the tele-medical system. In this paper they presented the telemedical environment based on hardware implemented with Java mobile agent technology. The agent provides support for interaction between mobile agents and different user interface components in the telemedical system. They developed an agent framework with four types of agents: data agents, processing agents, presentation agents, and monitoring agents. Data agent abstract data from date source and create view on different types of data. Processing agents produce derived data. Presentation agents create user interface using a variety of user data views. User interface are created based on HTTP, SMS and WAP protocols. Monitoring agents monitor some possible emergency by analyzing the data like heart rate or blood pressure. They found that the flexibility of distributed agent architecture is well suited for the telemedical application domain.

In Terry Ellis's research [4], they explored the feasibility, acceptability and preliminary evidence of the effectiveness of a virtual exercise coach to promote daily walking for persons with Parkinson Disease. In this paper, they said that exercise improves function and quality of life in persons with Parkinson's Disease. They used a virtual exercise coach, an animated character viewed on a notebook computer in the subject's home that emulates face to face interactions. The effectiveness of a virtual exercise coach to improve exercise adherence in people with Parkinson Disease had never been explored before. The aim of their study was to explore the feasibility, acceptability of a virtual exercise coach. They promote daily walking for persons with Parkinson Disease over a one-month period. The result showed that sedentary persons with Parkinson Disease successfully used a computer and interacted with a virtual exercise coach. Retention, satisfaction, and adherence to daily walking were high over one-month and significant improvements were seen. They reported high satisfaction and adherence.

Nasim Motalebi's research [5] applied conversational agents in medical domains. Conversational agents (CAs) like Amazon Alexa can potentially enable a new way to deliver therapy to patients with serious mental illnesses. Specifically, they can be used to provide support for real-time family therapy and interventions in a scalable way. However, this requires significant changes in traditional treatment since interaction with CAs is fundamentally different than reading or using eHealth applications. they aimed to identify challenges in adapting a clinically treatment for Post-Traumatic Stress Disorder (PTSD) to conversational agents. Specifically, they described their initial design and development process to use Amazon Alexa to deliver special treatment for PTSD. Their initial design process resulted in an interaction model that emphasizes short dialogues and interactivity. This design process and interaction model can potentially be useful for future studies focusing on using conversational agents for the treatment

3.2 Related Works on Self Health Care System

In the self-health care area, Ari H Pollack's research [6] studied the problem we need to solve for the self-health care comparing with hospital health care. In this paper, they described that patients face many challenges when doing self-health care. They identified three key elements for this kind of challenge: knowledge, resources, and self-efficacy. They describe how patients lack the necessary medical knowledge, how patients cannot easily access the necessary medical resources, and how patients can't manage themselves effectively. In addition, they outline design opportunities and direction for systems to support patients in their self-management.

Scott Bickerer's research [7] presented an agent-based system based on an existing test website to help the user get treatment advice on the website. This paper reported a project to design an agent-based prototype based on the existing test website that can interact with user, which is similar to the relationship that exists between doctors and patients. Existing health testing website provided treatment advice to patients. Patient can describe his symptom on website, and this kind of information will be sent to health professionals. Health professionals give the treatment advice and website will send messages to user. However, this kind of system have problems getting patients to adhere to their advice, particularly over long treatment periods. In their research, they make an agent-based prototype alternative of the existing test website improve and sustain the patient's motivation and adherence to their treatment as well as to usage of the website.

Mengxuan Ma's research [8] proposed a smart application that monitors health status by combining the data collecting wearable sensor and the voice interactive devices of the Amazon Echo. The result showed that interactive voice interfaces such as the Amazon Echo provide an easy way for individuals to access a variety of data by using voice commands. They also present test results of the Amazon Echo speech recognition on different populations.

Hongwei Huo's research [9] integrated the technologies of wireless sensor networks and public communication networks to construct a healthcare system for people's self-health care. This system provided functionalities including indoor monitoring, outdoor monitoring, activity and health state decision, emergency decision and alarm. The results showed that the system performed high validity and reliability.

3.3 Related Works on Agent Searching

For the agent searching, some related researches [10] focus on future trends in search interfaces. Some related researches [11][12][13][14] focus on using agent searching to get information from Internet. Some related researches [15] [16] focus on enhance the interaction between agent and user.

Marti A. Hearst's research [10] explored the future trends in search interfaces. In their research, they described that in the future user interfaces for search will be natural dialogue-like interaction, and input will be non-textual content. User interfaces will involve support for natural human interaction, gesturing with fingers, speaking rather than typing, watching video rather than reading, and using IT socially rather than alone. This article has explored why these trends will also affect user interfaces for search, highlighting recent work reflecting these trends.

Tessa Laua's research [11] presented an intelligent conversational assistant to search the web information. They presented a system that automates web tasks on a user's behalf through an interactive conversational interface. Given a short command such as "get road conditions for my home," system synthesizes a plan to accomplish the task, executes it on the web, extracts an informative response, and returns the result to the user as a snippet of text. A novel aspect of their approach is that they use user's personal web browsing history to determine how to complete each requested task. This paper described the design and implementation of their system, along with the results of a brief user study that evaluates how likely users are to understand what the system did for them.

Bernard J. Jansen's study [12] shows that agent searching is similar to human searching with the exceptions of the duration and speed of interaction. Their research investigated how web agents search for online information. They first provided a classification for information agent using stages of information gathering, gathering approaches, and agent architecture. They then examined an implementation of one of the resulting classifications in detail, investigating how agents search for information on Web search engines, including the session, query, term, duration and frequency of interactions. The result showed that agent searching is not substantially more sophisticated than a by user. Equipped with this information, search engines developer and other Web information providers can design their Web sites to accommodate these automated information gathers.

Maxwell Harper's research [13] focus factors that affect search results in social question and answer web sites. Using searchable data from three popular QA sites, they showed that when users search for information, a conversational question such as "do you believe in evolution?" might successfully engage people in discussion but may be less useful than informational questions. They confirmed that humans can distinguish between conversational questions and other informational questions and presented evidence that conversational questions typically have much lower potential archival value than informational questions.

Chatbots and conversational assistants are becoming increasingly popular[17]. However, for information seeking scenarios, these systems still have very limited conversational abilities. Alexandra Vtyurina's research [14] tried to find out what would conversational search look like with a truly intelligent assistant. In this paper, they investigated human behavior when using conversational systems for complex information seeking tasks. They also compared participant behavior when talking to a human expert with the automatic system. They observed that people do not have biases against automatic systems and are glad to use them as long as their expectations about accuracy were met.

To make agent can help user to access the resources of the computer, Luke S. Settelmeyer's research [15] presented a virtual agent that interact with applications like a human user. This paper described a specialized software agent that exists in the environment of the user interface. Such an agent interacts with applications through the input device as same as human user. Its sensors process screen contents and generate mouse/keyboard events to interact with applications as same as user's actions. They described the architecture of their agent and its algorithms for image processing, event management, and state representation. They illustrated the use of the agent with a small feasibility study in the area of software logging. They provided a tool to access information of the mouse, keyboard, and display when interacting with computer.

Alyssa Glass's research [18] identified and discussed eight major themes that significantly impact user trust in complex systems, including: High-Level Usability of Complex Prototypes, Being Ignored, Context-Sensitive Questions, Granularity of Feedback, Transparency, Provenance, Managing Expectations, Autonomy and Verification. They studied issues governing the trust and usability of complex adaptive agents. By interviewing users of these agents, they have identified several themes that describe the willingness of users to adopt and trust these agents. Some researches[19] [4] also focus on making the system's decisions and behavior understandable by users.

3.4 Related Works on Agent Based E-commerce System

Wang's research [20] proposed an approach to use intelligent agents to enhance the user's shopping experience, making the agent more like a real guide through more human-like motions and languages. In their research, they used Robohon, a new generation of smart robots, to act as a new type of agent to achieve a better shopping experience through interaction with personal computers and human. A lot of voice and action were used as the main elements to interact with humans. This guiding system made the shopping experience more interesting and interactive.

Chapter 4

System Design

4.1 Providing Health Information

We have established a triangle relation for Providing Health Information as shown in Fig 4.1. In this triangle relation, we propose the interaction between robot and human, the interaction between robot and PC, and the interaction between human and PC. For the interaction between human and robot, we designed a dialogue-based voice interaction between user and robot. The robot asks voice questions to user, and user gives voice answers to robot. After that, robot gets the user's health condition based on these answers. Besides, robot agent can do the motion performance during the conversation. For the interaction between robot and PC, firstly, robot sends user's health condition to computer. Secondly, computer searches for health-cares information based on health condition and sends back the information of candidate web pages to robot. Thirdly, robot decides which web page to open according to the user's choice. And for the interaction between human and PC, we make computer display a user interface on the browser with several options allowing user to choose. Each option includes the information of candidate web pages.

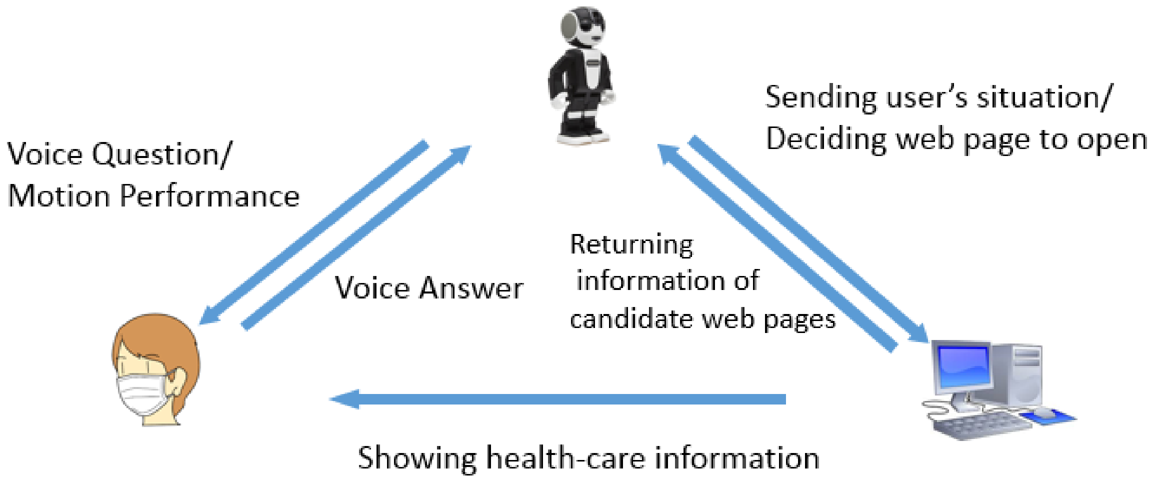

Fig. 4.1 Triangle relation in Providing Health Information function

4.1.1 Interaction between Human and Robot

For the interaction between Human and Robot, we design voice question, voice answer and motion performance for interaction between users and robots.

Voice Question

When the user tells the robot some key words like "I feel uncomfortable", robot will starts the dialogue module and asks questions according to the user's symptoms. In order to make it easy for users to answer, most of these voice questions are in the form of whether or not, or ask users where they are uncomfortable.

Voice Answer

When answering the question, user can answer yes/no, or where he feels uncomfortable. Robot Agent will recognize the content of the user's answer and gets the result by analyzing user's answers, which are used it as the input of the interaction between the robot and the PC.

Motion Performance

During the dialogue between the user and the robot, we also make the robot do motion performance according to the content of the conversation. For example, when user gives answer and robot agent confirms it, it will say "OK" and nods. When user gives answer but robot can't recognize what user is saying, he will say "Can't hear clearly" and shake its head and ask the user to say it again. When robot needs user to confirm the result, he will say "Is that OK" and open his hands like a human, as shown in Table 4.1. We designed the motion interaction between the robot and the user to make the procedure of Providing Health Information more interesting and reduce the user's resistance to self-disclosure. The motions of robot are shown as Fig 4.2

Keyword	Motion
"OK"	Nod
"Can't hear clearly"	Shake head
"Is that OK"	Open hands

Table 4.1 Motion performance when saying special sentence

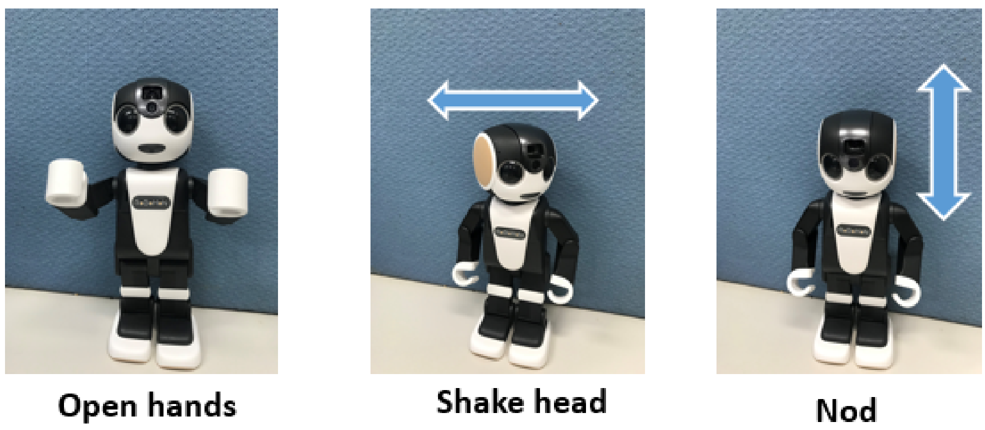

Fig. 4.2 The motions of robot for Providing Health Information

4.1.2 Interaction between PC and robot

When searching for health-care information, different users may like different types of information. Some people like reading articles, while others prefer intuitive videos. In order to enable users to choose the information they want, we designed the interaction between personal computer and robot, which allows users to freely choose information including articles, videos and social network services. The interaction between PC and robot includes sending user's situation, returning information of candidate web pages and deciding web page to open.

Sending user's situation

Traditionally, when user operates PC by himself, he needs to use the search engine to input relevant keywords. In this system, we allow robot to enter this information into the website instead of user. For the video information, robot send the user's situation and a http request for video information to the computer. Then, computer will search for video information based on user's situation. Similarly, for the article information, robot sends the user's situation and a http request for article information. And for the SNS information, robot sends the user's situation and a http request for SNS information, shown as Table 4.2.

Type of information	What the robot sends
video	user's health condition and http request for video information
article	user's health condition and http request for article information
SNS	user's health condition and http request for SNS information

Table 4.2 Robot sends requests for different types of information

Returning information of candidate web pages

After receiving the user's situation and request from robot agent, computer searches for these three kinds of health-care information from Internet. For the video information, computer uses a list of video sites and keywords to call the API and get the detailed information of the candidate video web page, including the link, thumbnail and text description of the target web page. For the article information, as the article information is widely available on the Internet compared to video, if the search range is too wide, many results may not be useful. Thus we enable the robot agent to narrow down the search range and make the results more reliable. And for the SNS information, as the type of contents of information from SNS is complicated, even we use right keywords to search for the information, many of them may be not related to health care. Thus, computer does not extract information from SNS web page, but directly open the whole SNS web page, and let user himself to choose, shown as Table 4.3

Type of information	What computer sends	Information get from Internet
video	a list of video websites	link, thumbnail and text description of candidate web pages
article	specific article websites	link, thumbnail and text description of candidate web pages
SNS	specific SNS websites	whole web pages

Table 4.3 Search range and search result for different types of information

Deciding web page to open

After getting the information of candidate web page from the Internet, PC will send the results back to the robot. At the same time, PC shows web page to display this information with three options: video, articles and SNS. Robot agent can control the computer to open the corresponding web page according to the user's choice.

To support this function, when robot receives user's voice of "watch video", it will send the user's choice for video to computer, to let computer display video information. Similarly, when robot receives the "read article" voice, it sends the user's choice. When robot receives the voice of "view sns", it will send the user's choice, situation and SNS websites, shown as Table 4.4.

Voice command	What agent sends	Response of PC
"watch video"	user's choice for video	show web page with video results
"read article"	user's choice for article and article websites	show web page with article results
"view sns"	user's choice for sns	open web page of SNS using keywords

Table 4.4 Robot agent tells PC which type of information to view based on user's voice

After that, computer displays several options allowing user to choose on the browser. Each option includes the information of candidate web pages. Robot can determines which web page to open according to the voice command from user. Besides, robot can also send return operation to the web browser, so that user can freely browse other kinds of information.

When robot determines candidate web page which web page to open, if robot receives the "open left" voice, it sends the user's selection to the computer and controls the computer to display the information of the first candidate video or article web page. Similarly, if robot receives an "open middle" voice, it will control the computer to display information for the second candidate video page. If robot receives the "open right" sound, it will control computer to display the information of the third candidate video or article page. In addition, when robot receives a "return" voice, it will control the computer to display the original web page, shown as Table 4.5.

Voice command	What agent sends	Response of PC
"open left"	request of opening first link	open the first link of web page
"open middle"	request of opening second link	open the second link of web page
"open right"	request of opening third link	open the third link of web page
"go back"	request of going back to the original web page	go back to original web page

Table 4.5 Robot agent tells PC which web page to view based on user's voice

4.1.3 Interaction between Human and PC

To improve user satisfaction and the reliability of the results, we let the computer display web page showing several options allowing user to choose. Each option includes the information of candidate result. These web pages includes main menu, information of candidate video, and information of candidate article.

User interface for main menu

After getting candidate web page from the Internet, PC will display main menu including three options: video, articles and SNS. We use a user interface to show the three options. At the top of the page is the title "Browse Categories" describing the type of information currently available, and below it is the information for each candidate web page, including thumbnails and descriptions of candidate web page, as shown in Fig 4.3. While watching these options, user can select the type of information web page he wants to view by voice control.

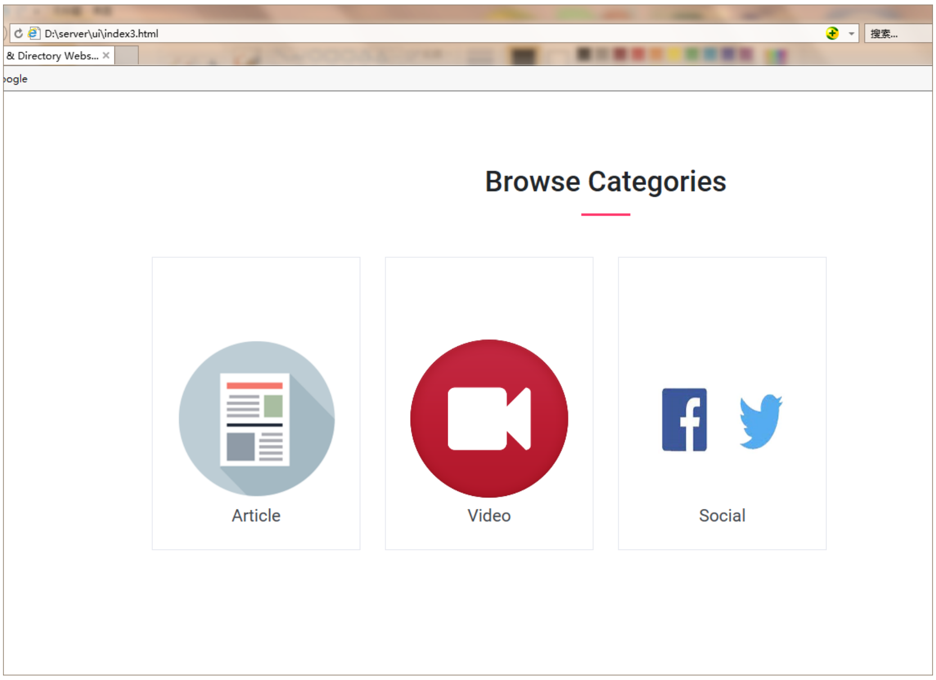

Fig. 4.3 User interface for main menu

User interface for information of candidate video

When displaying the information of candidate web pages, system extracts the elements of target page, including the description of the web page, the link, and the thumbnail. We use a simple and clear user interface to show this information, shown as Fig 4.4. At the top of the page is the title"Video" describing the type of information currently available, and below it is the information for each candidate web page, including thumbnails and descriptions of candidate web page.

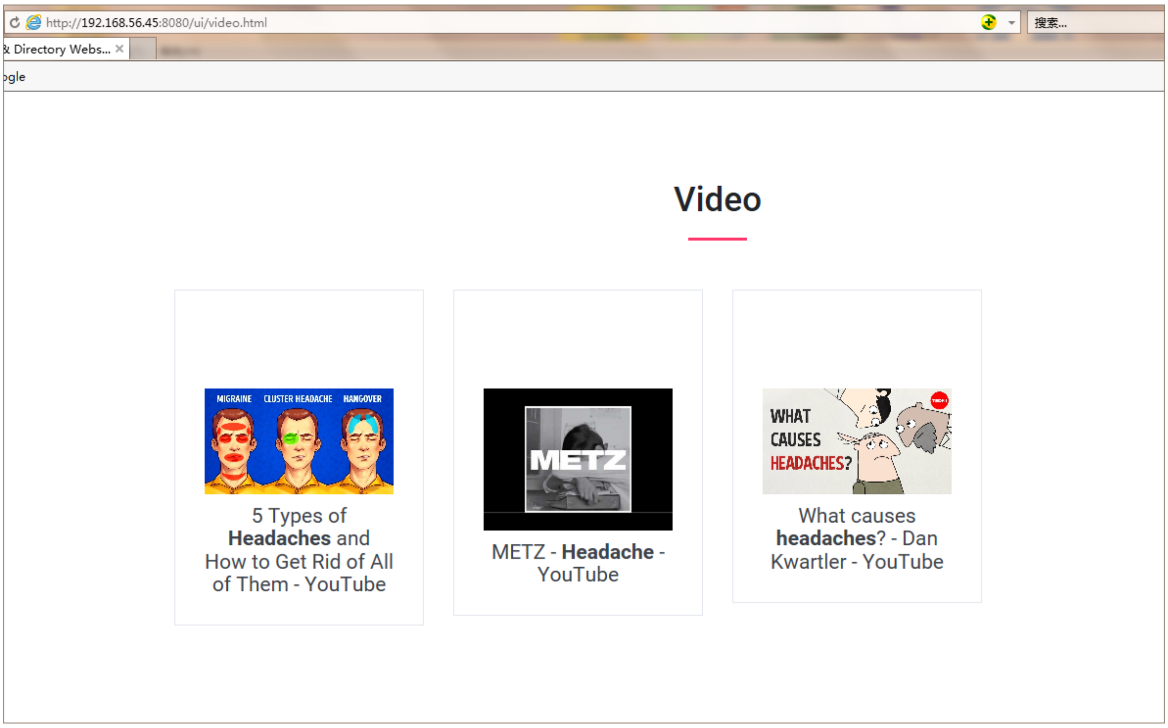

Fig. 4.4 User interface for information of candidate video for keywords"headache"

User interface for information of candidate article

We use a user interface to show candidates of article, shown as Fig 4.5. At the top of the page is the title "Article", and below it is the information for each candidate web page, including thumbnails and descriptions of candidate web page.

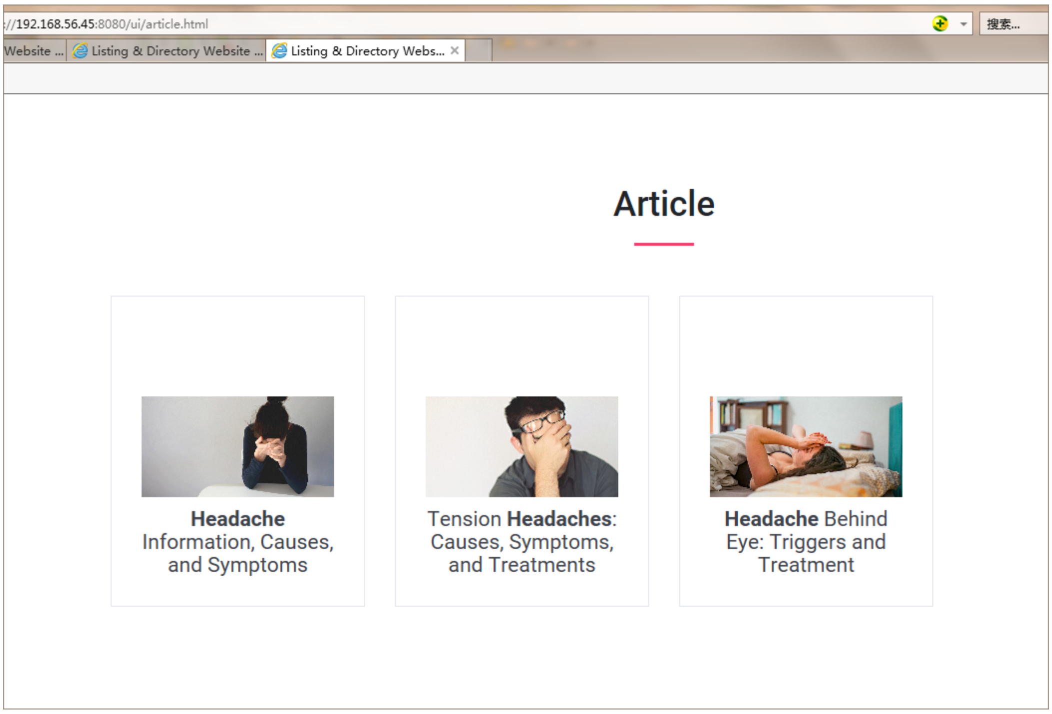

Fig. 4.5 User interface for information of candidate article for keywords"headache"

4.2 Showing Medicine Instruction

We establish a triangle relation for Showing Medicine Instruction as shown in Fig 4.6. In this triangle relation, we propose the interaction between robot and human, the interaction between robot and PC, and interaction between human and PC. For the interaction between human and robot, the user shows the package of the medicine to the robot. The robot will takes a photo of the medicine package and sends it to the PC. Robot agent can do the motion performance during this procedure. For the interaction between robot and PC, firstly, robot sends image of medicine package to computer. Secondly, computer recognizes the word information on the medicine package and return information of candidate medicine instruction web pages. Thirdly, robot decides which web page to open according to the user's choice. At the same time, for the interaction between human and PC, we make computer display a user interface on the browser with several options allowing user to choose. Each option includes the information of candidate medicine instruction web pages. User can choose the web page he wants to view by voice control.

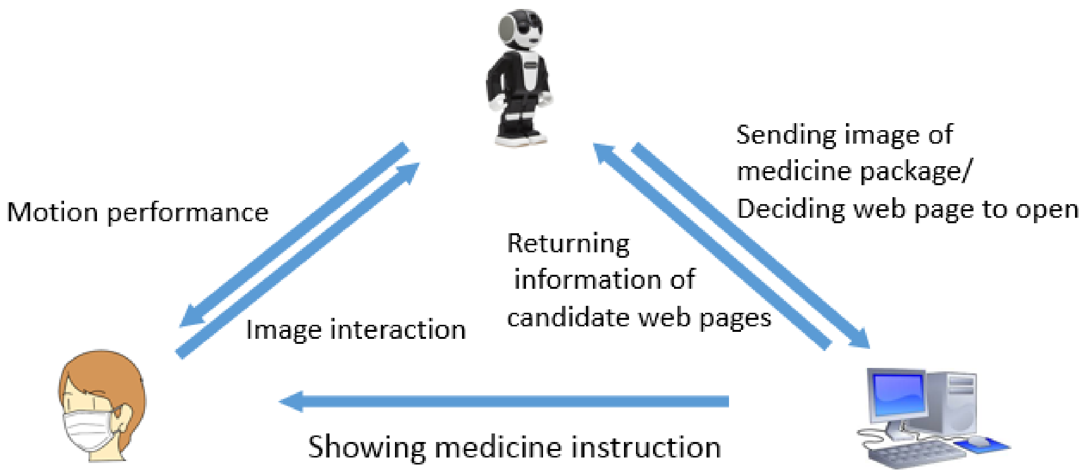

Fig. 4.6 Triangle relation in Showing Medicine Instruction function

4.2.1 Interaction between human and robot

In Showing Medicine Instruction function, agent can interact with user through image interaction and motion interaction.

Image interaction

We use image interaction to make the input of medicine information more interactive and interesting. When the user tells the robot specific words to user, the robot will starts the camera module. At the same time, the robot agent takes a photo of the medicine box with the built-in camera, and output the image of medicine to PC to let PC call image recognition API.

Motion Performance

Robot agent will keep making motions during the conversation with user. When the robot says "OK", it will nods. When the robot can't recognize what the user is saying, he will say "Can't hear clearly" and shake his head. When the robot say "Is that OK", it will open his hands . When it ask user to wait, agent will bow like a human. And when counting down to take photo, agent wave arm up and down like a referee countdowns during the game, shown as Table 4.6. The motions of robot are shown as Fig 4.7.

Hot word/Situation	Motion
"Please wait a little"	Bow
"OK"	Nod
"Can't hear clearly"	Shake head
"Is that OK"	Open hands
"three, two,one"(count down)	Pull the right arm forward and move it up and down

Table 4.6 Motion performance when saying special sentence

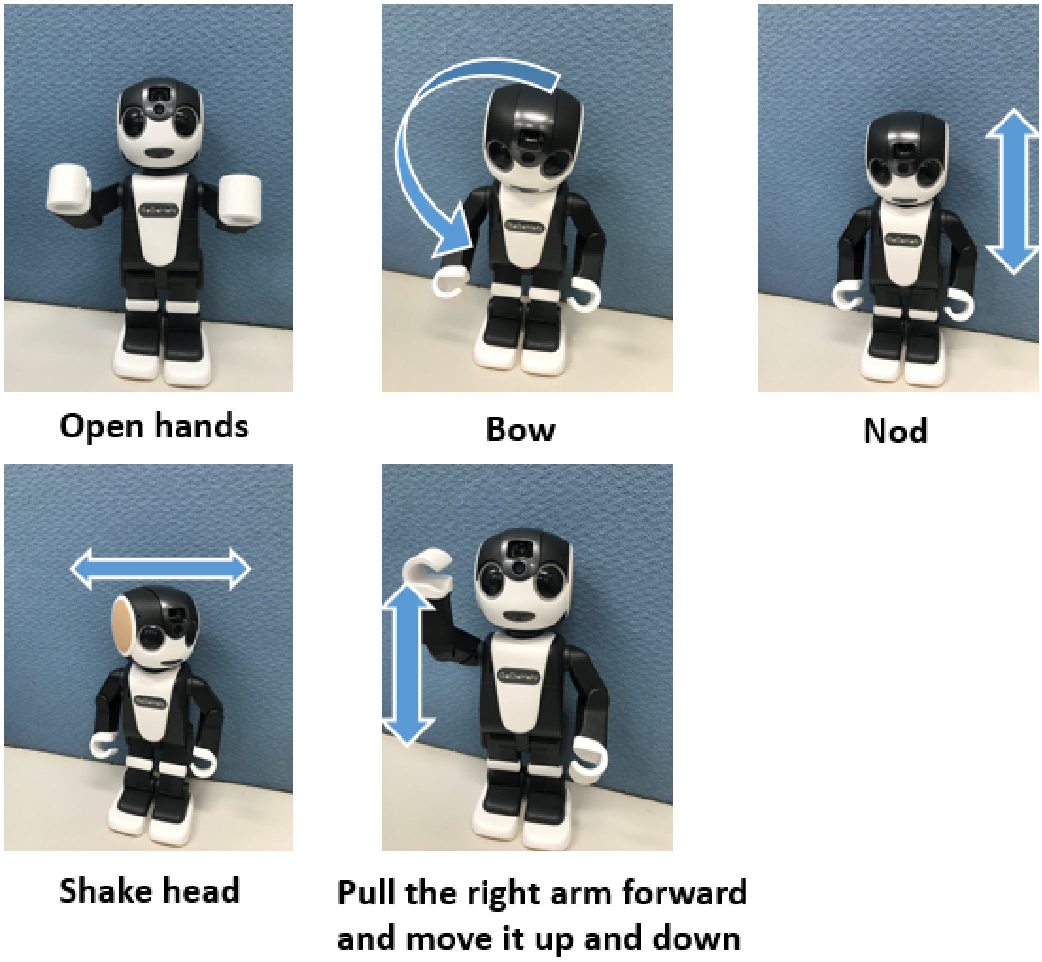

Fig. 4.7 The motions of robot for Showing Medicine Instruction

4.2.2 Interaction between PC and Robot

The interaction between PC and robot includes sending image of medicine package, returning information of candidate web pages and deciding web page to open.

Sending image of medicine package

After taking photo of medicine package, robot sends the image of medicine package get the words information on the medicine package so that system can search the information based on it.

Returning information of candidate web pages

When computer received image of medicine from robot agent, computer will send it to image recognition API (Google Vision OCR) and get the words information of the image. After computer gets the image recognition result, it will uses the data to call search engine API to get the detailed information of the candidate instruction web page.

Deciding web page to open

After getting the information of candidate instruction web page from Internet, computer will send the results back to the robot. At the same time, PC shows a user view showing several options allowing user to choose on the browser. Each option includes the information of candidate instruction web pages. Robot can determines which web page to open according to the voice command from user.

4.2.3 Interaction between Human and PC

After getting candidate web page from the Internet, PC displays the information of candidate medicine instruction web pages. System extracts the elements of target page, including the description of the web page, the link, and the thumbnail. We use a user interface to show these information, shown as Fig 4.6. At the top of the page is the title "Medicine", and below it is the information for each candidate web page, including thumbnails and descriptions of web page.

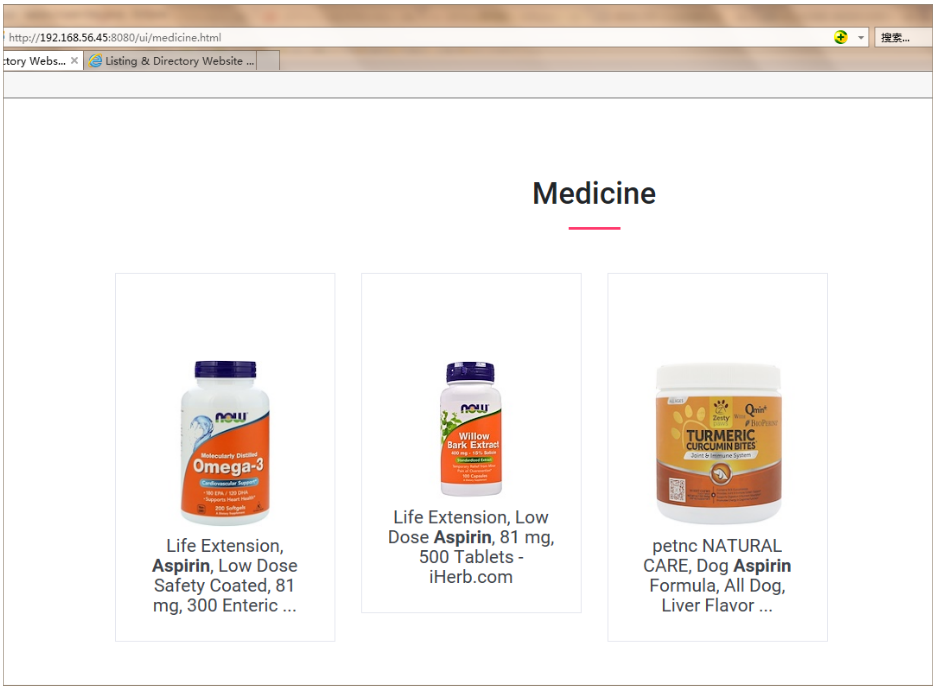

Fig. 4.8 User interface for information of candidate medicine instruction for keywords"aspirin"

Chapter 5

System Implementation

5.1 Development Environment

We have developed this system using Android Studio 2018[URL1] and Microsoft Visual Studio 2017[URL2]. Development languages include Java, hvml, JavaScript and Python. Hvml is a XML Language developed by Sharp[URL3]. We have used hvml and Java language to realized the conversation between robot and user. Python is used to build and manage server in computer. JavaScript is used to build user interface. The voice recognition API is AmiVoice API[URL4], and the imagine recognition API is Google Vision OCR[URL5]. The search engine API is Google Search API[URL6].

URL Reference

[URL1:Android Studio 2018: https://developer.android.com/studio/releases/#3-2-0]

[URL2: Microsoft Visual Studio 2017: https://visualstudio.microsoft.com/ja/vs/]

[URL3: Hvml Lanuage: https://robohon.com/sdk/app.php]

[URL4: AmiVoice API: https://www.advanced-media.co.jp/products/service/amivoice-cloud-2]

[URL5: GoogleVision OCR: https://cloud.google.com/vision/]

[URL6: Google Search API: https://developers.google.com/custom-search/]

5.2 System Hardware

For the personal computer, we have used a windows machine. For the robot agent, RoBoHoN[URL1] is used as an external robot agent. RoBoHoN is a smartphone-robot hybrid that specializes in communicating through conversation. It can recognize voices and responds to voice commands.

Fig. 5.1 RoBoHoN

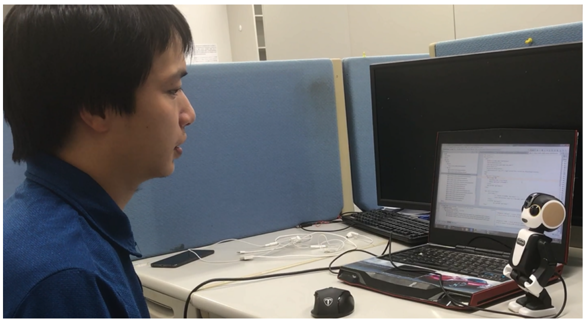

Fig. 5.2 Using RoBoHoN as an external robot agent

URL Reference

[URL1:RoBoHon: https://robohon.com/]

5.3 Providing Health Information

5.3.1 Interaction between Robot and Human

Realize conversation between robot and human

Hvml language is used to design and realize conversation between robot and human. In this conversation, the agent asks voice question and user gives answer question. When agent receives voice information, it will match the content of user's voice with the hot words in dictionary. If match is succeed, agent will execute certain statement and speak out text information. After that, system will go to different conversation branches according to the response of user, as shown in Fig 5.3. We use two ways of words matching, one is rough match. This matching is succeed when user's talk includes hot words. Another is exactly match. This matching is succeed only when user's words is as some as the hot words.

Fig. 5.3 Conversation between robot and human

Collect user's answer

When making the conversation with user, robot asks some typical questions to collect user's health condition like "Do you have a fever" or "Do you have a dizzy". When user gives the voice answer, robot will store the answer and ask the next question, shown as Fig 5.4.

```
<topic id="t9" listen="false">
    <action index="1">
        <speech>Do you feel dizzy</speech>
        <behavior type="normal" id="0x040800"/>
    </action>
    <next href="#t10" type="default"/>
</topic>

<topic id="t11" listen="false">
    <action index="1">
        <speech>I'm sorry I cannot hear you clearly.</speech>
        <behavior id="assign" type="normal"/>
    </action>
</topic>

<topic id="t5" listen="false">
    <action index="1">
        <speech>do you have a fever?</speech>
        <behavior type="normal" id="0x040800"/>
    </action>
    <next href="#t6" type="default"/>
</topic>

<topic id="t7" listen="false">
    <action index="1">
        <speech>I'm sorry I cannot hear you clearly.</speech>
        <behavior id="assign" type="normal"/>
    </action>
</topic>
```

Fig. 5.4 Voice questions to collect user's answer

Generate advice

When making suggestion, agent ask voice questions based on user's health condition like dizzy, fever and degree of pain. The answers given by user are stored in arrays. Then, agent analyzes the contents of user's answers and get results based on answers. To make suggestion, we use decision tree to train data. We use ID3 algorithm to realize simple conversation between user and robot. The data includes possible input when user are asked whether he is dizzy or have some other symptoms. For example, when user feels fever and dizzy and he does not have feeling of pain, he gives the answer"yes, yes, normal" shown as Fig 5.5.

System will judge severity of result into two levels: If situation is serious, robot agent will directly suggest user to see a doctor; if user just needs some self-regulation, agent analyzes related tag for keywords based on etiology or treatment. Then agent send the result to computer.

```
Object[] arrays = new Object[]{
        new String[]{"fever","dizzy","Degree_of _pain"},
        new String[]{"yes","yes","normal","no"},
        new String[]{"yes","yes","high","yes"},
        new String[]{"yes","yes","very high","yes"},
        new String[]{"no","yes","normal","no"},
        new String[]{"no","no","high","no"},
        new String[]{"no","yes","very high","yes"},
        new String[]{"yes","no","high","yes"}};
UtilID3 ID3Tree = new UtilID3();
ID3Tree.create(arrays,  index: 3);

return ID3Tree.result(ddata, dlabel);
```

Fig. 5.5 Training data for ID3 algorithm

Motion performance

We have made robot do motion performance when agent speak out special words by using motion ID. Motion ID is a control code which is used to control the movement of robohon. For example, when we set motion ID as 0x04080a, robot agent shakes head , which is used when agent says "Can't hear clearly". When we set motion ID as 0x060030, robot agent open hands, which is used when agent says "Is that OK". When we set motion ID as 0x060021, robot agent nods, which is used when agent says "OK", as shown in Table 5.1.

motion ID	motion description	what agent says
0x04080a	Shake head	"Can't hear clearly"
0x060030	Open hands	"Is that OK"
0x060021	Nod	"OK"

Table 5.1 Using motion ID to realize motion performance for Providing Health Information

5.3.2 Interaction between Robot and Computer

Set Up Communication Framework

We have set up an internal server in PC using python language. We use 8080 port and built a communication framework for local network. On the robot agent side, we have used HttpURL Connection package of Java to sent quest to PC. Target server address is 192.168.43.4, and port is 8080, as shown in Fig 5.6.

On the PC side, we import http.server package to receive the incoming request from robot agent. We create a web server and define the handle to manage the incoming request, and server will wait forever for incoming http requests, as shown in Fig 5.7.

```
public static String sendInstruction(String instruction, String webURL) {

    String serverUrl = "http://192.168.43.4:8080/"+webURL;
    Log.d( tag: "CONTROL_COMPUTER_URL", serverUrl);
    try {
        return HttpUtil.sendGet(serverUrl);
    }catch(IOException e) {
        e.printStackTrace();
        return "exception";
    }
}
```

Fig. 5.6 Send request from robot to PC

```
try:
    #Create a web server and define the handler to manage the
    #incoming request
    server = HTTPServer(('', PORT_NUMBER), myHandler)
    print ('Started httpserver on port ' , PORT_NUMBER)

    #Wait forever for incoming htto requests
    server.serve_forever()

except KeyboardInterrupt:
    print ('^C received, shutting down the web server')
    server.socket.close()
```

Fig. 5.7 Receive request from robot

How to show Video Information

Send user's situation to PC

We have used Google Search API to get article information from Internet. For the input, robot will send web URL including keywords to server. We have used POST to set up an http request. The type of content to be transferred is "eapplication/x-www-form-urlencoded". We have made the method of connection as Keep-Alive. The charset of http quest is UTF-8. And the string of URL consists of genre of information to search, keywords of symptom and etiology, list of video web sites, target server address which is 192.168.43.4, and port which is 8080, as Table 5.2 shows.

Field	Value	Description
RequestMethod	POST	How to set up an HTTP
Content-Type	application/x-www-form-urlencoded	Type of content to be transferred
connection	Keep-Alive	Set connection method: Keep
Charset	UTF-8	Set HTTP request properties: Character set
url	"http://192.168.43.4:8080/" +keywords+"video"	Send keyword for search

Table 5.2 Request parameters sent from agent to PC

Use Google search API

After server receiving the request from robot agent, it will send request to Google Search API and receive the result. The request url of API and call method are shown as the follows:

Request URL:

https://www.googleapis.com/customsearch/v1

Call methods:

POST

For the request parameters sent to API, in field "q", We sent keywords for search. And in field "cx", We sent hash of list of video web pages, the value of which is "******** *". The number of results to return is 3. And in field "developerKey", we sent licence which is used for this custom search, the key of which is "*************", as shown in Table 5.3.

Field	Value	Description
q	[keywords]	keywords for search
cx	******** :*******	hash of list of video websites
num	3	number of results to return
developerKey	****** ******	licence of custom search

Table 5.3 Request parameters used to get candidate video information when calling Google search API

For the return value sent from google search API, the content in field "link" represents web address of target webpage. The content in field "htmlTitle" represents description of target web page, and the content in field "cse_image" represents overview picture of target web page, as shown in Table 5.4.

Field	Type	Description
link	string	web address of target webpage
htmlTitle	string	description of target webpage
cse_image	string	overview picture of target webpage

Table 5.4 The result given from Google search API

Extract information of candidate video web pages

We selected the most correspond results from JSON file, sent from Google search API. In this part, system extract target information from the result, including web address, description, and overview picture, as shown in Fig 5.8.

```
#video
service = build("customsearch", "v1",
          developerKey="AIzaSyCgyDaK4w1VWFfBzYbCjUXib_fqNTMFNsU")

res = service.cse().list(
    q=need,
    cx='014457193830722971302:rq0fu5b0cxk',
    num=3,
  ).execute()
f1 = open('D:\\server\\tmp\\link.txt','w')
array=['','','']

pos=0
linkpos=0
tasks = res.get('items', [])
for task in tasks:
 array[pos]=task['htmlTitle']
 pos=pos+1
print(array)


for linktask in tasks:
 linkarray[linkpos]=linktask['link']
 linkpos=linkpos+1
print(linkarray)
```

overview picture

description

web address

Fig. 5.8 Core code for extract information from JSON file

Decide the type of information to view

Robot agent can decide which web page to view by receiving voice command. For example, if user's voice command include the words "video", robot agent will start the function of opening the web page that shows candidate video information. If user's voice command include the words "article", robot agent will start the function of open the web page that shows candidate video information. And if user's voice command include the words "social", robot agent will start the function of showing SNS information, as shown in Fig 5.9.

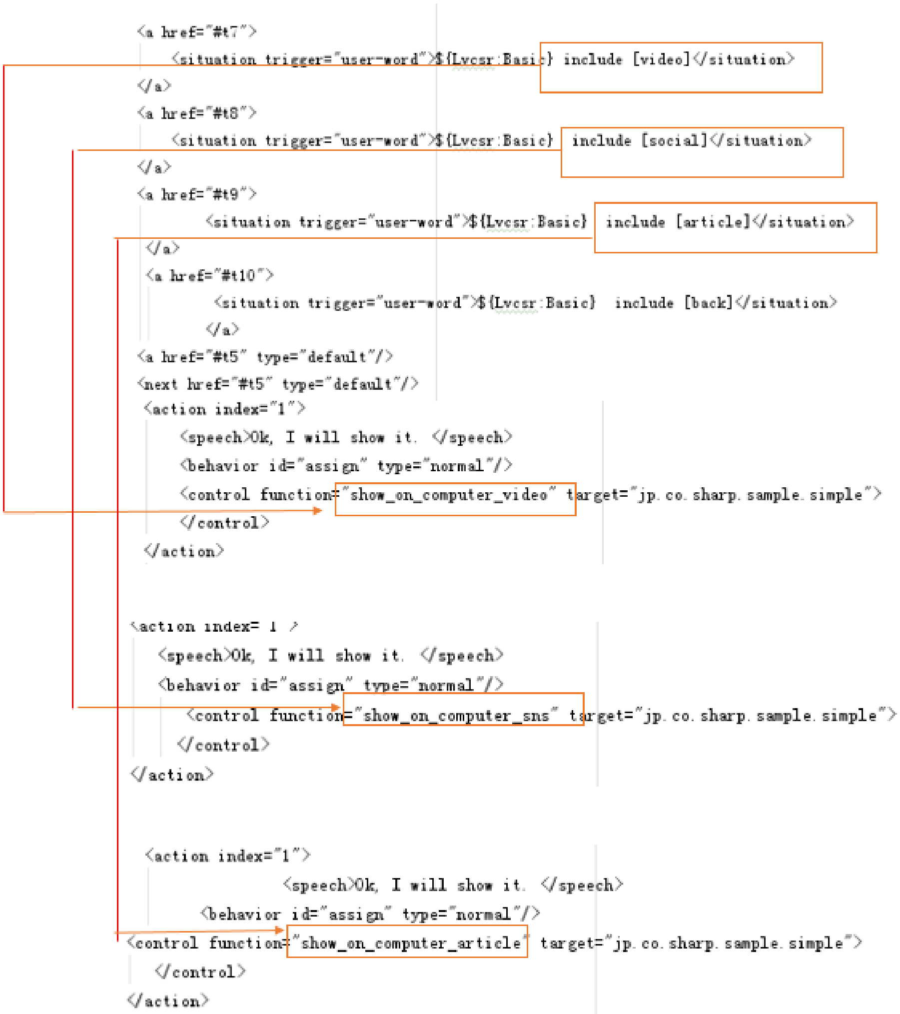

Fig. 5.9 Decide which kind of information to view according to voice choice

Decide suitable video web page

After that, robot agent will decide which web page to open by receiving voice command. For example, if user's voice command include the words "left", robot agent will start the function of opening the first web page. If user's voice command include the words "middle", robot agent will start the function of opening the second web page. And if user's voice command include the words "right", robot agent will start the function of opening third web page, as shown in Fig 5.10.

Fig. 5.10 Decide which web page to view according to voice choice

On the agent side, agent sends http request according to user's choice. On the PC side, system store the links of candidate web pages in an array defined as link_array. After receiving http request, server opens corresponding link stored in array according to the http request. For example, when user chooses left, agent will send a http quest"http://192.168.43.4:8080/_.first" to PC and will open the first link in array. When user chooses middle, agent will send a http request"http://192.168.43.4:8080/_.second" to PC and PC will open the second link in array. When user chooses right, agent will send a http request"http://192.168.43.4:8080/_.third" to PC and store it in the third element of array, as shown in Table 5.5.

Target Option	Request to server	Element in array
Left	http://192.168.43.4:8080/_.first	link_array[0]
Middle	http://192.168.43.4:8080/_.second	link_array[1]
Right	http://192.168.43.4:8080/_.third	link_array[2]

Table 5.5 Http request used to decide suitable web page

Open the target web page

We have made the server use package that control the web browser to open corresponding link stored in array. Server used the function of webbrowser.open() to start up the web browser and access the corresponding web page. For example, if server opens the left web page, the first link in array storing the links of candidate web pages is used as the parameter. If server opens the middle web page, the second link in array is used as the parameter. And if server opens the right web page, the third link in array is used as the parameter, as shown in Fig 5.11.

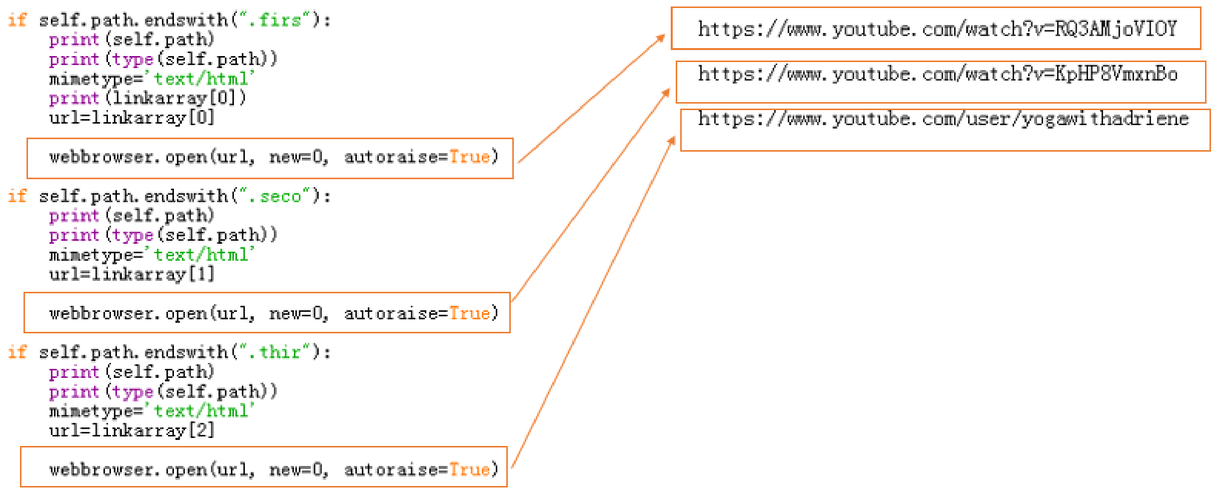

Fig. 5.11 Open the target web page according to the request

How to show Article Information

Compared to communication for Article Information, most processes are the same except sending request to PC and using Google search API.

Sending user's situation to PC

We also have used Google Search API to get article information from Internet. For the input, robot will send web URL including keywords to server. We have used POST to set up an http request. The type of content to be transferred is "eapplication/x-www-form-urlencoded". We have made the method of connection as Keep-Alive. The charset of http quest is UTF-8. And the string of URL consists of genre of information to search, keywords of symptom and etiology, specific article web sites, target server IP address which is 192.168.43.4, and port which is 8080, as shown in Table 5.6.

Field	Value	Description
RequestMethod	POST	How to set up an HTTP
Content-Type	application/x-www-form-urlencoded	Type of content to be transferred
connection	Keep-Alive	Set connection method: Keep
Charset	UTF-8	Set HTTP request properties: Character set
url	"http://192.168.43.4:8080/" +keywords+"article"	Send keyword for search

Table 5.6 Request parameters sent from agent to PC

Using Google search API

After server receiving the request from robot agent, it will give sent request to Google Search API. For the request parameters sent to API, in field "q", We sent keywords for search. And in field "cx", We sent hash of list of article webpages, the value of which is "*******:******". The number of results to return is three. And in field "developerKey", we sent licence which is used for this custom search, the key of which is "*********", as shown in Table 5.7.

Field	Value	Description
q	[keywords]	keywords for search
cx	***** :********	hash of list of article websites
num	3	number of results to return
developerKey	******* *******	licence of custom search

Table 5.7 Request parameters used to get candidate article information when calling Google search API

How to show information from SNS

For the input, robot sent web URL based on keywords to server. Then, PC will directly open the target web page. After that, PC will open the URL on the web browser. For example, when pc shows information on twitter, it will call twitter's own search method. And when pc shows information on facebook, it will call facebook's own search method, as shown in Table 5.8.

Website	Http request to server	Web address that PC opens
Twitter	http://192.168.43.4:8080/ [keywords].sns1	'https://twitter.com/search? f=tweets&vertical=default&q='+ [keywords]+'&src=typd&lang=ja
Facebook	http://192.168.43.4:8080/ [keywords].sns2	'https://www.facebook.com/search /str/'+[keywords]+ '/keywords_search'

Table 5.8 Request parameters used to get candidate SNS information when sending http request

5.3.3 Interaction between Computer and Human

We have used JavaScript language and web browser to build UI and show candidate video web pages to user. To make the UI web page show information of candidate video/article web pages, we stored the information of candidate web pages in temporary file, including description, websites, and picture of overview. When UI page is opened, it will read the information in the temporary file and display a selection interface allowing user to select the target they need, as shown in Fig 5.12 and Fig 5.13.

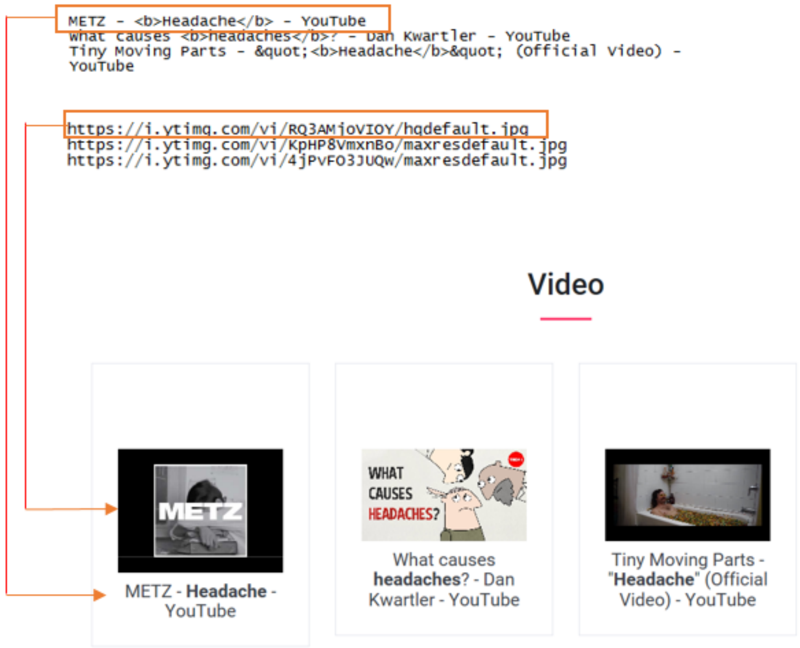

Fig. 5.12 Generate user interface for video

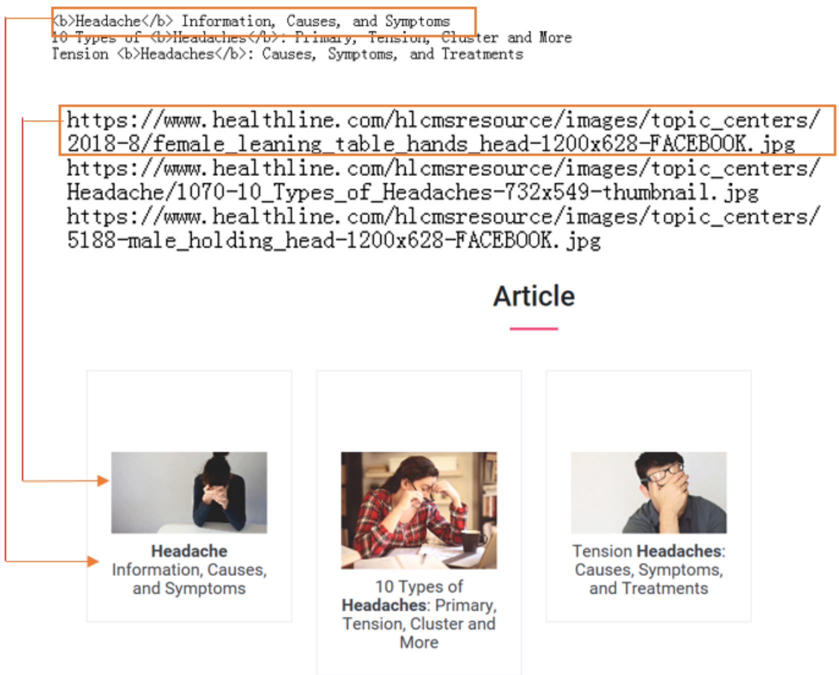

Fig. 5.13 Generate user interface for article

5.4 Showing medicine instruction

5.4.1 Interaction between Robot and Human

Using camera module to take photos

We have used hvml language to design and realize the image interaction between robot and human. When agent receives voice information, it will match the content of user's voice with the hot words. If it detects the sentence "How to use it", agent will and start to count down by saying "3, 2, 1" and execute camera module. User put the medicine box in front of the build-in camera of robot agent and agent will take photo of this medicine. After that, system will save this image and get ready to send it to computer.

Motion performance

We have made robot do motion performance by using the motion ID which is used to control the movement of robohon. For example, when we set motion ID as 0x04080a, robot agent shakes head, which is used when agent says "Can't hear clearly". When we set motion ID as 0x060030, robot agent open hands, which is used when agent says "Is that OK". When we set motion ID as 0x060021, robot agent nods, which is used when agent says "OK", When we set motion ID as 0x060007, robot agent bows, which is used when agent says "please wait a little". When we set motion ID as 0x06005d, robot agent pull the right arm forward and move it up and down, which is used when agent counts down and says "three, two, one", as shown in Table 5.9.

motion ID	motion description	what agent says
0x060007	Bow	"please wait a little"
0x06005d	Pull the right arm forward and move it up and down	"three, two, one"(count down)
0x04080a	shake head	"Can't hear clearly"
0x060030	Open hands	"Is that OK"
0x060021	Nod	"OK"

Table 5.9 Using motion ID to realize motion performance for Showing Medicine Instruction

5.4.2 Interaction between Robot and Computer

Sending image to PC

To allow the transmission of image between PC and agent, we transformed the image file into text information and use http post/get request to exchange the data. After taking photos of medicine package, robot agent reads the contents of the image file and uses a string builder to transforms it into a string. Then, agent use a byte input stream to send it to the PC. When receiving the data sent from agent, PC create a buffer to save the string it has read.

Image Recognition Using Google Vision OCR API

PC call the Google Vision OCR API to recognize the words information on the image. The request url of API and call method are shown as the follows:

Request URL:

https://vision.googleapis.com/v1/images:annotate

methods:

POST

For the request parameters sent to API, in field "app_id", We sent application identifier which is 1257071219. And in field "time_stamp", We sent the request time stamp, the value of which is 7999999. And in field "sign", we sent signature information which is "dFUUrzOg**********". And in field "image", We sent the image to be identified, as Table 5.10 shows.

Field	Value	Description
app_id	1257071219	Application identifier
time_stamp	7999999	Request timestamp
sign	dFUUrzOg**********	Signature information
image	[photo of medicine package]	Image to be identified

Table 5.10 Request parameters used to get candidate instruction information when calling Google Vison OCR search API

For the return value sent from Google version OCR API, the content in field "confidence" represents confidence corresponding to the recognized single character. The content in field "item" represents identification information. And the content in field "itemcoord" represents the pixel coordinates of the word in the image, as shown in Table 5.11.

Field	Type	Description
confidence	float	Confidence corresponding to the recognized single character
item	string	identification information
itemcoord	object	The pixel coordinates of the word in the image

Table 5.11 The result given from Google Vison OCR API

Extract target information

System extracts the all the word information from image recognition result, as shown in Fig 5.14. After that, system matches all the result using keyword in the dictionary. These keywords are For example, if the "itemstring" has the words of "Aspirin", system will judge the name of medicine as aspirin. The results of image recognition are as shown in Fig 5.15.

```
for (int i=0;i<myJsonArray.length();i++){
    JSONObject jo3 = myJsonArray.getJSONObject(i);
    key[i]=jo3.getString( name: "itemstring");

}

return key;
```

Fig. 5.14 Core code to extracts target word information

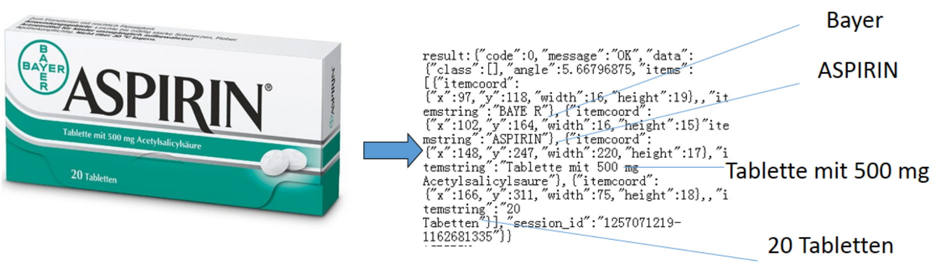

Fig. 5.15 Results of image recognition

Using Google Search API

After PC receiving the image recognition result, it will send request to Google Search API and get the result. The request url of API and call method are shown as the follows:

Request URL:

https://www.googleapis.com/customsearch/v1

Call methods:

POST

For the request parameters sent to API, in field "q", We sent keywords for search. And in field "cx", We sent specific medicine instruction web pages, the value of which is "01445719**** :2wp*****". The number of results to return is 3. And in field "developerKey", we sent licence which is used for this custom search, the key of which is "AIzaSyCgyDa*****", as shown in Table 5.12.

Field	Value	Description
q	[keywords]	keywords for search
cx	01445719**** :2wp*****	hash of list of instrcution webpages
num	3	number of results to return
developerKey	AIzaSyCgyDa*****	licence of custom search

Table 5.12 Request parameters used to get candidate instruction information when calling Google search API

For the return value sent from google search API, the content in field "link" represents web address of target webpage. The content in field "htmlTitle" represents description of target web page, and the content in field "cse_image" represents overview picture of target web page, as shown in Table 5.13.

Field	Type	Description
link	string	web address of target webpage
htmlTitle	string	description of target webpage
cse_image	string	overview picture of target webpage

Table 5.13 The result given from Google search API

Extract information of candidate instruction web pages

We have selected the most correspond results from JSON file, sent from Google search API. System extract target information from the result, including web address, description, and overview picture of candidate medicine instruction, as shown in Fig 5.16.

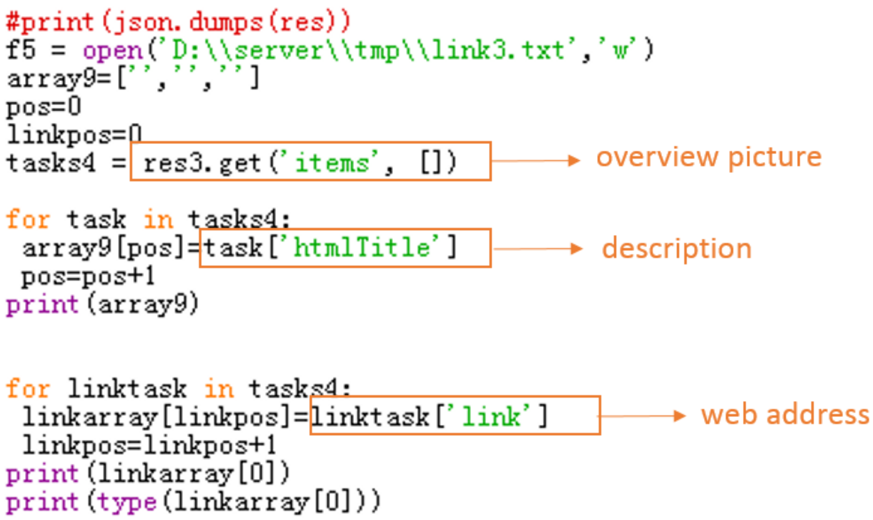

Fig. 5.16 Core code for extract information from JSON file

5.4.3 Interaction between Computer and human

We also have used web browser to build UI and show candidate medicine instruction web pages to user. To make the UI web page show information of candidate medicine instruction web pages, we have stored the information of candidate web pages in temporary file, including description, websites, and picture of overview. When UI page is opened, it will read the information in the temporary file and display a selection interface allowing user to select the target they need, as shown in Fig 5.17.

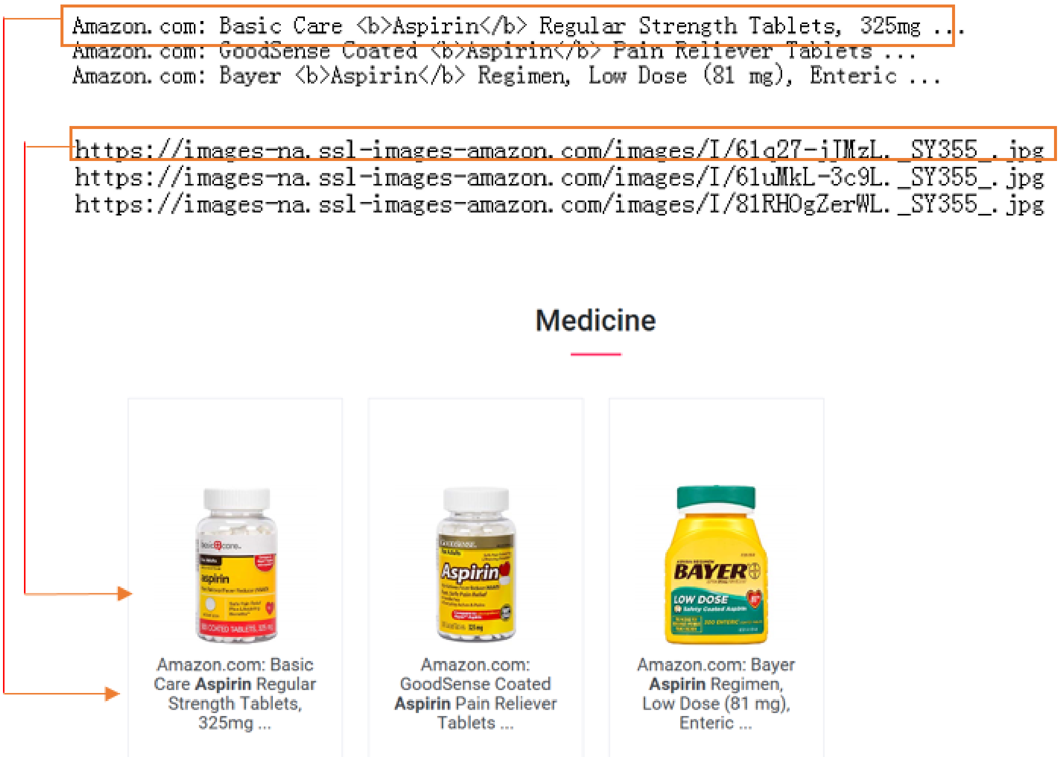

Fig. 5.17 Generate user interface for medicine instruction

Chapter 6

Preliminary Evaluation

6.1 Participants

To evaluate usability and efficiency of our system, we plan to recruit 2 groups. Group 1 consists 6 persons and Group 2 includes 6 persons. Totally we have 12 participants. All participants have basic skill of English reading and speaking.

6.2 Method

In each system, we design two parts. Participants will use the two systems to complete the following experiments:

a) Search for the health-care information according to the symptoms.

b) Search for instruction of the same medicine.

The method we use for evaluation is described in the following n steps:

1. Each participant in Group 1 will be asked to search for the health-care information on the Internet using external robot agent based on same symptom and record the time they have spent before finding the result, while each participant in Group 2 will be asked to do it through system using PC.

3. Each participant in Group 1 will be asked to search for instruction of the same medicine using external robot agent and record the time they have spent before finding the result, while each participant in Group 2 will be asked to do it through system using PC.

4. Each participant will be asked to do the questionnaire after finishing the previous parts and answer 5 questions about using each of the three different systems and write down the time they have spent before finding the result for health-care information and instruction of medicine. The answer is grading from 1 to 5 (1 = very negative, 5 = very positive).

Question	1	2	3	4	5
Do you think it is easy to search the health-care information?					
Do you think it is easy to search the instruction of medicine?					
Do you think the result is helpful for health-care information?					
Do you think the result is helpful for instruction of medicine?					
Do you think this system is attractive?					

Table 6.1 Investigative questions after using the system

The questionnaire is showed in Fig 6.1. We also plan to investigate the time they spend for the task and get their feedback.

Questionnaire for (1. System with external agent 2. System using PC)

(1 = Not at all, 5 =Really think so).

Question	1	2	3	4	5
Do you think it is easy to search the health-care information?					
Do you think it is easy to search the instruction of medicine?					
Do you think the result is helpful for health-care information?					
Do you think the result is helpful for instruction of medicine ?					
Do you think this system is attractive?					

The time you have spent before finding the result for health-care information________.

The time you have spent before finding the result for instruction of medicine__________.

Do you have any comment for the improvement of our system? If any, please write below

____

____

____

Fig. 6.1 Questionnaire

6.3 Result

We have collected the results given by the participants, as shown in Table 6.2 and Table 6.3.

The Table 6.2 shows all the answers from participants using system with external agent. There are 3 participants that choose Grade 5 in Q1, 2 participants that choose Grade 4, and 1 participants that choose Grade 3. In Q2, 2 participants choose Grade 5, 3 participants choose grade 4 and 1 participant chooses grade 3. In Q3, 2 participants choose Grade 5, 2 participants choose grade 4, and 2 participants choose grade 3. In Q4, 2 participants choose Grade 5, 4 participants choose grade 4. In Q5, 4 participants choose Grade 5, 2 participants choose grade 4.

Question	1	2	3	4	5
Q1:Do you think it is easy to search the health-care information?			1	2	3
Q2:Do you think it is easy to search the instruction of medicine?			1	3	2
Q3:Do you think the result is helpful for health-care information?			2	2	2
Q4:Do you think the result is helpful for instruction of medicine?				4	2
Q5:Do you think this system is attractive?				2	4

Table 6.2 Answers statistics of investigative questions from participants for system with external agent

The table 6.3 shows the answers from participants using system with PC. There are 2 participants that choose Grade 5 in Q1, 1 participants that choose Grade 4, and 3 participants that choose Grade 2. In Q2, 1 participants choose Grade 5, 2 participants choose grade 4, 2 participants choose grade 3 and 1 participant chooses grade 2. In Q3, 5 participants choose Grade 4, 1 participant chooses grade 3. In Q4, 3 participants choose Grade 5, 2 participants choose grade 4, and 1 participant chooses grade 3. In Q5, 1 participant chooses Grade 5, 2 participants choose grade 3, and 1 participant chooses grade 2, and 2 participant choose grade 1.

Question	1	2	3	4	5
Q1:Do you think it is easy to search the health-care information?			3	1	2
Q2:Do you think it is easy to search the instruction of medicine?		1	2	2	1
Q3:Do you think the result is helpful for health-care information?			1	5	
Q4:Do you think the result is helpful for instruction of medicine?			1	2	3
Q5:Do you think this system is attractive?	2	1	2		1

Table 6.3 Answers statistics of investigative questions from participants for system using PC

The Fig 6.2 shows the average of grade to each question of all participants. The blue column represents the result for system with external agent, and the orange column represents the result for system with PC. We can see that in Q1 and Q2, system with external agent has relatively higher average grade than system using PC. It means that participants think system with external agent is more convenient than system using PC. And in Q3 and Q4, system with external agent has no significantly difference. It means that for the satisfaction with the returned information, system with external agent and system using PC are almost the same. Especially, for the question 5, the grade of system with external agent is obviously higher than system with external agent, which means that system with external agent is much more attractive than system using PC.

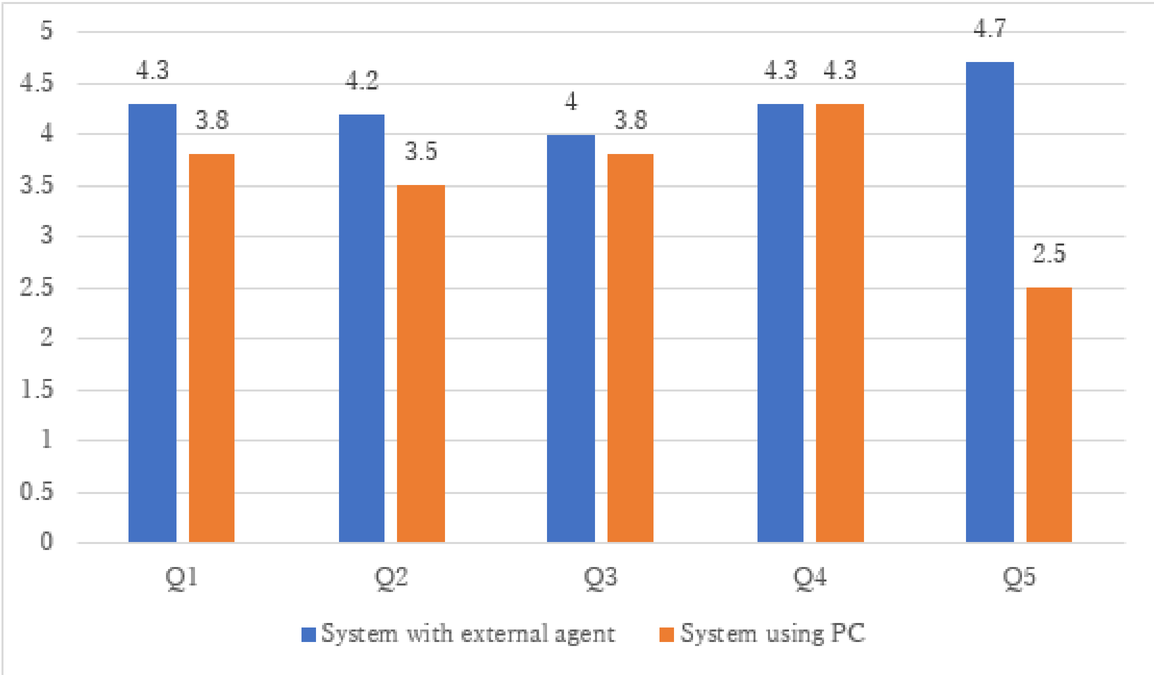

Fig. 6.2 Grade average of each question

The Fig 6.3 shows the average time of searching the health-care information or instruction of medicine. For the average time spent for searching health-care information, system with external agent is 2.8 minutes, while system using PC is 4.8 minutes. And for the average time spent for searching instruction of medicine, system with external agent is 2.3 minutes, while system using PC is 3.5 minutes. The results show that system with external agent costs less time on searching health-care information than system using PC.

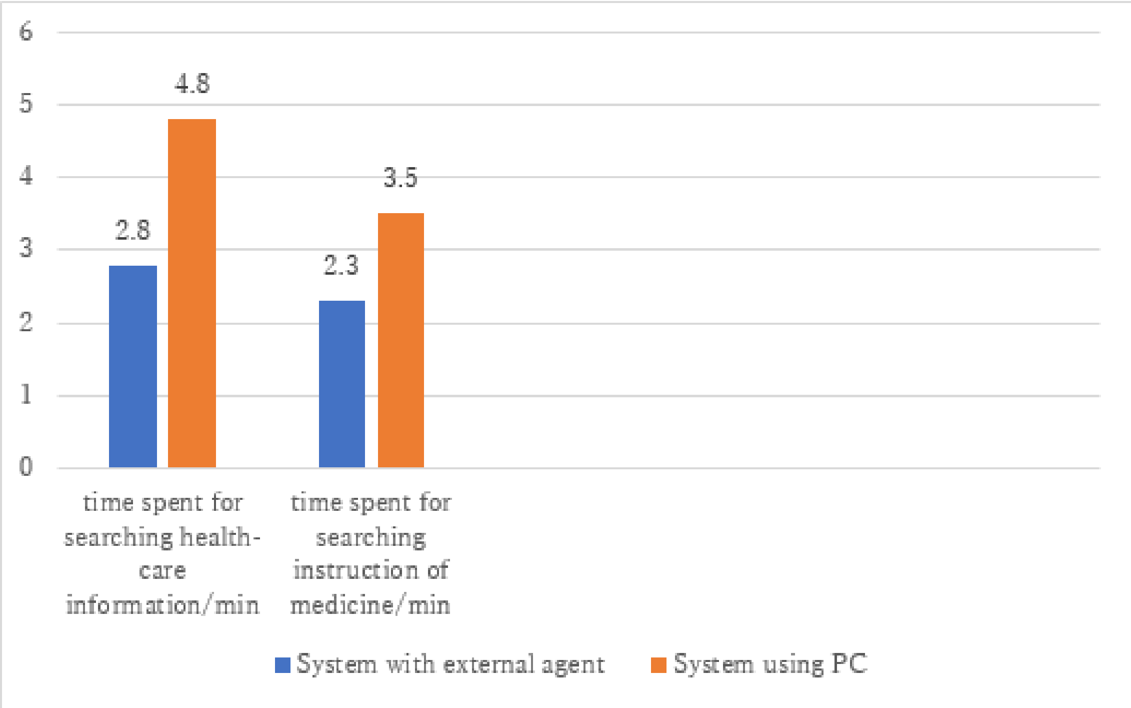

Fig. 6.3 Average time of searching health-care information or instruction of medicine

We also get some comments from participants:

1. "I think you can combine this system with regular medical device like blood pressure gauge"

2. "The conversational interaction is attractive, but sometimes the robot agent can't recognize my words."

Answer: There are some limitations of the voice recognition function of Robohon. Maybe it will be improved by subsequent updates of Robohon.

3. "This system is friendly to novice users. I think it is useful for my grandfather to get health-care information"

Chapter 7

Conclusion

Our study's contribution is to formalize self-health care system as a triangle relation of external agent, human, and computer. In our system, we separated the agents from PC and form a triangular relationship. To achieve this relationship, we took effort to design and implement the interaction between the robot and the user, and the interaction between robot and computer. Since the robot agent is separated, it is not as easy as internal agent to use the computer's resources. Therefore, in order to allow the external robot to operate PC, we made the robot receive the user's input and send it to the computer, and the computer displays the required information on the screen according to the user's input. So that the relation of external agent, human, and computer is established. For the triangle relation of external robot agent-computer-user, all the parts are independent. Thus, the external agent has more expandability and exibility. If it is needed, we can easily extend interactions or improve one of them. This makes the system more scalable. In addition, as the agent is separated from computer, not only the spatial extent of the interaction is allowed, but also the system can support more kinds of interactions. Users can take the agent away and interact with it in a comfortable way such as searching for information in a distance. Through external robots, we can increase the scope of interaction, and the interaction of system is more vivid.

We have integrated motion interaction, voice action, and image interaction to make the process more interactive and interesting. We designed a dialogue-based voice interaction between user and robot. To allow the user to interact with the robot through real object, we have introduced the image interaction. At the same time, during the dialogue between the user and the robot, we also made the robot do motion performance according to the content of the conversation. By integrating these interactions, we made the robot more human-like, allowing users have more participation when communicating with the robot.

We have proposed interactions between robot agent and web browser on PC to make user view health-care web page more interactively and conveniently. Even if the agent searches for information instead of the user, not all results will satisfy the user. To improve user satisfaction and the reliability of the results, we let the computer display several options allowing user to choose on the browser. Each option includes the information of candidate web pages. User can choose the web page he wants to view by voice control. For example, when user says" Open left", PC will open the web page corresponding to the option on the left. When displaying the information of candidate web pages, system extracts the elements of target page, including the description of the web page, the link, and the thumbnail. This information is displayed in a clear and simple user interface. We allowed user to choose the information he wants to make system has better usability and reliability.

References

[1] Jonathan Gratch, Gale Lucas, Aisha Aisha King, and Louis-Philippe Morency. It's only a computer: the impact of human-agent interaction in clinical interviews. In *AAMAS '14 Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 85–92. ACM, 2014.

[2] Timothy Bickmore, Laura Pfeifer, and Brian Jack. Taking the time to care: empowering low health literacy hospital patients with virtual nurse agents. In *CHI '09 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1265–1274. ACM, 2009.

[3] Zeljko Obrenovic, Dusan Starcevic, Emil Jovanov, and Vlada Radivojevic. An agent based framework for virtual medical devices. In *AAMAS '02 Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 659–660. ACM, 2002.

[4] Terry Ellis, Nancy Latham, Tamara DeAngelis, Cathi Thomas, Marie Saint-Hilaire, and Timothy Bickmore. Feasibility of a virtual exercise coach to promote walking in community-dwelling persons with parkinson disease. In *Am J Phys Med Rehabil*, pages 472–485. PMC, 2013.

[5] Nasim Motalebi and Saeed Abdullah. Conversational agents to provide couple therapy for patients with ptsd. In *PervasiveHealth '18 Proceedings of the 12th EAI International Conference on Pervasive Computing Technologies for Healthcare*, pages 347–351. ACM, 2018.

[6] Ari Pollack, Uba Backonja, Andrew Miller, Sonali Mishra, Maher Khelifi, Logan Kendall, and Wanda Pratt. Closing the gap: Supporting patients' transition to self-management after hospitalization. In *CHI '16 Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 5324–5336. ACM, 2016.

[7] Scott Baker, Deborah Richards, and Caldwell Patrina. Putting a new intelligent virtual face on a medical treatment advice system to improve adherence. In *IE2014 Proceedings of the 2014 Conference on Interactive Entertainment*, pages 1–9. ACM, 2014.

[8] Mengxuan Ma, Marjorie Skubic, Karen Ai, and Jordan Hubbard. Angel-echo: a personalized health care application. In *CHASE '17 Proceedings of the Second IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies*, pages 258–259. ACM, 2017.

[9] Hongwei Huo, Youzhi Xu, Hairong Yan, Saad Mubeen, and Hongke Zhang. An elderly health care system using wireless sensor networks at home. In *2009 Third International Conference on Sensor Technologies and Applications*, pages 258–259. IEEE Xplore, 2009.

[10] Marti Hearst. 'natural' search user interfaces. In *Communications of the ACM Volume 54 Issue 11*, pages 60–67. ACM, 2011.

[11] Tessa Lau, Julian Cerruti, Guillermo Manzato, Mateo Bengualid, Jeffrey Bigham, and Jeffrey Nichols. A conversational interface to web automation. In *UIST '10 Proceedings of the 23nd annual ACM symposium on User interface software and technology*, pages 229–238. ACM, 2010.

[12] Bernard Jansen, Tracy Mullen, Amanda Spink, and Jan Pedersen. Automated gathering of web information: An in-depth examination of agents interacting with search engines. In *Transactions on Internet Technology (TOIT) TOIT Homepage archive Volume 6 Issue 4*, pages 442–464. ACM, 2006.

[13] Maxwell Harper, Daniel Moy, and Joseph Konstan. Facts or friends?: distinguishing informational and conversational questions in social qa sites. In *CHI '09 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 759–768. ACM, 2009.

[14] Alexandra Vtyurina, Denis Savenkov, Eugene Agichtein, and Charles Clarke. Exploring conversational search with humans, assistants, and wizards. In *CHI EA '17 Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 2187–2193. ACM, 2017.

[15] Luke Zettlemoyer, Robert Amant, and Martin Dulberg. Ibots: agent control through the user interface. In *IUI '99 Proceedings of the 4th international conference on Intelligent user interfaces*, pages 31–37. ACM, 1999.

[16] Simone Stumpf, Vidya Rajaram, Lida Li, Margaret Burnett, Thomas Dietterich, Erin Sullivan, Russell Drummond, and Jonathan Herlocker. Toward harnessing user feedback for machine learning. In *IUI '07 Proceedings of the 12th international conference on Intelligent user interfaces*, pages 82–91. ACM, 2007.

[17] Fabio Ballati, Fulvio Corno, and Luigi De Russis. Assessing virtual assistant capabilities with italian dysarthric speech. In *ASSETS '18 Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 93–101. ACM, 2018.

[18] Alyssa Glass, Deborah McGuinness, and Michael Wolverton. Toward establishing trust in adaptive agents. In *IUI '08 Proceedings of the 13th international conference on Intelligent user interfaces*, pages 227–236. ACM, 2008.

[19] Simone Stumpf, Margaret Burnett, Volkmar Pipek, and Weng-Keen Wong. End-user interactions with intelligent and autonomous systems. In *CHI EA '12 Extended Abstracts on Human Factors in Computing Systems*, pages 2755–2758. ACM, 2012.

[20] Xuan Wang, Chunmeng Lu, Soh Masuko, and Jiro Tanaka. Interactive online shopping with personalized robot agent. In *IUI '18 Companion Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion Article No. 12*. ACM, 2018.