Voice Input Tutoring System for Older Adults using Input Stumble Detection

Toshiyuki Hagiya
KDDI Research, Inc. &
Kyoto University
to-hagiya@kddi-research.jp

Keiichiro Hoashi
KDDI Research, Inc.
hoashi@kddi-research.jp

Tatsuya Kawahara
Kyoto University
kawahara@i.kyoto-u.ac.jp

ABSTRACT

Many older adults are interested in smartphones but encounter difficulties in self-instruction and need support, especially text input. Voice input is a useful option for text input, but also presents some difficulties for older adults. In this paper, we propose a tutoring system for voice input that detects input stumbles using a statistical approach and provides instructions to overcome them. We construct the tutoring system based on the data from a user study with novice older adults. In an evaluation experiment, the number of input stumble and the sentence completion time of the participants using the tutoring system were significantly smaller than those without it. The results showed that the tutoring system resulted in the improvement of the efficiency of voice input for novice older adults.

Author Keywords

Voice input; older adults; tutoring system;

INTRODUCTION

Smartphones offer new opportunities to improve the lives of older adults [8]. Although these individuals would like to learn how to use smartphones [1], those who have never used one may face difficulties and some of them may give up on using a smartphone and go back to using their old feature phone, Therefore, providing support in the initial stages is very important. To make full use of the functions of a smartphone, it is essential to master text input, which is one of the operations that novice older adults find most difficult. Automatic speech recognition (ASR), which is an option of the text input, are easier to those who were unfamiliar with touch input [12]. However, users need to get accustomed to ASR, in order to input as intended. In addition, ASR needs to be used along with software keyboards when precisely input sentences are required, e.g., when dealing with documents. In this study, we developed a tutoring system for voice input that detects input stumbles using a statistical approach, and then provides instructions that help users resolve input stumbles independently. We outline our development of the tutoring system in three steps. First, we describe a user study that clarified the problems that older adults encounter with voice input. Second, we explain designing the structure of the tutoring system on the basis of the user study. Finally, we evaluate the performance of the resulting tutoring system.

RELATED WORK

Interface Design for Voice Input

At present, automatic speech recognition (ASR) is still imperfect, although many researchers have worked to improve its performance. Therefore, some research has been proposed for voice input interfaces to include the assumption that misrecognition occurs. Goto *et al.* [4] proposed a system to alter a sentence when users hesitate by lengthening a vowel during a phrase. Ogata *et al.* [11] proposed a system to display the result of ASR as a sentence that includes alternative word candidates. In a similar way, Liang *et al.* [9] proposed a simple gesture-based error correction interface where a user marks the error region once, and then the error region is replaced by the top candidate. Speech Recognizer for Android [2] gives feedback on the results of errors, such as audio recording errors and network-related errors. This function mainly gives feedback on the cause of equipment-side rather than user-side problems.

Tutoring System for Older Adults

A wealth of research has focused on designing better instructional resources to assist older adults. For example, Morrell *et al* [10] have studied what the optimal amount of guidance is. Rogers *et al.* [13] investigated the kind of resources most useful in the learning process, and found that step-by-step interactive tutorials were the most effective approach to the learning process for older adults. With respect to using smartphones, Leung *et al.* [8] surveyed and investigated how older adults learned. According to their report, older adults tend to prefer an instruction manual to trial-and-error. Kelleher *et al.* [6] proposed stencil-based tutorials that overlay step-by-step instructions on the screen. Hagiya *et al.* [5] proposed a tutoring system for text entry on smartphones that provided instruction about the next operation. However, tutoring systems for voice input have not been studied.

TUTORING SYSTEM FOR VOICE INPUT

As described in the previous section, although many studies deal with ASR and others with tutoring systems, tutoring systems for voice input have not been studied. Therefore, we propose a tutoring system for voice input that detects input

IUI 2018, March 7–11, 2018, Tokyo, Japan.
Copyright © 2018 ACM ISBN 978-1-4503-4945-1/18/03 ...$15.00.
https://doi.org/10.1145/3172944.3172995

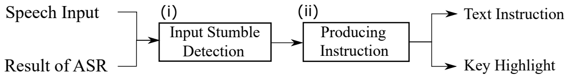


Figure 1. Block diagram of the voice input tutoring system

stumbles using a statistical approach and provides instructions that help users resolve input stumbles. As shown in Figure 1, the tutoring system is composed of two functions, (i) input stumble detection and (ii) producing instruction. The function of input stumble detection is to detect voice input stumble based on a model trained by features from such as acoustics and results of ASR. The function that produces instruction displays the instructions for the next operation, using text and key highlighting.

USER STUDY TO CLARIFY HOW NOVICE OLDER ADULTS MAKE INPUT STUMBLES

This study has two purposes: One is to clarify how novice older adults make input stumbles in smartphone voice input. Second is to collect the input data to construct a tutoring system. In this study, participants first input a statement by voice, and then are permitted to modify what was recognized, using a combination of voice and touch.

Participants

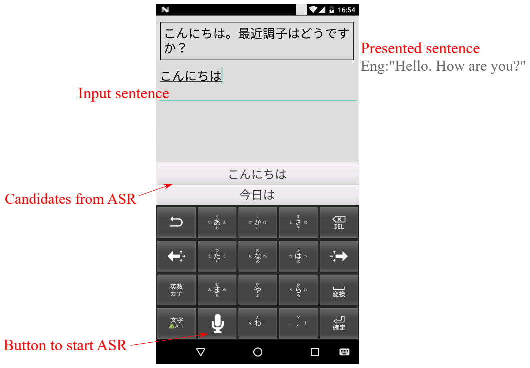
Ten older adults were recruited from a local social institution to take part in this experiment. There were five males and five females, ranging in age from 65 to 72 years, with a mean age of 69.8 (sd = 3.8). None of them owned a smartphone but all had owned a featurephone and had used their own PC routinely. None of them had tremor disorders, eye problems, or other relevant health problems.

Apparatus

Google Cloud Speech API [3] was used for ASR. As shown in Figure 2, a key for ASR was implemented on a software keyboard. After the key is pushed, a tone acknowledges the push, and the end of the tone announces the start of ASR. ASR terminates when the API detects the end of a speech or when five seconds have passed without input. Then Google Cloud Speech API outputs the N best candidates of recognition results and the highest confidence score between 0 and 1. The smartphone records the user's voice and the keys touched. All operations were also recorded by an overhead video camera. Participants used a Nexus 6 running Android 7.0.

Procedure

First, an overview of the experiment was explained and the informed written consent was obtained from the participants. Next, the participants were given explanations on how to operate a smartphone, including touch and swipe operations, and instructed on how to use the voice input and the software keyboard by a human tutor. Then, they input sentences to match a sentence presented in Japanese, using an application as shown in Figure 2. The sentences were selected from an email corpus collected originally from personal conversations via email. The corpus contained roughly 30,000 sentences (average character length = 23.6). Participants were asked to input twenty sentences within 60 minutes, with a ten-minute break after completing ten sentences. They operated the smartphone while holding it in their hand and sitting on a chair. They were instructed to type by themselves if possible. However, they were permitted to ask the human tutor when they lacked the confidence to perform the next action. After the experiment, they took part in an interview.


こんにちは。最近調子はどうですか？

こんにちは

こんにちは

今日は

Figure 2. A screen of an input application and a software keyboard

Table 1. Input stumble of voice input

	Input stumble
(1)	Volume was too high for voice recognition
(2)	Volume was too low for voice recognition
(3)	Utterance started before ASR began working
(4)	Filler was inserted
(5)	Utterance paused due to long phrase
(6)	Intended homonyms were not displayed due to short utterance
(7)	Not knowing how to insert punctuation mark
(8)	Not knowing how to insert question mark
(9)	Not knowing how to select candidates
(10)	Not knowing how to delete a word
(11)	Forgetting to move cursor to end of sentence

Upon completion of this stage, three annotators independently extracted the patterns of input stumbles from the logs and the recorded videos, and defined categories of the stumbles based on discussion. Next, the annotators associated labeled input stumbles of each input sentence, and the labels given by at least two annotators were used in the next stage.

Results of the User Study

All participants completed the task, taking an average time of 27.3 minutes (sd = 7.4). Voice input stumbles fell into 11 categories, as shown in Table 1. The concordance rate of the annotation of the input stumble by Fleiss' kappa was 0.79. In the interview, many comments similar to these were made:

> *"It was easier to input by voice than touch, but it was difficult to correct it when I made mistakes. So I wanted to know how I should correct and why I failed."*

This study showed that older adults have some difficulties for voice input and need instructions.

CONSTRUCTION OF TUTORING SYSTEM

In this section, we describe the construction of the two functions of the tutoring system for voice input.

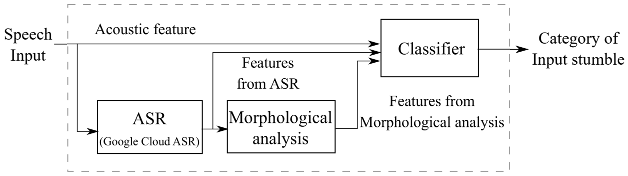


Figure 3. A block diagram of input stumble detection

Input Stumble Detection

Input stumble detection was constructed by machine learning based on data from the previous user study. The block diagram of input stumble detection is shown in Figure 3. First, acoustic features were extracted from speech input. Simultaneously, features from ASR including the confidence score and the candidates were the output from Google Cloud ASR. Next, the features from the morphological analysis of the results of ASR were extracted using Lucene-gosen [7]. Finally, the classifier of input stumbles was trained using the features and the labels.

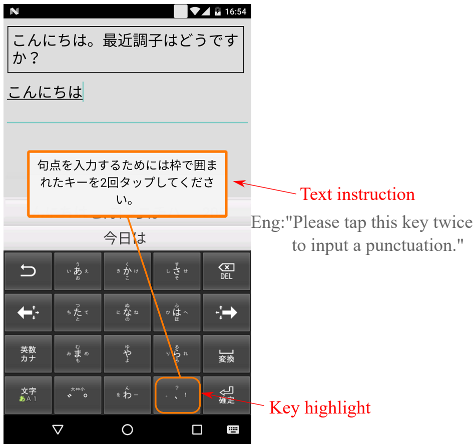
To select the best machine learning algorithm, we compared the performance of four models, a linear regression, a C4.5, a support vector machine (SVM) using an RBF kernel, and a deep neural network (DNN) [14]. A DNN with five layers was trained by minimizing loss function with backpropagation and the stochastic gradient descent method with a dropout ratio of 50%.

A total of 28 features was used, 17 of which were acoustic features: the number of amplitude overflows, the duration of voice activity, five kinds of amplitude averages, namely time gaps between voice activity, 0.5 sec before and after the start and the end of voice activity, and the differences in the five kinds of averages. The remaining 11 features were textual features related to the ASR and morphological analysis results: the number of candidates, the confidence score, the length of the sentences, the number of morphemes, the ID of morphemes of sentence terminations, the number of conjunctions, the number of end-forms, the number of fillers, the number of homonyms of the shortest words (except particles), the time to the next operation, and the position of the cursor.

Features used for SVM and logistic regression were selected by L1 regularization. Those for a C4.5 were selected by stepwise backward selection of each participant with 10-fold cross-validation (CV). The F-measure from the 10-fold CV was 0.70 with a linear regression, 0.73 with C4.5, 0.72 with SVM and 0.73 with DNN. Anova（$\alpha = 0.05$）showed no significant differences among these models, so we adopted C4.5 for use here as it had the highest rate of accuracy.

Producing Instructions

The definition of instructions to be presented to users was done manually, based on the observation of effective instructions given by human tutors. The instruction was provided by text and with overlaid key highlighting. The instruction provided for the input stumble of “Not knowing how to insert a punctuation mark” is exemplified in Figure 4. The instruction was displayed when there was no operation for 5 seconds after a detected input stumble, and remained on the screen for


16:54

こんにちは。最近調子はどうですか？

こんにちは

句点を入力するためには枠で囲まれたキーを2回タップしてください。

今日は

Figure 4. Screen shot showing instructions after input stumble (7)

7 seconds or until the screen was touched. The time and the text of instructions were modified after discussions with three older adults (60s) who did not participate in this study. The texts were accordingly modified to show not only the method of operation but also the reason for the correction, as well as tips for voice input.

EVALUATION EXPERIMENT WITH TUTORING SYSTEM

Participants

Twenty older adults took part in this experiment. They were recruited from a local social institution and did not participate in the previous user study. There were ten males and ten females, ranging in age from 65 to 71 years, with a mean age of 68.6 (sd = 1.9). Ten participants did not use the tutoring system (group A), and the other ten used it (group B). None of these older adults owned a smartphone or had any previous experience of using a voice input, but all had owned a featurephone and all had routinely used their own PC. None of them had tremor disorders or other relevant health problems.

Procedure and Apparatus

The experiment was conducted in a similar way to the previous user study. In brief, an overview of the experiment was explained and the informed written consent was obtained from the participants. Next, the participants were given explanations on how to operate a smartphone and how to use the voice input and the software keyboard. While seated in a chair, they input 40 sentences to match the presented sentences. Participants first used voice input, then were permitted to modify the result with a combination of voice and touch. They had a ten-minute break after completing twenty sentences. After the experiment, they filled out a 5-point Likert-scale questionnaire, and took part in an interview. After all the participants had completed the experiment, three annotators labeled input stumbles in the logs. We adopted the labels given by more than two annotators.

Experimental Results

All participants completed the task. The concordance rate of annotation of input stumble using Fleiss’ kappa was 0.85.

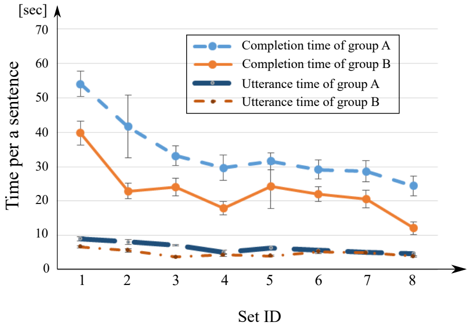


Figure 5. The average completion time and the utterance time for a sentence in each set

The detection performance of input stumble for group B using F-measure was 0.74. The word recognition rate of ASR was 91.1% for group A and 92.6% for group B, respectively. Welch's t-test ($\alpha = 0.05$) showed no significant difference between groups A and B. To evaluate the effect of the tutoring system, we adopted three metrics: the task completion time, number of input stumble, and the questionnaire.

Completion Time

To analyze the learning effect of our system, we conducted a chronological analysis of the performance of the subjects, by dividing all 40 sentences into 5 sets (8 sentences per set), and measuring the average completion time and utterance time for each set during the experiment. The results of the analysis are shown in Figure 5. The completion time of both groups decreased gradually. The completion times of group B were shorter than those of group A in all sets. Welch's t-test showed a significant difference between the completion time of both groups for the first set ($t(12) = 2.79, p < .05$) and the final set ($t(15) = 2.89, p < .05$). On the other hand, the utterance time of both groups decreased gradually, and there was no significant difference between the two groups.

Number of input stumbles

Figure 6 shows the number of input stumbles per sentence. As shown in the figure, the number of input stumbles of both groups decreased gradually, as did the completion time. The number of input stumbles in group B was less than in group A in all sets. Welch's t-test showed a significant difference between the number of stumbles of both groups for the first set ($t(16) = 3.05, p < .01$) and final set ($t(9) = 8.65, p < .01$).

Questionnaire results

The results of questionnaires are shown in Table 2. The scale is 1-5, where 5 indicates the highest level of agreement. The score of the items *"The voice input was useful"* were 3.7 in group A and 3.6 in group B, showing no significant difference. On the other hand, there was a 0.5 point inter-group score difference for *"The correction was easy"* and *"I can learn the voice input method by myself"*. A Wilcoxon signed-rank sum test ($\alpha = 0.05$) showed a significant difference between the groups for the only latter item.

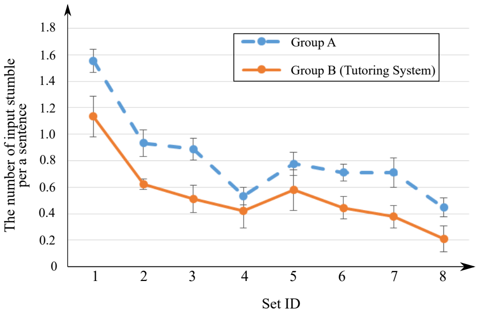


Figure 6. The number of input stumbles in a sentence in each set

Table 2. Questionnaire responses. The scales are 1-5, where 5 is higher agreement; the standard deviations are given in parentheses)

Question	group A	group B
The voice input was useful	3.7 (0.64)	3.6 (0.66)
The correction was easy	2.6 (0.66)	3.1 (0.70)
I can learn the way of the voice input by myself	3.7 (0.45)	4.2 (0.40)

DISCUSSION

In summarizing the results of the evaluation, the completion time and the number of input stumbles of group B were significantly smaller than those of group A. These results indicate that the tutoring system works effectively for novice older adults. Focusing on the time, the time taken to find out how to modify and the time to needed to modify sentences decreased as a result of using the system because the ratio of the utterance time to the completion time was small. In addition, there is a possibility that the amount of modification decreased because instructions influenced the input efficiency. Some participants of group B made a comment:

"I realized that ASR worked well when I want to input a long sentence or several sentences if I input each short sentence."

These comments imply that the instruction provided by the tutoring system worked effectively.

On the other hand, some participants made multiple input stumbles simultaneously in one voice input. However, the tutoring system provides an instruction only for the most probable input stumble. Therefore, in this case, they often did not know how to perform the next operation after resolving an input stumble. To detect multiple stumbles and provide step-by-step instruction is a topic for our future works.

CONCLUSION

We developed a tutoring system for voice input that detected input stumbles, and provided instructions. First, we conducted a user study to clarify the problems older adults encountered in voice input. Second, we constructed the tutoring system. Finally, we evaluated the performance of the tutoring system. In the evaluation experiment, the number of input stumble and the sentence completion time of the participants using the tutoring system were significantly smaller than those without it. The results showed that the tutoring system resulted in improvement in the efficiency of voice input for novice older adults. In future research, we are going to develop a system to detect multiple stumbles and provide step-by-step instructions to improve usability.

REFERENCES

1. Beverly Beisge and Marilyn Kraitchman. 2003. Senior Centers: Opportunities For Successful Aging. *Springer Publishing Company* (2003).
2. Google. 2017a. Android Developers. (2017). https://developer.android.com/index.html.
3. Google. 2017b. Cloud Speech API. (2017). https://cloud.google.com/speech.
4. Masataka Goto, Katunobu Itou, and Satoru Hayamizu. 2002. Speech Completion: On-demand Completion Assistance Using Filled Pauses for Speech Input Interfaces. In *Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP '02)*. World Academy of Science, Engineering and Technology (WASET), 1489–1492.
5. Toshiyuki Hagiya, Toshiharu Horiuchi, and Tomonori Yazaki. 2016. Typing Tutor: Individualized Tutoring in Text Entry for Older Adults Based on Input Stumble Detection. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 733–744.
6. Caitlin Kelleher and Randy Pausch. 2005. Stencils-based tutorials: Design and evaluation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '05)*. ACM, 541–550.
7. Taku Kudoh, Tsuyoshi Fukui, Takashi Okamoto, and Matt Francis. 2017. lucene-gosen. (2017). https://github.com/lucene-gosen.
8. Rock Leung, Charlotte Tang, Shathel Haddad, Joanna Mcgrenere, Peter Graf, and Vilia Ingriany. 2012. How older adults learn to use mobile devices: Survey and field investigations. *ACM Transactions on Accessible Computing* 4, 3 (2012), 11:1–11:33.
9. Yuan Liang, Koji Iwano, and Koichi Shinoda. 2014. Simple Gesture-based Error Correction Interface for Smartphone Speech Recognition. In *Proceedings of the 15th Annual Conference of the International Speech Communication Association International Conference on Science and Technology for Humanity (INTERSPEECH '14)*. International Speech Communication Association (ISCA), 1194–1198.
10. Roger W. Morrell, Denise C. Park, Christopher B. Mayhorn, and Catherine L. Kelley. 2000. Effects of age and instructions on teaching older adults to use eldercomm, an electronic bulletin board system. *Educational Gerontology* 26, 3 (2000), 221–235.
11. Jun Ogata and Masataka Goto. 2005. Speech Repair: Quick Error Correction Just by Using Selection Operation for Speech Input Interfaces. In *Proceedings of the 9th European Conference on Speech Communication and Technology (Eurospeech '05)*. International Speech Communication Association (ISCA), 133–136.
12. Pei-Luen Patrick Rau and Jia-Wen Hsu. 2005. Interaction Devices and Web Design for Novice Older Users. *Educational Gerontology* 31, 1 (2005), 19–40.
13. Wendy A. Rogers, Elizabeth F. Cabrera, Neff Walker, D. Kristen Gilbert, and Arthur D. Fisk. 1996. A survey of automatic teller machine usage across the adult lifespan. *Human Factors* (1996), 38,1; 156–166.
14. Yuan Tang. 2016. TF.Learn: TensorFlow's High-level Module for Distributed Machine Learning. In *arXiv*. 1612.04251.