ISBN 978-981-14-4787-7
Proceedings of 2020 the 10th International Workshop on Computer Science and Engineering (WCSE 2020)
Yangon (Rangoon), Myanmar (Burma), February 26- February 28, 2020, pp. 57-62
doi: 10.18178/wcse.2020.02.010

Development of Spoken Language Recognition System for Humanoid Robot

Khaing Yee Mone [1+], Yoshio Yamamoto [2]

[1] Course of Mechanical Engineering, Graduate School of Engineering, Tokai University, Hiratsuka, Japan
[2] Department of Precision Engineering, School of Engineering, Tokai University, Hiratsuka, Japan

Abstract. "Pepper" robot has a variety of embedded abilities of interaction to communicate with humans such as Speech, Image, Detection, Gesture, and Tablet Service. Our research focused on interaction by speech recognition system of Pepper robot and analyzed its effectiveness. Since there are some limitations in built-in speech recognition system, we combine it with cloud-based speech recognition. The purpose of this research is to get the correct and reliable recognition of verbal speeches from users. Therefore, Speech API from Google Cloud service was used to implement more interactive behavior for Pepper robot.

Keywords: Pepper Robot, Human Robot Interaction, Cloud-based Speech Recognition

1. Introduction

Humanoid robots are increasingly appearing in daily contexts of people's lives, for example, they can assist humans in customer service, welcoming, informing and amusing humans in a kind and familiar manner. The deployment of robots in all these potential applications enriches our lives and, robot's social capability and widespread intelligence are needed to improve more than ever. In this regard, human–robot interaction (HRI) have been drawing considerable attentions in the robotic research community and substantial amount of efforts have been devoted to humanoid robots to improve robot's social skill [1].

Pepper robot, released from Softbank Robotics, is very popular in these kinds of human robot interaction applications. Initially it was particularly designed for an application of business uses in Softbank stores, it becomes a platform of interest for various other applications including academics and home-entertainment areas. Pepper robot was designed by these principles: pleasant appearance, safety, interactivity, affordability and good autonomy [2]. Pepper robot has a smart appearance with its humanoid upper body, operated with amazing functionalities such as emotion perception, speaking with gestures and omnidirectional movement provided by the wheeled base. It can interact with humans through speech, and if the customer requests some information, it can also give answer by either speech feedback, or visual feedback using a tablet [1].

Fig. 1: Waving posture of Pepper robot.

[+] Corresponding author. Tel.: + 81 80 9566 7021
E-mail address: 8bemm084@mail.u-tokai.ac.jp

Fig. 1 shows the waving posture of Pepper robot, welcoming students in our laboratory. Additionally, Pepper's face recognition function is for learning about the users, and continuously updating its knowledge base about the users. For extended interaction capabilities, the android tablet on Pepper's chest can be used either for developing apps that integrate with the robot or as a display by loading web pages, pictures or videos [2]. Although most features on Pepper robot are pre-programmed and demonstrate a conscious thinking process, its structural design and software capabilities offer a great predisposition for human interaction [3]. Therefore, researchers are trying to improve them in order to provide more personalized human-robot interactions.

The aim of our research is to create a behavior for Pepper in which Pepper can imitate human's speech. For this purpose, we investigated the possibility of improvement of speech recognition by using cloud-based speech recognition. It enables robots to access large amount of computing power and Pepper's application can interrogate databases and formulate the most appropriate response.

This paper is organized as follows: we present the experience from the operation of Pepper humanoid robot and its software environment in the next section, these are very important to understand for creation of new robot behaviors, and then we show the development of speech recognition system and the behavior consideration and implementation. In the last section, we discuss current results and describe our future work.

2. Pepper for Human Robot Interaction

Pepper is a fully autonomous humanoid robot designed by Aldebaran Robotics, and released in 2015 by SoftBank Robotics (SoftBank acquired Aldebaran Robotics in 2015). In 2015 Pepper was available only in Japan, in June 2016 also in Europe, and in November 2016 in the US [4]. Pepper robot seems to be one of the best options for implementing and research on HRI, and its height and physical proportions and dimensions appearing more like a human-being and resulting better interaction in HRI. The hardware design and functionality are preprogrammed in a form of an API (application programming interface) [5]. A significant advantage of this robot design is its full programmability.

2.1. Embedded Software

NAOqi is the name of the main software that runs on the robot and controls it. NAOqi provides programming framework to develop applications on the Softbank's robots: NAO and Pepper. Different software development kits are provided for any of these programming languages: Python, C++, Java, and Robot Operation System (ROS).

The NAOqi process provides lookup services to find methods, and network access, allowing methods to be called and executed remotely. Local modules are in the same process, so they can share variables and call each other methods without serialization nor networking, thereby allowing the fast communication. Local modules are ideal for closed loop control. Remote modules communicate using the network, and so it is impossible to do fast access using them [6]. Fig.2 shows the process of NAOqi framework.

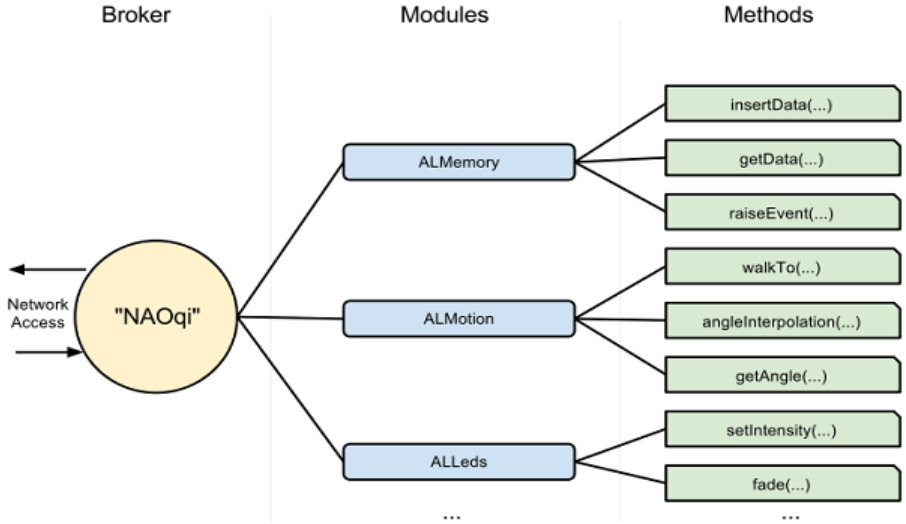

Fig. 2: NAOqi Operating System [6].

The NAOqi executable comes with a list of core modules and a public APIs, which are functionally divided in groups. The default APIs are: NAOqi Core, NAOqi Motion, NAOqi Audio, NAOqi Vision, NAOqi People Perception, and NAOqi Sensors. Among them, the two important modules, used in our interaction behavior creation, are NAOqi Audio and NAOqi People Perception.

NAOqi Framework provides Pepper robot to have an autonomous life, having the basic awareness capabilities and seemingly alive. This shows that Pepper robot is different from others, simple, active and ready to help or interact with humans. And what is the most interesting factor for developers is to develop more interactive behaviors.

Since Pepper has been popular in media as a conversational agent, the work presented in [7] used IBM Bluemix Speech Recognition Service to enhance the robot ability to interact with its users. They integrated Pepper's NAOqi software with ROS to improve the autonomy and making Pepper robot moves autonomously in an environment with humans and obstacles.

Previous work [8] showed that their integration of Pepper with state-of-the-art vision and speech recognition system, and they introduced a learning algorithm to improve communication capabilities over time, which can update speech recognition through social interaction.

2.2. Experience with Pepper Humanoid Robot

This section shows user's experiences with desktop software to control Pepper remotely. One of the effective software to control Pepper is "Choregraphe". The Choregraphe environment is supposed to be used for constructing an application by using some of the built-in function blocks/objects; that are grouped into thematic libraries [9]. It becomes possible to modify functionality of a block/object by developing its Python code. These blocks/objects can be easily and intuitively used for building custom applications with varying degrees of complexity [10]. By working Choregraphe with Python SDK, users can have full access to all build-in modules and can expand robot's functionality with new, practically unlimited resources.

It is a powerful graphical environment that allows the user and developer to work with Pepper with a friendly interface and easy to understand behavior creation, we use Choregraphe version 2.5.5.5. We have the following experiences by using Choregraphe;

- Creation of animations, behaviors and dialogs,
- Testing them on a simulated robot, or directly on a real one,
- Developing Choregraphe behaviors with Python code.

We had successfully done the presentation behavior for Pepper robot created in Choregraphe for the purpose of helping teachers. We presented our work at JSME Robotics and Mechatronics Conference, 2019 in Hiroshima [11].

3. Development of Speech Recognition

In this section, we describe the approach to develop Pepper's speech recognition and people perception. We studied the robustness of Pepper's built-in speech recognition and combination of the existing system with cloud-based speech recognition to improve the accuracy of Pepper's speech interaction.

3.1. Built-in Speech Recognition

Pepper speech recognition system runs by Nuance Solution, a compact speech solution for embedded systems, which is accessed through NAOqi, Speech-To-Text module. This module process speech recognition function with the help of four microphones, placed in the head to provide sound localization, and two loudspeakers, laterally placed on the left and the right sides of the head [12].

However, there are two key limitations for the speech recognition module. The first limitation is insufficient variety of languages since Nuance solution offers only a few languages to choose from, and furthermore the embedded library can recognize only phrases from a predefined set. The human users need to set the vocabulary of phrases to be recognized. Whenever the input audio is matched with a phrase inside the vocabulary, Pepper robot can recognize the audio and understand human's speech. Recognition accuracy is considerably lower for a longer sentence, as the software has more opportunities to make errors. When users say something that is not included in the vocabulary, sometimes, the built-in speech recognition

software is not enough and wrongly matches it to a phrase in the vocabulary. The second one is speech signal weakness. During the conversation, the movement of the robot's head mechanisms (motors, fan) introduces interference. That can slightly change in the strength of the speech signal, which can reduce the quality of speech recognition.

Therefore, the sound collected from Pepper's microphones require to be filtered to remove the noise affected by head's movement. Furthermore, we would like Pepper to learn about its environment and how humans prefer to interact with it, and for that we need Pepper to be able to recognize previously undefined speech.

3.2. Cloud Speech API

In order to solve the limitations encountered in Pepper's speech recognition, we used cloud speech API (speech-to-text service) provided from Google. Google Cloud Speech is a cloud-based streaming speech recognition software platform. Google's speech algorithm consists of a deep neural network that has been trained on a large amount and variety of speech from Google users, and is able to recognize general speech [13]. Users connect to the service and send streaming audio, and the service returns transcription results in real-time, along with a confidence level. While Pepper works with cloud-based speech recognition, the creation of robot's applications can be extended as Natural Language Processing, language translation and free word recognition as shown in Fig. 3.

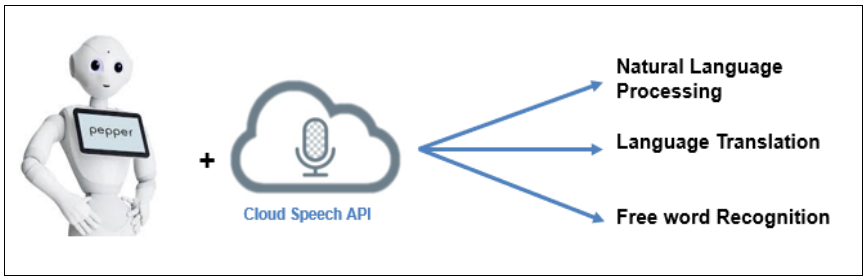

Fig. 3: Pepper robot working with cloud speech API

4. Knowledge Obtained from Trial Results

We tried several experiments for cloud-based speech recognition with the purpose to design minimum hardware requirements and to get the correct operation of speech recognition service. Unlike Pepper's speech recognition, users do not specify the speech they expect to receive. The input audio from each of Pepper's four-microphones is streamed to Google and use the recognition result with the highest confidence. The cloud-based speech recognition service allows for more general speech and appears to perform somewhat better, but it also requires an active internet connection with enough speed for streaming audio.

Table.1 shows the comparison of recognition accuracy between built-in NAOqi speech API and cloud speech API from our trial results. Our experiments work on Pepper robot with NAOqi version 2.9, and we used three kinds of speeches; simple speech (one word), short sentence (three words) and random sentence (more than five words). These results come out from fifty trial times by one user. It shows that using Google cloud speech has higher accuracy than built-in system. And we consider that the possibility of combination of two systems can get the highest accuracy.

Table 1: Comparison of recognition accuracy

Speech Recognition Software	Recognition Accuracy
NAOqi Speech-To-Text	0.56
Google cloud speech API	0.70

The important factor of this research is to test the sound transmission time to the Google Cloud service and resultant text transmission time to robot. We expected that the acceptable maximum processing time between the ends of the human speaking to the beginning of the robot speaking is 2 seconds. During this time, the system will perform the following steps;

• Perform Speech-to-Text on Google cloud service

• Return transcriptions to the robot control system

• Initialize Text-to-Speech module to make robot speak

At the present stage, the speech-to-text result is received immediately after sending human speech to Google cloud. It shows that these steps can be performed in total time of less than 2 seconds. The following section describes the idea of interaction behaviour for Pepper robot.

4.1. Interaction Behaviour

The Pepper robot is supported to be great for entertainment or amusement purposes. We would like to make Pepper imitate human speech after human says something. This means that Pepper robot hears human speaking and speech recognition will be done by cloud speech and then the speech-to-text transcription will be sent to Pepper tablet immediately. And robot's control system will connect the tablet and NAOqi to pronounce the text on the tablet. We show the general configuration of our system in Fig. 4.

Fig. 4: Configuration of our interaction behaviour

5. Summary

This paper presented the idea to develop human's spoken speech recognition of Pepper robot. The purpose is to improve the familiarity between robot and human through communication. At the present stage, we have done with trial experiments to test the robustness of Google cloud speech API and the strength of audio signals recorded by Pepper's microphones. The speech-to-text response time form Google cloud is at acceptable level and it exhibits better accuracy if we can reduce noise affected on audio signal. For the ongoing work, we would like to conduct on behaviour creation that Pepper can imitate human speech.

6. Acknowledgements

We would like to offer special thanks to JICA Innovative Asia Program for their funding support for this research and particularly grateful for opportunities of study in Japan by JICA Innovative Asia Program.

7. References

[1] IEEE Spectrum. How Aldebaran Robotics Builds Its Friendly Humanoid Robot, Pepper. Retrieved February 2015, from https://spectrum.ieee.org/robotics/home-robots/how-aldebaran-robotics-builds-its-friendly-humanoid-robot-pepper, 2014.

[2] SoftBank Robotics, "SoftBank Guidelines Documentation", available on-line (2018-04-01): http://doc.aldebaran.com/download/Pepper_B2BD_guidelines

[3] A. Gardecki, M. Podpora, "Experience from the operation of the Pepper humanoid robots", 978-1-5386-1528-7/17/$31.00, 2017 IEEE.

[4] N. M. Hombur "Designing HRI Experiments with Humanoid Robots: A Multistep Approach", In *Proc. of the 51st Hawaii International Conference on System Sciences*, 2018.

[5] Pepper Robot Programming tutorials from, https://www.about_robots.com/pepper_robot_programm ing.html

[6] *NAOqi APIs and Documentations*, http://doc.aldebaran.com/2-5/naoqi/index.html/

[7] V. Perera, T. Pereira, J. ConnIell, and M. M. Veloso, "Setting up pepper for autonomous navigation and personalized interaction with users," *In CoRR*, vol. abs/1704.04797, 2017. [Online]. Available: http://arxiv.org/abs/1704.04797

[8] Michiel de Jong, Kevin Zhang, Aaron M. Roth, Travers Rhodes, Robin Schmucker, Chenghui Zhou, Sofia Ferreira, Jo ão Cartucho, and Manuela Veloso. "Towards a Robust Interactive and Learning Social Robot". In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS2018),* Stockholm, Sweden, July10–15, 2018, IFAAMAS, 9pages

[9] Choregraphe Tutorials, “how to script the python boxes” http://doc.aldebaran.com/2-1/software/choregraphe/tutos/

[10] E.Pot, J.Monceaux, R.Gelin, and B.Maisonnier, “Choregraphe: a Graphical Tool for Humanoid Robot Programming,” Aldebaran Robotics, ResearchGate, 2009.

[11] Khaing Yee Mone “Advanced Presentation Behavior for Pepper Robot Based on Speech Recognition”. In *proceedings of the Robomech, Robotics and Mechatronics Lecture 2019 in Hiroshima*, June 2019.

[12] G. Suddrey, A. Jacobson and B. Ward, “Enabling a Pepper Robot to provide Automated and Interactive Tours of a Robotics Laboratory”, In *the International Conference on Intelligence Robots*, 2018 IEEE.

[13] Google Cloud Speech, from https://cloud.google.com/speech/