Speech Processing Component Report for PiNAOqio

Xavier Laguarta (xl5512@ic.ac.uk)

***Abstract* - In order to test the hypothesis set in our previous design report, our robot must be able to hold a conversation (speech recognition) and read a physical book (text to speech). Here we will give a brief background on both the speech recognition and text to speech technologies as well as outline the importance of having a natural human robot interactive system. Next, a more in depth study of both speech recognition and text to speech technologies will take place, followed by a comparison of different tools available. Finally the current implementation and the remaining steps will be discussed.**

I. INTRODUCTION

A very important requirement to prove our initial hypothesis defined on the previous Design Report is for our robot to have a natural interaction with humans. This report will focus on the speech interaction, which will take place through our spoken dialog system, comprising of both a speech recognition system and a text to speech system **[1]**. The overall system will involve an initial conversation between the human user and the robot, followed by the reading action and further interaction throughout the reading. Some requirements have been set in order for our system to be robust and available to any user, instead of trained for one specific user **[2]**. These involve having a short domain of words to be recognised, as well as a good quality headset microphones to eliminate as much environmental noise as possible. Initially, some background knowledge on spoken dialog systems will be set, before moving to the human-robot interaction section. Following, an overview of speech processing technology, the tools considered and the main difficulties will be specified. Finally, text to speech technology tools and the current implementation will be outlined.

II. BACKGROUND

Spoken dialog systems constitute of several components that will be described further in the report. First, we have an automatic speech recognizer (ASR) system, with the function of decoding speech into text. Next, a dialog manager and domain reasoner will be in charge of the flow of conversation between the robot and the user **[3]**. More specifically, this will decide what actions must be taken depending on the words spoken and what should be answered. Although some dialog managers incorporate ways to clarify misunderstandings in the conversation, we will not be implementing such a complex system due to the time constraints of the project **[4]**. Finally the text to speech (TTS) system will, as its name infers, transform utterances into speech. Several ASR and TTS systems will be compared.

III. HUMAN-ROBOT INTERACTION (HRI)

The initial conversation between the human and the robot prior to the reading can be seen in figure 1:

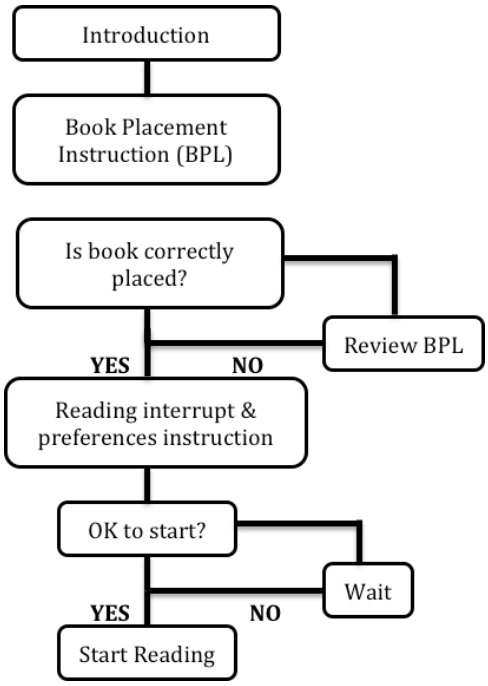

Figure 1: Human robot interaction prior to reading

Above we can observe, the speech interaction is kept at a minimal level to allow for a minimum

number of words in our dictionary. As explained further on, this will give a higher accuracy for the speech recognition system while keeping it user-independent. The initial introduction will comprise of basic salutations and instructions as to how to position the book on the page turning mechanism. Following, a couple questions referring the user's reading preferences will be asked and instructions as to how the user should interrupt the reading in the case they should desire it will be stated. Finally, while the robot reads the passage, it will be listening to any instructions coming from the user.

We are targeting kids in the ages ranging from 4-12 years of age, meaning we must focus on having a simple and robust system. For the best possible user experience, our system should be capable of listening to the user's commands and appropriately answer to them. When looking at the important of communication, several parameters can be analysed for a better HRI. These include both verbal and para-verbal parameters. We identify verbal parameters as the expressions used and para-verbal parameters as the characteristics of speech, including the speed and the volume **[5]**. Due to the complexity of the verbal parameters and the scope of the project, only the para-verbal parameters will be considered.

The overall speech system will communicate through an Ethernet connection. More specifically, an Ethernet cable will be used to connect the raspberrypi, where the computer vision and page turning mechanisms will be running, with the NAO robot. The whole spoken dialog system and NAO movement will be run on the NAOqi operating system. Furthermore, the dialog manager will be created in python. The purpose of a dialog manager is to follow the current state of interaction and combine this with the user inputs to give an appropriate action, which would be the next state **[6]**.

IV. SPEECH PROCESSING

Speech coming from kids is very different than that from adults, and not only in frequency. We must therefore develop a very robust spoken input system. As all things coming from our environment, information has to be digitalised in order to be processed. In the case of speech, we are sampling and quantising air pressure. Once in the form of an audio stream, there is a lot of information we can obtain, varying form linguistic information, prosodic information (emotional) and identity information **[7]**.

Focusing on linguistic information, an Automatic Speech Recognition (ASR) system is used, along with voice decoder software. The decoder is comprised of two sections, the language model and the acoustics model. Firstly, the acoustic model contains sounds we call phonemes, which make up a language. These are sounds that we say when phrasing words or sentences, and are written exactly as they sound. A statistical speech recogniser, which uses a Hidden Markov Model (HMM) **[8]**, can determine the likelihoods of those acoustic observations. A finite-grammar or statistical language model can construct a space where HMMs are built by the acoustic model. Secondly, the language model is divided into word lexicon, the database of words to be recognised by the ASR system, and grammar, containing the sets of words that follow the grammar rules. These will link each sequence of phoneme to words. The grammar will consist of predefined links between words and how likely each link is. The language model however will statistically tell us how likely it is for a word to follow the previous word thanks to the frequency of such followings. N-Grams tells us the N-1 preceding words that can arise. Tri-grams and bi-grams are most often used as these require less training than higher order N-grams and are therefore more often used **[9]**.

Furthermore, there are two methods that can be used to improve the accuracy rate of a system: training an acoustic model for a specific user or having a small vocabulary of words from which to chose from. We have decided to keep our vocabulary of words small in order to keep our model flexible for any user.

V. SPEECH PROCESSING TOOLS

Although the NAO comes with a default speech recognition tool from NUANCE **[10]**, we will be using pocketsphinx for this project because of its local use (offline), being open

source, and its adaptability to robots with relatively small computing power, as well as a better accuracy when compared to the default tool. With pocketsphinx we find an acoustic-model called Hub4, as well as a language model that can be adapted for any speaker **[11]**. Due to the time frame of this project and the goal of interacting with several people at a time, training for specific speakers will not take place, although this would increase the accuracy rate.

When deciding between pocketsphinx and other tools with better performance, such as Google Speech or Nexiwave **[12]**, our choice was based on the tools being local instead of being distributed. Not only do distributed tools drain more battery, they also have a delay of about three seconds, which was not desired for our system. Instead we decided to keep our system local and design it to a more basic standard. Other tools that also proved better performance, such as Julius or Kaldi, were discarded due to the complexity of their documentation and the extensive knowledge needed to operate **[13]**. Finally, NUANCE's best selling software tool called Dragon seems to provide very high accuracy, although this tool must run on machines much more powerful than a rasberrypi, let alone the NAO robot.

VI. ENVIRONEMT NOISE

One of the main issues with speech recognition is the noise coming from the environment. This noise will deteriorate the quality of the signal being inputted into our ASR and therefore decrease the accuracy of our system. Several methods can be used to surpass this complication. Studies show that one of the most important factors in speech recognition is the microphone **[14]**. As the microphone built into the NAO works at an unacceptable level and our robot is intended to read in a fixed position, microphones can be position wherever we find them most desirable to improve the quality of the signal. The best method of receiving a clean signal is through a headset, which the user will speak into directly **[15]**. The short distance between the user's mouth and the headset microphone will allow for a very small amount of noise to be added to the signal. Although headset is the best option, desktop microphones can also work well if its distance to the user is kept low.

VII. TEXT TO SPEECH

Speech synthesis allows for the formulation of human speech from a text string. In PiNAOqio's case, this would be used during the initial introduction with the user, and most of all while reading the book. Here, strings will be passed from the computer vision system and synthesised into speech **[16]**. In order to create these sounds, a system will have a database that stores either words or phones and diphones. The latter will allow for a more flexible system, although the words will not be as clear **[17]**. Festival Speech Synthesis System is an example of a diphones based synthesis tool **[18]**. However, new tools have been developed that allow for a clearer sound.

When choosing the most appropriate text to speech tool, Api.ai and IBM's Watson were the most powerful tools considered. These tools allow for more flexibility when passing text to speech, allowing us to experiment with para-verbal manipulations mentioned earlier **[19]**. More specifically they allow for expression of emotions through voice or the expression of a childlike voice. However, due to the time constraint on our project, and the complexity of the speech recognition section, Aldebaran's default text to speech tool will be used. Aldebaran uses NUANCE's text-to-speech technology, where some customization of the voice can also be accomplished: the speed, pitch and echo of the speech can be pre-defined **[20]**.

VIII. IMPLEMENTATION

Using pocketsphinx as the speech recognition tool and open source code from GitHub, I have started to record sounds that are then to be repeated back to me. This has been a good way to evaluate the most basic features of pocketsphinx. Furthermore, in order to evaluate the improvement from personalized speech recognition, I have done research as to how to create a personalized acoustic model. Next, the initial conversation system will be built, where the conversation will be guided to the desired direction and the reading will begin once the book is well placed. Finally, Aldebaran's text to speech tool allows for an

easy implementation of string to speech. Although individual strings have been tested, the next step will be testing this tool with strings coming from rasberrypi's computer vision.

IX. CONCLUSION

Speech processing is one of the main points to consider for PiNAOqio's user experience while reading, and hence an important point in our hypothesis. Although spoken dialog systems are constantly ameliorating and can be customised to great extents, for the purpose of this project the potential of speech recognition and text to speech tools have been kept to a basic level. The implemented system will allow PiNAOqio to have a basic initial conversation and read while still allowing further input from the user. Although the implementation is currently at an initial stage, the results of the most basic version have been positive.

REFERENCES

[1] T. Kanda, M. Shiomi, Z. Miyashita, H. Ishiguro, and N. Hagita, "A communication robot in a shopping mall," IEEE Robotics and Automation Society, vol. 26, no. 5, pp. 897–913, 2010.

[2] M. A. Goodrich and A. C. Schultz, "Human-robot interaction: a survey," Foundations and Trends in Human-Computer Interaction,
vol. 1, no. 3, pp. 203–275, 2007.

[3] McTear, M. F., 2004. "Spoken Dialogue Technology: Toward the conversational user interface", 1st ed. Springer, pp. 323-324.

[4] Shriberg, Elizabeth, (2004): "Direct modeling of prosody: an overview of applications in automatic speech processing", In SP-2004, pp. 575-582

[5] Z. M. Hanafiah, C. Yamazaki, A. Nakamura, and Y. Kuno, "Human- robot speech interface understanding inexplicit utterances using vision," in CHI'04 extended abstracts on Human factors in computing systems. ACM, 2004, pp. 1321–1324.

[6] Kozima, H., Nakagawa C., and Yasuda, Y, "Interactive robots for communication-care: A case-study in autism therapy", 2005, pp. 341-346, 2005.

[7] S. Fujie, Y. Ejiri, Y. Matsusaka, H. Kikuchi, and T. Kobayashi, "Recognition of paralinguistic information and its application to spoken dialogue system," IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU '03), pp. 231-236, 2003.

[8] M. Gerosa, D. Giuliani, S. S. Narayanan, and A. Potamianos, "A review of asr technologies for childrens speech," Proc. WOCCI'09, 2001, vol. 49, pp. 847–860.

[9] D. Jurafsky and J. H. Martin, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition", 2nd ed. Prentice Hall, 2009.

[10] ALSpeechRecognition — Aldebaran 2.1.4.13 documentation. 2016. (Available at: http://doc.aldebaran.com/2-1/naoqi/audio/alspeechrecognition.html.)

[11] D. Huggins-Daines, "Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices," in Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 2006. (ICASSP 2006). Toulouse, France: IEEE Xplore, May 2006.

[12] Lange, P., Suendermann-Oeft, D.: Tuning Sphinx to outperform Google's speech API. In: Proc. of the ESSV. (2014)

[13] Lee, A., Kawahara, T., & Shikano, K. (2001). Julius-An open source real-time large vocabulary recognition engine. NAISTAR.

[14] Y. Sasaki, S. Kagami, H. Mizoguchi, and T. Enomoto, "A predefined command recognition system using a ceiling microphone array in noisy housing environments," in Proceedings of the 2008 IEEE/RSJ. Nice, France: IEEE Xplore, Sep. 2008, pp. 2178–2184.

[15] W. Khaewratana, "Development of a Voice-Controlled Human- Robot Interface," Thesis, RIT Scholar Works.NY, 5th March 2016.

[16] Taylor, Paul (2009). Text-to-speech synthesis. Cambridge, UK: Cambridge University Press.

[17] Allen, Jonathan; Hunnicutt, M. Sharon; Klatt, Dennis (1987). From Text to Speech: The MITalk system. Cambridge University Press.

[18] W. Black and K Lenzo. 2007. Festvox: Building synthetic voices, Version 2.1. http://www.festvox.org/bsv/.

[19] Imai, T. Kobayashi, K. Tokuda, T. Masuko, K. Koishida, S. Sako, and H. Zen. 2009. Speech signal processing toolkit (SPTK), Version 3.3. http://sp-tk.sourceforge.net.

[20] Nuance.com. 2013. Aldebaran Robotics and Nuance Revolutionize Human – Machine Interaction. http://www.nuance.com/company/news-room/press-releases/Aldebaran_web.docx.