The speech interaction system

Abstract. The speech interaction system is mainly based on the Baidu-voice and Xunfei open platform, and the main function is to realize speech recognition and speech synthesis. Through the combination of the speech system and the Turing chat robot, the robot can chat with people and answer some questions. Judging the user's intention by analyzing the result of speech recognition, then choose to control the movement of the robot or chat with the robot or play music. In addition, in order to be able to chat more naturally with the robot, I also studied the voice activity detection technology.

1 Introduction

In this easy, firstly, I will discuss the traditional voice activity detection technology, and then I will introduce the main function of the speech interaction system.

2 Background

Imagine that there is a scene where several people who have just finished their meal and began to take the taxis home. And then the robot will help them carry their luggage and lead them to the taxi spot. In this process, people will have some interaction with the robot, especially the speech interaction, for example, calling the robot, chatting with the robot, etc. As an important means of interacting with the robot, speech interaction can not only realize the communication and interaction with the robot, but also control some functional modules of the robot through voice.

3 Voice Activity Detection

Voice Activity Detection, also known as Endpoint Detection, is mainly used to detect the start and end points of speech in an input audio signal, and to separate speech from non-speech. The voice activity detection is essentially to find the feature parameters that can distinguish the speech segment and the background noise segment to accurately divide the speech parts and the non-speech parts. Excellent voice activity detection method can reduce detection time, adapt to harsh noise environment and improve detection accuracy. And I used the traditional double threshold detection method.

3.1 Main Parameter

We know that in a speech, there are silent parts and voiced parts. The silent parts include unvoiced, noisy and silent parts. The energy of the speech signal changes obviously with time. On the one hand, the energy of the different parts is different, the silent part has the lowest energy while the voiced part has the highest energy and the

unvoiced part is smaller than that of the voiced part. On the other hand, the zero-crossing rate of the different parts is also different, the unvoiced part is higher than that of the unvoiced part. Therefore, the speech and non-speech parts can be distinguished by energy and zero-crossing rate.

3.1.1 Short-term Energy

The short-term energy is an abbreviation of short-term average energy. After the speech signal is subjected to frame processing, the short-term energy value of each frame is equal to the sum of the squares of the sample values in the frame.
Short-term energy calculation formula is as follows:

$$E_n = \sum_{m=-\infty}^{\infty} \square \left[x(m)\omega(n-m) \right]^2 = x(n)^2 * \omega(n)^2$$

Where $\omega(n)$ is a window function.
The short-time energy parameters have better performance in the following aspects:
(1) can be used as a distinguishing parameter for unvoiced and voiced sounds;
(2) in the case of high signal-to-noise ratio, short-term energy can be used as a basis for distinguishing between sound and silence.

3.1.2 Short-term Zero-crossing Rate

The short-time zero-crossing rate of a speech signal refers to the number of times a signal waveform passes through the horizontal axis (zero level) in a unit time to change the symbol. By calculating the number of horizontal axes that pass through each frame time and then dividing by the number of sample points per frame, the short-term zero-crossing rate in each frame of speech can be obtained. The calculation formula is as follows:

$$Z\,n = \frac{1}{2} \sum_{m=-\infty}^{\infty} \left| sgn\left[x(m) \right] - sgn\left[x(m-1) \right] \right| \cdot \omega(n-m)$$

Where sgn(n) is a symbolic function:

$$sgn\left[x(n) \right] = \begin{cases} 1, x(n) \geq 0 \\ -1, x(n) < 0 \end{cases}$$

The w(n) function is used to calculate the ratio:

$$\omega(n) = \begin{cases} \frac{1}{2N} & 0 \leq n \leq N-1 \\ 0 & others \end{cases}$$

In a complex noise environment, the single zero-crossing rate feature does not have good discrimination characteristics, especially in a strong noise environment, the value of the zero-crossing rate continues to increase, which increases the difficulty of discriminating speech and noise to some extent.

3.2 VAD Algorithm Implementation

Before starting voice activity detection, first determine two thresholds for short-term energy and zero-crossing rate: one is a low threshold, its value is relatively small, it is sensitive to signal changes and easily overtaken. The other is a high threshold, the value is relatively large, the signal must reach a certain intensity and then the threshold may be exceeded. If the low threshold is exceeded, it is not necessarily the beginning of speech, it may be caused by short-time noise. If the high threshold is exceeded, it can be basically believed to be due to the voice signal.

The entire voice activity detection can be divided into 4 segments: silence segment, transition segment, voice segment, and end. In the silent segment, if the short-term energy or zero-crossing rate exceeds the low threshold, start marking the starting point and entering the transition segment. In the transition section, since the value of the parameter is relatively small, it is not sure whether it is in the real speech segment. Therefore, as long as the values of the two parameters fall below the low threshold, the current state is restored to the silent segment and if the two any one of the parameters exceeds the high threshold, and it can be sure that it has entered the voice segment.

3.3 Problems and Solutions

3.3.1 Main Problems

Voice is greatly affected by factors such as equipment, interference sound, and surrounding environment. In many cases, due to the uncertainty of the actual application environment, there may be various noises with different signal-to-noise ratio (SNR), and even multiple noises may be aliased. The traditional double threshold method uses fixed short-time zero-crossing rate and short-time energy threshold to discriminate speech signal and background noise. As the audio SNR decreases, the spectral entropy value of speech and background noise is small. Therefore, the traditional The double threshold method has poor accuracy in speech endpoint discrimination in low SNR environments. In addition, if use the static threshold to detect the speech, its value is difficult to determine, and it is unscientific to use a fixed threshold to detect the speech of different speakers in different situations.

3.3.2 Solutions and other VAD Technology

Bell Laboratory first proposed the technology of speech signal endpoint detection in 1959. It has been nearly 60 years since then, and endpoint detection technology has also innovated many methods. There are endpoint detection algorithms specifically for low SNR, for example, in 2011, Zhang Xiaolei et al. proposed the support vector machine (SVM) based VAD using the multiple observation compound feature (MO-CF)[1], in 2012, Wang Hongzhi et al. proposed the voice activity detection algorithm based on Mel frequency cepstrum coefficient (MFCC) similarity[2] and so on. And there are also other good algorithms like the dynamic double-threshold speech endpoint detection algorithm based on spectral entropy and KL divergence Adaptive threshold speech Endpoint Detection algorithm, etc.

4 Speech Interaction

Voice technology is an indispensable part of realizing a reliable and convenient human-computer interaction system. At present, there are many institutions engaged in the research of voice technology. Among them, the voice technology of Xunfei and Baidu is very mature, and they also provide the interface to be used, and users can obtain massive voice resources through the network connection open platform, and realize many speech functions such as speech recognition, speech synthesis, and semantic understanding and so on.

Using the voice cloud platform to build a product with voice interaction function under the cloud technology architecture is the current mainstream solution. The terminal device only needs to be responsible for the capture of the voice signal and the final audio output. For the process such as speech recognition and semantic understanding, all are composed of the voice cloud platform.

4.1 Audio Admission

The endpoint detection algorithm I used was not very effective, so I finally decided to use voice wake-up technology to start the node that recorded the audio. The main idea is to identify the keyword "jack" by pocketsphinx. The get_audio node starts running and records the audio after receiving the message, and then saves the audio file to a folder. The recording and saving of audio from the microphone is mainly implemented by Python's pyaudio library.

4.2 Speech Recognition

At present, both Xunfei and Baidu provide developers with free speech recognition services, and the recognition accuracy of both parties is comparable. To support different platforms, the service modes supported by these two recognition engines are the same. The first way is to request the service from the cloud recognition engine through the audio file and obtain the recognition result. The second is to send the real-time audio data stream to the cloud recognition engine by means of TCP/IP long connection, and continuously feedback the recognized result. Different ways have their own advantages and disadvantages, the former is not as good as the latter in real-time, but the latter will occupy broadband resources for a long time, and this time I used the first way.

The main task of the speech recognition node is to subscribe to the topic published by the audio recording node, parse the cloud service request parameter in the configuration file, and then initiate the request to the cloud, and finally publish the obtained message.

4.3 Speech Control

The module will subscribe to messages from speech recognition, and through the analysis of keywords, judge the user's intentions, mainly divided into three directions, including controlling the movement of the robot, chatting with the Turing robot, and

playing music.

4.5 Speech Synthesis and Audio Play

Speech synthesis is the conversion of text information into audio data. Voice play is the last functional module of the speech interaction system, the function of this module is mainly to play the synthesized audio or the local music files. The sound_play package provides a way to say strings, and sound_play provides a ROS node that translates commands on a ROS topic (robotsound) into sounds and supports built-in sounds, playing OGG/WAV files. Therefore, speech synthesis and audio play can be achieved by sound_play.

Figure 1 shows the relationship of each node in the speech interaction system.

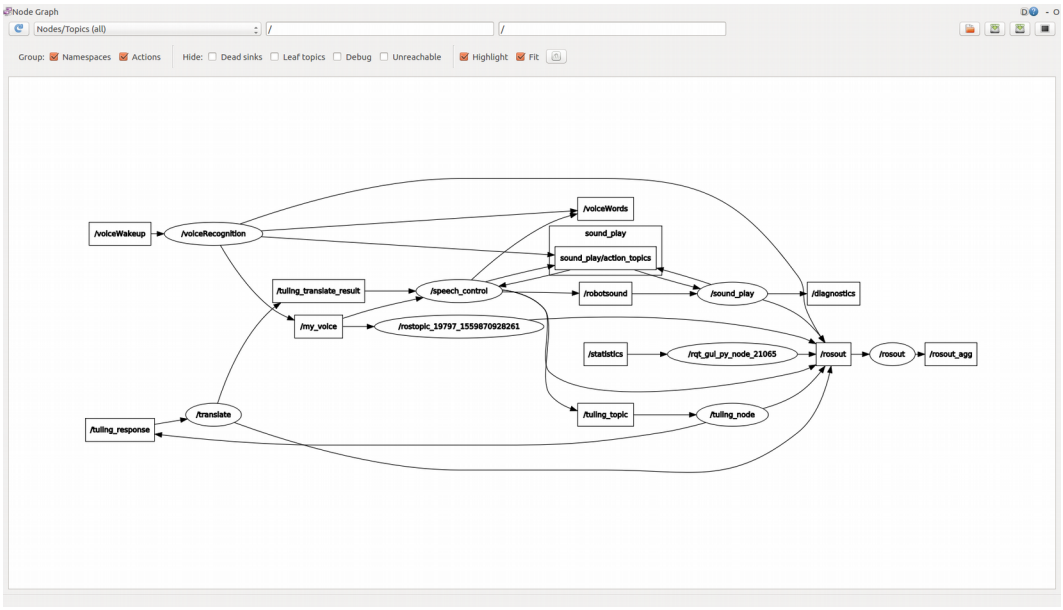

Figure 1

4.6 Problems and Solutions

First of all, when recording audio, all the sounds transmitted into the microphone are directly recorded without any audio pre-processing, and the voice is often affected by factors such as equipment, interference sounds, and the surrounding environment. If the venue noise is relatively large, And the microphone denoising effect is poor, then the recorded sound quality is very bad, this situation will make the speech signal in the subsequent processing with a non-pure signal as the processing object has an impact on other applications, especially the accuracy of the speech recognition. In fact, the results of many experiments show that the effect of speech recognition is really not very good, usually it has a better effect only when the robot is in the environment of relatively small noise interference. Therefore, if the audio data is subjected to pre-processing such as endpoint detection, frame windowing, voice noise reduction, and voice enhancement while recording audio, and then the audio file is saved, in these way, the effect of speech recognition will be greatly increased.

In addition, since the wake-up word needs to be used to wake up the voice recording

node, each time you want to communicate with the robot, you should call the wake-up word "jack", although this will ensure the recording of the speaker's voice, but this will also greatly affect the user experience. In addition, online speech recognition and speech synthesis sometimes cause slow response due to poor network, resulting in a disfluency chat with the robot, but this is a small problem, the main problem is how to improve the accuracy of speech recognition.

5 Experiments and results

Through many experiments, it is found that the effect of Baidu speech recognition is not ideal, and the recognition effect is poor. Even in the case of less noise, the recognition accuracy is not high. Therefore, I finally chose the online speech recognition of Xunfei. The recognition accuracy has been greatly improved, and I can chat with the robot normally. In addition, through the keyword analysis of the recognition results, even if not accurately identify each word spoken by the user, the user's intention can also be judged by keyword recognition, and then voice interaction with the robot is performed.

6 Conclusions and future work

6.1 Conclusions

In this paper, I studied the speech endpoint detection technology and discussed the methods to improve the quality of audio signal and the accuracy of speech recognition. Since the traditional double-threshold endpoint detection technology did not achieve the expected effect, the speech wake-up technology is finally used to wake up the speech collection node. And then through the Python pyaudio library to complete the work of collecting voice signals from the microphone, and next use Baidu speech recognition technology to achieve speech recognition. What's more, human-robot interaction is achieved through the combination with the Turing robot or with other functional nodes.

6.2 Future work

This paper focuses on the cloud voice service to achieve some human-robot interaction functions, but due to the limitations of resources and my own level, there are many areas to be improved.

First, I only used a single-channel microphone for audio acquisition, and there is no in-depth study on speech preprocessing. For the endpoint detection problem, I only studied the traditional double-threshold method, but it didn't achieve satisfactory results. However, good voice information is indispensable for a reliable human-robot speech interaction system. Therefore, in the future, it can be considered how to solve the noise problems, speech enhancement problems, sound source localization problems, etc. Noise problems and speech enhancement problems can be achieved by optimization algorithms. For sound source localization problems, microphone array support can be added in subsequent developments to support sound source localization.

Second, echo cancellation is an important component of the speech interaction system, but this paper does not involve the analysis of the algorithm. In a complete speech interaction process, voice playback should support real-time interruption, so in the next step it should be considered the combination of hardware and software to improve this aspect.

Third, since this paper is based on the cloud model, the network state will inevitably affect the performance of the system and the system's response speed is heavily dependent on the current network state, so as long as there is a little network latency, its user experience will be greatly compromised . Therefore, it is necessary to add offline speech recognition and speech synthesis support for the system and to achieve a better interactive experience by monitoring the network status and switching between online and offline at any time.

References.

[1]Zhang Xiaolei, Wu He and Lu Ping. Speech endpoint detection based on support vector machine (SVM) based VAD using the multiple observation compound feature (MO-CF)[J]. Journal of Tsinghua University (Natural Science Edition), 2011, 51 (09): 1209-1214.

[2]Wang Hong-zhi, Xu Yu-chao, Li Mei-jing. Voice activity detection algorithm based on Mel frequency cepstrum coefficient(MFCC) similarity[J]. , 2012, 42(05): 1331-1335.