Development of Note-Taking Support System with Speech Interface

Kohei Ota*, Hiromitsu Nishizaki† and Yoshihiro Sekiguchi†
* Department of Education, Interdisciplinary Graduate School of Medicine and Engineering,
† Department of Research, Interdisciplinary Graduate School of Medicine and Engineering,
University of Yamanashi, Kofu-shi, Yamanashi, Japan
E-mail: {kota,nisizaki,sekiguti}@alps-lab.org Tel/Fax: +81-55-220-8361/8778

***Abstract*—This paper describes a note-taking support system with a speech interface. To solve problems with existing note-taking methods, we implemented a speech interface consisting of a combination of a touch panel and graphical user interface in a note-taking support system. As a system user listens to a speech, the content of the speech is recognized and displayed on the system's screen. Users can take notes by simply touching or tracing the words automatically displayed on the screen. In addition, the system can support keyboard and handwritten input to cope with speech recognition errors. The developed system was experimentally compared with another note-taking method, a text editor on a personal computer. Most of the subjects could take a note more quickly using the system than using the text editor. The effectiveness of the system was demonstrated in the experiment.**

I. Introduction

We often take notes while listening to a speech or a talk when we attend a lecture or meeting. We usually write a note on a scratch paper or take a note using a text editor on a personal computer (PC) using a keyboard. In addition, we usually record a speech for later access to the entire speech.

These memo tools have both advantages and disadvantages. For example, to take a handwritten note on a paper, we need only a pen and a paper, so it is easy to take a note anywhere. Therefore, most people like making handwritten notes on paper. However, it takes time to write a long or complex phrase, and people may miss hearing something important because they focus too much on writing. On the other hand, note-taking using a PC's keyboard may be a good method for people who are familiar with computers. However, it is inappropriate for the computer-illiterate.

An automatic transcription system for meetings that uses automatic speech recognition (ASR) technology was recently developed [1]. And, a note-taking system for classroom lectures at universities has also developed [2]. In addition, note-taking systems, which can work at Android mobile devices, with a speech interface are released. They are automatic note-taking systems and can automatically record the transcription of a speech without any human effort. However, the transcription is not readable if it contains many errors.

Kurihara et al. [3] reported a speech pen that can be used to make handwritten notes on electronic paper. This system estimates characters using handwritten character recognition and ASR technology and provides character candidates to a user. It focuses on reducing the user's burden; however, recognition errors interrupt the note-taking operation.

Therefore, we have developed a note-taking support system with a speech recognition interface called the Kikimimi interface [4]. The Kikimimi interface always captures a speech and recognizes it. This reduces a user's work load of the note-taking operation.

The system has the following three main features:

1) users can reduce the time and work required for note-taking,
2) the system has a user-friendly interface for the computer-illiterate, and
3) speech recognition errors do not dramatically affect the work required for note-taking.

To evaluate the effectiveness of our system, we performed an experiment regarding note-taking and asked subjects to answer a questionnaire. The experimental results showed that our system was useful for note-taking work.

II. Note-taking support system

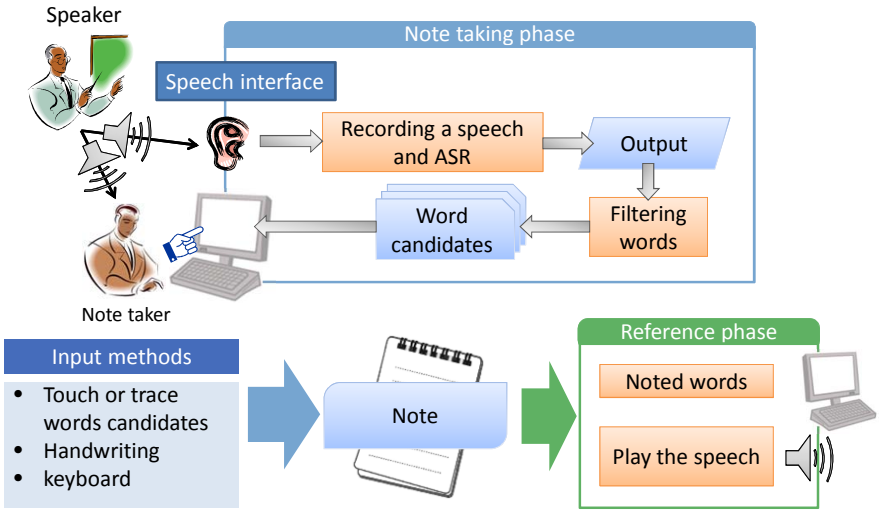
A. System outline

Figures 1 and 2 show the processing flow and a screenshot of our note-taking system, respectively. As a user on the system listens to a speech, the speech is automatically recognized by ASR. The word sequences generated by the ASR system are filtered by rules and displayed on the screen. The user can take a note by simply touching or tracing the words on the screen. Therefore, handwriting and pushing keyboard buttons are unnecessary. The speech is also recorded in the system, and the locations of recognized words in the speech are identified by using word-to-speech alignment information produced by the ASR. Therefore, users can easily play the speech from a chosen point. It is not necessary to listen to the entire speech when the user looks up a note.

B. Speech interface

Our note-taking support system has a speech interface. This speech interface always captures a speech that a user hears and takes notes, and performs speech recognition on it.

The recognized words are displayed on the screen. The user can take a note by simply touching or tracing the relevant words with his/her finger. Therefore, this reduces the burden on the user when he/she takes a note while hearing the speech.



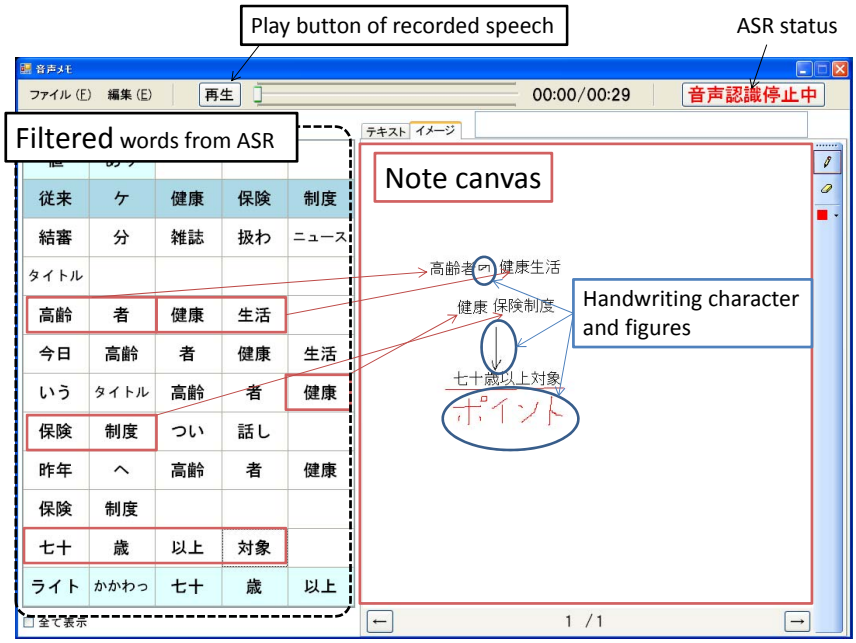


Fig. 2. Screenshot of note-taking support system.

However, this system depends on the speech recognition performance. If the words that the user wants to take note are not correctly recognized, the user cannot note them. Speech recognition errors are unavoidable. Therefore, it is undesirable to completely trust the ASR technology.

The key concept of our system is to use the ASR as an accessory function. The ASR helps system users to take notes, but it must disturb their work as little as possible. If the words that the user wants to note are not correctly recognized, the user does not need to touch or trace the words on the screen. Instead, the user can take a note using a keyboard or handwrite them with an electronic pen. Our concept of using the ASR as an accessory function differs from other systems with speech interface. The usability of the systems depends entirely on an ASR system's output, and the system's usability becomes worse if the ASR performs poorly.

A speech is recorded simultaneously with the ASR. Therefore, users can listen to the speech many times. Each noted word has location information that can describe where the word is located in the speech. Therefore, users can easily play the speech beginning at a specified word.

C. *Word filtering*

Figure 3 shows an example of the process for displaying word candidates from the ASR result.

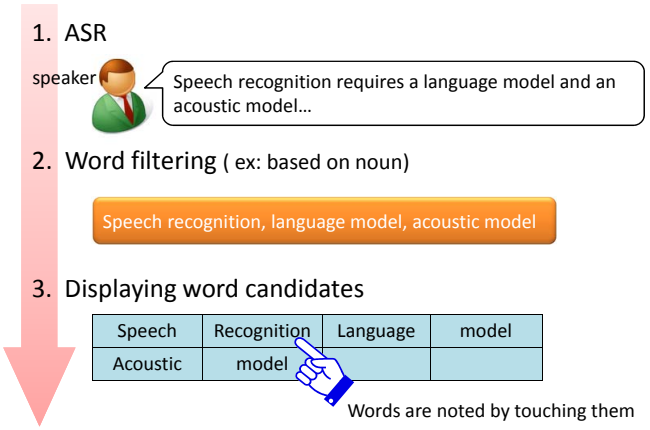


Fig. 3. Example of the process used to display word candidates from ASR output.

Out of all of the recognized words that are displayed on the screen, some could not be viewed easily. It may be difficult for users to find the necessary words among all the recognized words, some of which are incorrectly recognized. Therefore, all the recognized words are filtered on the basis of their part-of-speech (POS) information. In POS-based filtering, users can set a POS list containing POS names that they want to filter out.

As shown in Fig. 3, only filtered words are displayed on the screen. In this example, only noun words are extracted from the ASR output.

D. Keyboard and handwritten input

As described in SectionII-B, a user can take a note using a (hardware or software) keyboard or handwrite with an electric pen in addition to the ASR. If displayed words on the screen are faultily transcribed by the ASR, the user may input the words using a keyboard or handwriting.

As shown in Fig. 2, users can draw arbitrary-shaped graphics such as arrows on the screen. In addition, users can also write words and circle finger-touched or handwritten words to emphasize them.

In this paper, all the handwritten and keyboard-input words are called objects. If an object is written on the system's screen, it is correlated with the relative time from the beginning of the recorded speech. Therefore, if users touch an object, they can listen to the speech starting from the object-specified time.

E. Note reference

Users can see the notes taken by them while listening to a recorded speech when they want to refer to them. As mentioned in Sections II-B and II-D, users can also play the speech beginning at the time specified by the location time of the word or object the user focuses on. This helps the user to effectively refer to the note and is a very useful function for system users.

All the recognized words from the speech are also stored in the system in addition to the notes taken by the user. This can enable the user to search for a specified word from the recorded speech. When the user enters a word for which he/she wants to search in the search window, the word's locations are identified if it is correctly recognized.

However, we can never search for words transcribed incorrectly from a speech. Therefore, we are going to develop a search method such as spoken term detection [5], which is robust to speech recognition errors.

III. System Evaluation

To evaluate our system, we performed a multiple-subject experiment in which the effectiveness of note-taking work was evaluated for each subject. In addition, we asked the subjects to answer a questionnaire related to system usability and other characteristics.

A. Experimental setup

The note-taking support system was developed using a Visual C# development environment as an application that runs on a Windows 7 Professional laptop or PC with a touch panel.

In the system, we used a Julius decoder [6], version 4.2. as an ASR system. An acoustic model, the tri-phone-based hidden Markov model with 25-dimensional mel-frequency cepstrum coefficient-based features, was trained from the Corpus of Spontaneous Japanese (CSJ) [7]. A language model, a word-based trigram with a 43k vocabulary, was also trained from transcriptions in the CSJ.

In the experiment, subjects took notes using two note-taking methods: our system and a text editor on a PC with a keyboard as hardware. We compared the effectiveness of the note-taking work using our system with that using the text editor, which is a typical note-taking method.

All the subjects were good at keyboard operation because they were students in the information science course at our university. The number of subjects was 10.

B. Multiple-subject experiment

We prepared two speeches (Speech A and Speech B), and the 10 subjects took notes while hearing these speeches using one of the note-taking methods. Both speeches were lectures in the CSJ. Five subjects took notes for Speech A using the system and for Speech B using the text editor with a keyboard. The other subjects took notes for Speech B using the system and for Speech A using the text editor.

One week after finishing the note-taking work, all the subjects answered questions on the content of the speeches (five questions per speech). Of course, the subjects were allowed to listen to the recorded speech when they answered the questions. When they referred to the notes taken in the text editor, the subjects listened to the speech using an IC recorder. We measured the time until a subject answered all the questions perfectly for a speech.

The ASR performance for both speeches was about 80%. All the subjects received an explanation of how to take a note using the system and to refer to notes from the system. In addition, they practiced using the system.

TABLE I
TIME REQUIRED TO ANSWER ALL THE QUESTIONS FOR EACH SUBJECT. [MM'SS"]

Subjects No.	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
System	16'42"	6'46"	18'50"	9'43"	11'11"	9'50"	8'55"	10'10"	9'51"	8'28"
Text-editor	4'56"	13'10"	13'45"	10'55"	13'20"	6'40"	13'16"	17'59"	10'28"	8'55"

TABLE II
AVERAGE TIMES REQUIRED TO ANSWER QUESTIONS FOR EACH SPEECH. FIGURE IN PARENTHESES SHOWS THE NUMBER OF SUBJECTS WHO ALWAYS ANSWERED FASTER WITH THE SYSTEM THAN WITH THE OTHER METHOD.

Speech type	Speech A	Speech B
System	12'38" (3)	9'26" (4)
Text-editor	11'27" (2)	11'13"(1)

TABLE III
QUESTIONNAIRE RESULTS FOR THE SYSTEM (AVERAGES FOR ALL SUBJECTS).

Items	note-taking	reference work
(1) handling	3.6	4.1
(2) user-friendliness	3.8	4.1
(3) ASR errors	3.9	—

C. Evaluation results

TABLE I shows the time required to answer all the questions for each subject, and TABLE II shows the average time for each speech. As shown in TABLE II, the average answering time for the five subjects using the text editor for Speech A was shorter than that for subjects using the system. However, three subjects using the system answered faster than those using the text editor. On the other hand, the average answering time using the system for Speech B was shorter than that using the text editor.

To summarize TABLEs I and II, seven subjects using the support system finished answering faster than those using the text editor, whereas the other three did not. Because subject #1 fortuitously made perfect notes using the text editor, he could submit perfect answers very quickly. On the other hand, subjects #3 and #6 were not accustomed to the system operation, and therefore, they could not finish answering the questions quickly. Thus, we expect that if users become accustomed to system operation, then they should be able to take notes faster using the system.

TABLE III shows the questionnaire results related to the support system. All the subjects answered the questionnaire, which evaluated the following three items: (1) ease of handling, (2) user friendliness, and (3) influence of ASR errors. Each item was evaluated on five-level scale (1-5) in which bigger numbers indicate more positive evaluations.

In addition, we asked the subjects which note-taking method made their note-taking operation comfortable. The results are shown in TABLE IV. Most of the subjects chose the support system for both taking a note and referring to the note.

These results suggest that the note-taking support system developed by us may be useful for note-taking operation.

TABLE IV
SUBJECTS' PREFERENCES FOR TWO NOTE-TAKING METHODS.

	system	text editor	neither
taking note	8	0	2
reference	8	1	1

IV. CONCLUSION

This paper proposed a note-taking support system with a speech interface. The speech interface always captures a speech and performs ASR. The words of the speech, which are filtered using rules set by a system user, are automatically displayed on the system's GUI. The user can take a note simply by touching and tracing the words. In addition, the user can also write other words or figures using a keyboard or a touch pen. These interfaces made the system users' note-taking work comfortable. Furthermore, users could play the recorded speech beginning at user-specified words when they referred to a note using the system. In an experiment, most of the subjects gave our system a good evaluation. Therefore, the developed system may be useful for note-taking work.

In the future, we are going to improve the word search module, which can search for user-specified words in a recorded speech, like spoken term detection technology. We also plan to develop a system that works on tablet-type computers or smart-phone devices to enhance system portability.

REFERENCES

[1] Y. Akita, M. Mimura, G. Neubig, and T. Kawahara, "Semi-automated update of automatic transcription system for the Japanese national congress." in Proc. of INTERSPEECH2010, pp.338–341, 2010.

[2] T. Kawahara, N. Katsumaru, Y. Akita, and S. Mori, "Classroom Note-taking System for Hearing Impaired Students using Automatic Speech Recognition Adapted to Lectures." in Proc. of INTERSPEECH2010, pp. 626–629, 2010.

[3] K. Kurihara, M. Goto, J. Ogata and T. Igarashi, "Speech Pen: Predictive Handwriting based on Ambient Multimodal Recognition," in Proc. of ACM SIGCHI Conference on Human Factors in Computing Systems(CHI'06), pp.851-860, 2006.

[4] T. Kamihira, H. Nishizaki, Y. Sekiguchi, R. Kurakane, K. Nishizaki, and H. Ikegami, "Development of hospital appointment system with user-friendly speech interface," in Proc. of the 2nd Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC 2010), pp. 490–493, 2010.

[5] S. Natori, H. Nishizaki, and Y. Sekiguchi, "Japanese spoken term detection using syllable transition network derived from multiple speech recognizers' outputs," in Proc. of INTERSPEECH2010, pp. 681–684, 2010.

[6] A. Lee and T. Kawahara, "Recent Development of Open-Source Speech Recognition Engine Julius," in Proc. of the 1st Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC2009), 6 pages, 2009.

[7] K. Maekawa, "Corpus of Spontaneous Japanese: Its design and evaluation," in Proc. of the ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition (SSPR2003), pp. 7–12, 2003.