Efficient Speech-Recognition Error-Correction Interface for Japanese Text Entry on Smartwatches

Ryotaro Toba
Doshisha University
Kyoto, Japan
ctwd0145@mail4.doshisha.ac.jp

Tsuneo Kato
Doshisha University
Kyoto, Japan
tsukato@mail.doshisha.ac.jp

Seiichi Yamamoto
Doshisha University
Kyoto, Japan
seyamamo@mail.doshisha.ac.jp

ABSTRACT

We propose an efficient speech-recognition error-correction interface for Japanese text entry on smartwatches. Although the accuracy of automatic speech recognition (ASR) has significantly improved, an interface for text modification is still essential. Considering the strict limitation of a narrow display area and practical demand of text modification, the proposed interface arranges the N-best results of ASR and a list of morphemes consisting of the 1-best result to enable quick access to any word to be modified. Specifically, multiple screens of the N-best results are switched by horizontal flicks, and another extended screen listing a morpheme sequence of the 1-best result is scrolled by vertical flicks. The proposed interface was compared with a software keyboard and a speech-input-enabled input method editor (IME), which was a simple combination of speech input and software keyboard. The proposed interface outperformed the other two interfaces in terms of time required to complete specified sentences, subjective score using system usability scale (SUS), and perceived workload quantified using the NASA Task Load Index (NASA-TLX).

CCS CONCEPTS

• Human-centered computing → Human computer interaction (HCI) → Interaction techniques → Text input

KEYWORDS

Text entry; Smartwatch; Speech input; Error correction;

MobileHCI '19, October 1–4, 2019, Taipei, Taiwan
 ACM ISBN 978-1-4503-5970-2/19/05...$15.00.
DOI: https://doi.org/10.1145/3338286.3340124

INTRODUCTION

Smartwatches are mainly used in passive ways, such as health monitoring and receiving notifications. However, when text entry becomes easier, they will also be used in more active ways, such as for short messaging, prompt response and web search in motion. Speech input is a major text-entry method on smartwatches due to easy, fast and hands-free usability. ASR outputs the most probable sequence of words given a speech signal using an acoustic model, a word dictionary and a language model. While the accuracy of the state-of-the-art ASR technology has significantly improved with advances in deep learning technologies, a user interface for text modification is still necessary. Even if the ASR result is perfect, users often find that some part of the entered text needs to be better modified. The problem is that the character-level correction with a software keyboard requires precise manipulation on a small touch screen, while massive amounts of ASR N-best results are sometimes helpful for correction, but not always.

Various user interfaces for correction of ASR-based text entry have been proposed. Most of them utilize probable segments of ASR results. Speech Repair [5] is a concurrent user interface that enables a user to correct transcription by choosing words from an ASR confusion network on a PC display. Parakeet [8,9] implements a comprehensive set of corrective gesture operations on a handheld mobile device based on a similar word-confusion network. Speech Dasher [7] is an extended interface that allows text to be corrected by choosing letters through eye gaze. These interfaces are well-designed and should work properly with a wide display. However, smartwatches do not have displays wide enough to show the confusion network or a list of letters. Some studies proposed automatic correction of an ASR result by enabling a user to simply mark positions and types of errors. One example is an easy handwriting interface for correcting text by listing alter-

(a)Speech input screen. **(b)N-best result screens.** **(c)Morpheme-level correction screen.**

Figure 1. Screens of proposed interface.

native N-best candidates based on template constrained posterior [10]. Another example is automatic error correction considering long context match [12,13]. These interfaces alleviate the workload for text correction. Furthermore, speech interfaces for correcting erroneous parts of text entered by speech input have been proposed. One such interface uses one-step error detection and correction by speech input [6]. The ASR of this interface detects a user's intention to correct an error during transcription. However, a space-efficient interface for text modification enables precise editing of the entered text.

We propose an efficient text-modification interface for smartwatches which takes into account the small display area and practical demand for text modification. While ASR is generally accurate enough in quiet environments, correctly inserting punctuation marks is difficult. All errors are made not in characters but in words by ASR processing based on a language model: however, texts in Japanese and some other Asian languages such as Thai and Chinese are not segmented by spaces as in European languages. Users have to set the cursor at the position to be modified by precise pointing on the small touch display or repetitive cursor operations on a long sequence of characters. Furthermore, Japanese has many homophones with different *Kanji* (Chinese characters used in Japanese), especially for proper nouns. Users always have to choose the correct *Kanji* from a list of alternatives or correct the *Kanji* by re-entry. The proposed interface firstly lists the N-best results of ASR. When the desired text is not found, a list of the morpheme sequence of the 1-best result is provided by segmentation with a Japanese morpheme analyzer. It enables quick access to any word to be corrected with high viewability and operability. Framing the problem in a more generic way, the small display area poses a trade-off between viewability and accessibility of ASR results. The proposed interface eliminates the trade-off with multiple screens of N-best results and an extended screen for listing the morpheme sequence of the 1-best result. Both screens are easily accessible by flick operations.

INTERFACE DESCRIPTION

The proposed interface first allows a user to enter text through speech input and then modify the text through touch operation if a misrecognized or an incorrect part of the text is found. The interface enables quick modification with two types of screens. One enables a user to select the N-best results of ASR, and the other enables the user to quickly access any word of the 1-best result by flicking a list of morphemes into which the Japanese morpheme analyzer segmented the 1-best result. The proposed interface consists of the following four types of screens.

- Speech-input screen - A user first enters a text through speech input with the screen shown in Figure 1 (a). When the microphone is on, the screen shows an animation indicating ASR is processing. The recognition result is displayed on the screen soon after ASR is completed. We use an Android standard API "Speech Recognizer" [2].
- N-best result screens - When a misrecognized text is found, the user can search for the correct text in the N-best screens, one of which is shown in Figure 1 (b). A user can switch to the screens by tapping the button at the bottom right on the speech-input screen. Each of the screens shows one of the 2nd to 4th best results, and the user can choose one by tapping that screen. The number of N-best results displayed is limited to three so as not to force the user to check too many screens, although a number of N-best results are obtained from the ASR engine. The user can switch screens by flicking right or left.

(a)Google's Japanese keyboard. **(b)Speech-input-enabled IME.**

Figure 2. Interfaces for comparison.

Table 1. Japanese kana syllabary table.

あ a	か ka	さ sa	た ta	な na	は ha	ま ma	や ya	ら ra	わ wa
い i	き ki	し si	ち ti	に ni	ひ hi	み mi		り ri	
う u	く ku	す su	つ tu	ぬ nu	ふ hu	む mu	ゆ yu	る ru	を wo
え e	け ke	せ se	て te	ね ne	へ he	め me		れ re	
お o	こ ko	そ so	と to	の no	ほ ho	も mo	よ yo	ろ ro	ん n

If the correct text is not found from the N-best result screens, the user can switch the screen to the morpheme-level correction screen by tapping the button at the bottom of the screen.

- Morpheme-level correction screen – The user can correct the text of the 1-best result from the screen shown in Figure 1 (c), when the correct text is not found on the N-best result screens. This screen lists morphemes of the 1-best result and enables the user to quickly access any of them through vertical flicks. The 1-best result is automatically segmented into morphemes by using a Japanese morpheme analyzer. The user can choose a morpheme to be corrected by tapping it, and a Japanese software keyboard appears for text modification. Unlike the case of keyboard-based text entry, all ASR errors are made not based on characters, but on words (nearly equal to morphemes), because ASR is processed based on a word dictionary and language model that provides probabilities to word sequences. Morpheme-level correction is reasonable in this sense and eliminates the need for repetitive cursor operations on the text-editing area.

  To further improve the operability of correction, we implemented two functions in addition to morpheme-level correction. The arrow buttons at the head of each morpheme in Figure 1 (c) insert a punctuation mark after the morpheme because the ASR results do not contain punctuation marks. A right flick on a morpheme deletes that morpheme. To insert text, the user taps a neighboring morpheme and edits the text. We use a network-based Japanese morphological analysis API provided by goo-labo [3].

- Keyboard screen: "Keypad-Flick" – Google's Japanese keyboard for smartwatches (Figure 2 (a)) is used for morpheme-level correction. This Japanese keyboard has a 3x4 keypad with 10 representative *kanas* (the Japanese syllabary characters) and symbols printed on the keys. Japanese *kana* is a syllabary character. One *kana* corresponds to one Japanese syllable, which is basically composed of a consonant and a vowel (CV), or only a vowel (V). Japanese sounds have five vowels 'a', 'i', 'u', 'e', o' and nine basic consonants. A total of 46 basic *kanas*, which comprise the five vowels, forty CV syllables and a nasal sound 'n' are allocated on a well-known Japanese *kana* syllabary table (Table 1) in 10 columns and 5 rows. The first column corresponds to vowels only (without a consonant), and the other nine columns correspond to the basic nine consonants. The rows correspond to the five vowels. In general, Japanese text, written using thousands of *kanji* characters and the *kanas*, is first entered with the *kanas*, and then converted to the standard text style by invoking a *kana-kanji* converter or a predictive converter. This keyboard is consistent with the most popular Japanese keyboard on smartphones. The keyboard assigns five kanas comprised of a specific consonant and either one of the five vowels to one key. The desired *kana* character is entered by either multiple tapping or flick operations. That is, a *kana* with vowel 'a' is entered by a single tap, and other four *kanas* with vowels 'i', 'u', 'e' and 'o' are entered by multiple taps or flicking to either one of four directions [14]. When a key is touched, as shown in Figure 2 (a), a flick guide that indicates five *kanas* on flick directions is displayed over the keypad. The seven keys surrounding the lower side of the keypad are used to 1) move the cursor leftward, 2) switch

character type modes, 3) enter symbols, 4) space, 5) backspace, 6) enter and 7) move the cursor rightward, from left to right. Furthermore, predicted word candidates are displayed over the keypad. The size of a key is 3mm in height and 7mm in width. We call this Japanese keyboard "Keypad-Flick" hereafter.

Two Baseline Conditions

We prepared two methods of text entry to conduct a comparative user study with the proposed interface. The first one is using only Keypad-Flick (Figure 2 (a)). Users are allowed to use all the product's functions including *kana-kanji* conversion and predictive conversion. The second one is using the speech-input-enabled IME shown in Figure 2 (b). The speech-input-enabled IME is a simple combination of speech input and Keypad-Flick. This interface enables a user to enter text through speech input and Keypad-Flick. The text-editing area in the center has a cursor to edit the text. Users locate the cursor by tapping on the display. The microphone button at the bottom left activates speech input. The globe button at the bottom right switches to Keypad-Flick, and a back-space button is placed between the other two buttons. In a user study, we asked participants to enter text with speech input first and then correct incorrect parts with Keypad-Flick so that they would not concentrate on either one of the two modalities.

USER STUDY

Evaluation method

We conducted a user study comparing the three interfaces. We asked participants to enter specified text exactly as shown on a sheet using each interface. Specifically, we requested them to enter the text without errors regarding homophones or missing punctuations, although the entered text was not checked, i.e., the system allowed uncorrected errors. The order of using the interfaces was randomized and counterbalanced across participants to eliminate the order effect. The performance of the interfaces was measured in terms of time required to complete one sentence. Subjective evaluation on usability was quantified using system usability scale (SUS) [4] and perceived workload was quantified using the NASA Task Load Index (NASA-TLX) [11].

Participants

The participants were ten university students ranging from 21 to 23 in age. All are male and native Japanese speakers. All owned a smartphone, but none owned a smartwatch. None had had an experience of operating a smartwatch before. Nine were right-handed. Seven usually entered text through flick operations on "Keypad-Flick" on their smartphones, one entered text through multi-tap operation on the same type of keyboard, and the other two entered text with a QWERTY keyboard.

Apparatus

The model of smartwatch used was HUAWEI WATCH 2, which has a watch face with a diameter of 30.5 mm and a resolution of 326 PPI (Pixel per Inch).

Phrase set

We composed 50 original Japanese sentences for evaluation (Appendix). The sentences were written in a plain style. Each sentence was composed of about 25 *kana* characters before the *kana-kanji* conversion and contained a couple of words that would need correction due to homophones and missing punctuation marks when entered using the ASR API. The ASR engine had been carefully tested by the experimenter if any utterance of the sentences causes an error that needs to be modified. In the evaluation, five sentences were randomly assigned from the 50 sentences to each of the three interfaces. The five sentences were mutually exclusive across the three interfaces, and differed from person to person.

Procedure

The user study was conducted in a quiet lab room where individual participants conducted their evaluations. After receiving a briefing, each participant was given a NASA-TLX practice session of four basic arithmetic operations to prepare for accurate measurement of perceived workload. After the practice session, the participant was given a 15- minute training session on the interfaces. The training session was for the participants to learn how to use the smartwatch and all three text-entry interfaces. Since all the participants were familiar with the "Keypad-Flick" interface through everyday use of their smartphones, they actually spent the 15-minute training time on adjusting to the smaller keypad and learning screen transitions and a few

Table 2. Mean and (standard deviation) of time required to complete one sentence, SUS score, and WWL score.

	Time (sec)	SUS score	WWL score
Keypad-Flick only	79.2 (21.7)	56.2 (25.8)	58.5 (18.9)
Speech-input-enabled IME	62.2 (11.0)	59.0 (13.5)	49.1 (22.8)
Proposed interface	51.4 (11.8)	69.6 (14.7)	27.9 (8.3)

additional functions of the added keys for easier text modification. After the training session, the participant started the evaluation session. The participant entered the five randomly-assigned sentences printed on a sheet by using the three interfaces in a specified order. After completing the text entry by using each interface, the participant rated usability on an SUS questionnaire and perceived workload on a NASA-TLX questionnaire. The whole evaluation procedure took less than an hour for all the participants.

Results

Text-entry time

The means and standard deviations of times required to complete a sentence with each interface (of the ten participants) are listed in Table 2. While "Keypad-Flick only" and "speech-input-enabled IME" took 79.2 and 62.2 seconds, respectively, the proposed interface took 51.4 seconds. We conducted an analysis of variance (ANOVA) test since the normality assumption was not rejected with a significant level at 0.05 in Shapiro-Wilk test. The results indicated a significant difference across the interfaces ($F_{(2, 8)} = 10.26$, $p < .05$). Bonferroni's post-hoc test showed that the difference between "speech-input-enabled IME" and the proposed interface was marginally significant with p = 0.057.

Figure 3 plots the time required to complete a sentence for each participant. Large gaps are observed between "Speech-input-enabled IME" and the proposed interface for participants #4, #6, #8, and #9, while small gaps are observed for participants #1, #2, #3, #5, #7 and #10. The large gaps were caused by using the N-best result screens for correction. To further investigate the

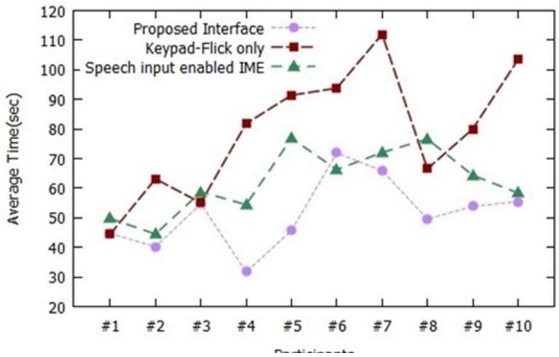

Figure 3. Time required to complete one sentence (sec) with each text-entry interface for each participant.

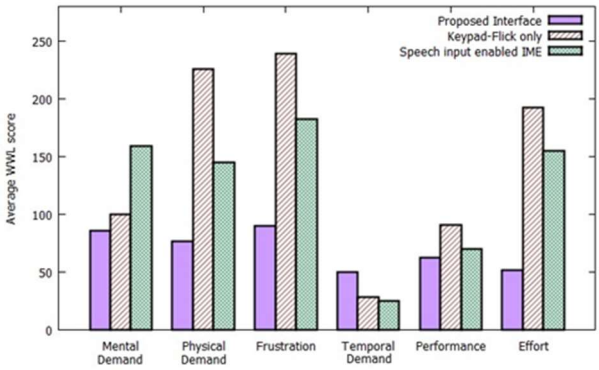

Figure 4. Average WWL score of six subjective subscales in NASA-TLX.

difference in the efficiency of selecting the N-best results and correction using Keypad-Flick, the time required to complete a sentence was tallied under two conditions: 1) corrected by choosing N-best results, and 2) corrected using Keypad-Flick. The time under condition 1) was 24.8 seconds, whereas that under condition 2) was 59.1 seconds. Correction by choosing one of the N-best results is obviously more efficient than that by using Keypad-Flick. The proposed interface is also easy to learn how to use, because, as shown in Figure 3, nine out of the ten participants took the shortest time to learn how to use it.

SUS

The mid column of Table 2 lists the total SUS scores for the interfaces. The proposed interface was ranked top among the three interfaces (69.6). The proposed interface was ranked top in eight out of ten statements of SUS questionnaire. The proposed interface can be interpreted as between "Good" and "OK" according to Bangor's adjective rating [1], which classifies the total SUS score as Best imaginable (90 or more), Excellent (85 or more), Good (70 or more), OK (50 or more), Poor

Table 3. Mean and (standard deviation) of numbers of touch operations, cursor operations and errors made in touch operations to complete one sentence.

	# of touch operations	# of cursor operations	# of errors made in touch operations
Keypad-Flick only	47.9 (11.7)	0.08 (0.57)	4.98 (3.44)
Speech-input-enabled IME	27.8(11.8)	13.1 (6.9)	0.76 (1.22)
Proposed interface	8.9 (7.9)	0.26 (0.69)	0.86 (1.40)

(40 or more), and Worst imaginable (other) derived from a number of case studies. We conducted an ANOVA test since the normality assumption was not rejected with a significant level at 0.05 in Shapiro-Wilk test. The results did not indicate a significant difference between the interfaces ($F_{(2, 8)} = 1.346$, $p > .05$).

NASA-TLX

The last column of Table 2 and Figure 4 show the weighted NASA-TLX scores and those of six subscales, respectively. The ratings on the six subscales are summed into a single measure called a "Weighted Workload" (WWL) score, by using weights computed on the basis of the participants' answers to pairwise comparison on perceived importance of the subscales. The WWL score was 58.5 for "Keypad-Flick only", 49.1 for "speech-input-enabled IME" and 27.9 for the proposed interface. We conducted an ANOVA test since the normality assumption was not rejected with a significant level at 0.05 in Shapiro-Wilk test. The results indicate a significant difference ($F_{(2, 8)} = 8.878$, $P<0.05$). Bonferroni's post-hoc test shows that the difference between "speech-input-enabled IME" and the proposed interface is marginally significant, namely, $p = 0.063$. Particularly, the proposed interface had far lower scores for "physical demand", "frustration" and "effort" than those for the other two interfaces. The quick selectivity of N-best results and easy access to the target word are considered to alleviate the workload in these subscales.

DISCUSSIONS

The proposed interface had significantly lower WWL than "speech-input-enabled IME" and "Keypad-Flick only" had in NASA-TLX although the proposed interface did not achieve a significant reduction in time required to complete a sentence when the morpheme-level correction screen was used for text modification. A probable reason is due to the reduction of the number of key operations. Table 3 lists the means and standard deviations of the numbers of touch operations, cursor operations counted in the touch operations and corrected errors made in touch operations to complete a sentence. The proposed interface needed 8.9 touch operations to complete a sentence in average, whereas "speech-input-enabled IME" needed 27.8. The total counts included those for cursor operations and back spacing. Most of the operations reduced in number by the proposed method were cursor operations. The number of corrected errors made in touch operations was below 1 for both the proposed method and "speech-input-enabled IME" while that for "Keypad-Flick only" counted about 5. Note that uncorrected errors remained, but they are ignorable. There were a few cases that the proposed interface performed worse than the two other methods. One case was when ASR errors spanned across many morphemes. In the user study, the participants were not allowed to re-enter the text by speech input; but in a real situation, users should try the speech input again.

CONCLUSIONS

We proposed an efficient speech-recognition error-correction interface for smartwatches. The proposed interface enables quick text modification with a narrow display area by providing easily comprehensible list of N-best results and easily accessible list of morphemes produced by automatic segmentation with a Japanese morpheme analyzer.

We evaluated the proposed interface by comparing it with a standard Japanese keyboard and a speech-input-enabled IME in a user study. The results indicate that the proposed interface required 35% and 17% less time to complete a sentence compared to "Keypad-Flick only" and "speech-input-enabled IME", respectively. The proposed interface had the highest in SUS score in terms of usability, and the lowest WWL score in terms of perceived workload.

We think the design concept of the proposed method is effective in many languages other than Japanese. In the case of some Asian languages, such as Thai and Chinese, the morpheme-level or word-level access to an ASR result should reduce the number of operations for text

modification. The multiple screens showing the N-best results and an extended screen listing a sequence of words should also be applicable and effective for European languages without the need of implementing a morpheme analyzer.

APPENDIX

Fifty original Japanese sentences for evaluation.

1. 書名の欄の下に署名を手書きで記述する
2. 最近学校へ行きたくないのは転校が理由
3. これはとても良い機械だと思い、いい機械を買った
4. 来週のクリスマスイブの日ですが相手いません
5. 生物のテスト対策で、器官の勉強をする
6. 彼は化学の知識も興味も全く持っていない
7. 魚貝類はエビやカニ、タコなどは含まれません
8. 彼の実家は海草の養殖をしている
9. チームの傘下の連絡を受けて皆驚かされた
10. 彼女は医師に従って、薬を毎日飲んだ
11. 大きな公園で行われた公演は成功だ
12. 三限目は恵道館でプログラミングの講義だ
13. 池に落としてしまったアイフォンはまだ浸かったまま
14. 私のオススメはグーグル製のスマートフォンです
15. 綺麗な風景の撮影を自分のアイフォンで撮る
16. 彼は数分に一回ツイッターに投稿をする
17. 彼はどんな苦しい状況でもにこにこしている
18. 公園に大きな丸太がごろごろところがっている
19. 保育園のアサガオがぐんぐんと成長している
20. 発表を終えた彼の目はきらきら輝いていた
21. カラメルの層をパリリと割るとクリームが飛び出てくる
22. 今夜は飲み会があるので朝からうきうきしている
23. のろのろ仕事をしてたら、終電を逃していた
24. もうおなかがぺこぺこだ。早く何か食べたいです。
25. 室内は時計のかちかちという音だけが聞こえる
26. 強風のせいで、窓枠ががたがた音をたてる
27. 板東先生の言いたいことが全く分からない
28. 来週の今日は伊東先輩の誕生日だ
29. 今日と明日の当番は長居さんが担当する
30. 真冬でも川井君は寒そうな格好をしている
31. 昨日から小嶋さんは風邪をひいて欠席している
32. 後輩の仲村が休むと、みんなしゃべらなくなる
33. となりの山元さん家はいつもワイワイにぎやかだ
34. 齋藤先生の評判はいつも二手に分かれる
35. 川野さんの持つ硬貨はクラスで一番きれいだ
36. 高梁監督の解任はかなりショッキングです
37. 先日広野さんからいろいろな話を聞かされた
38. 部長の喜田さんですがいつも提示に帰宅される
39. 阪田先輩は普段は優しいが、怒ると怖い
40. 夜神君はいずれ新世界の神になるだろう
41. 犯人の疑いがかけられているのは火口さんだ
42. 私の席の隣の仁志田さんはよく昼寝をする
43. 友達の橋下くんは皆からハッシーと呼ばれる
44. 正直言って、彼の言っていることはおかしい
45. テストの成績がよかったら、ゲームを買って帰る
46. 歯を磨くときはごしごし強く磨いてはいけない
47. 質より量の多いウマい飯が今夜待っている
48. カットマンのプレイングはみんなをわくわくさせられる
49. 今日は知真館三号館で授業がある
50. うちの部活の顧問は元プロの星島先生だ

REFERENCES

[1] Aaron Bangor, Philip T. Kortum and James T. Miller, 2008. An Empirical Evaluation of the System Usability Scale. In *International Journal of Human-Computer Interaction*, Vol. 24, No. 6, 574-594. https://doi.org/10.1080/10447310802205776

[2] Google. 2018. Android Developers. (2018). https://developer.android.com/reference/android/speech/SpeechRecognizer

[3] Goo-labo: Japanese-morphological-analysis-API. (2018). https://labs.goo.ne.jp/api/jp/morphological-analysis/

[4] John Brooke. 1996. SUS: a 'quick and dirty' usability scale. In Patrick W. Jordan, B. Thomas, Ian Lyall McClelland, Bernard Weerdmeester (Eds.) *Usability Evaluation in Industry*, 189-194.

[5] 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.

[6] Junhwi Choi, Kyungduk Kim, and Sungjin Lee. 2012. Seamless Error Correction Interface for Voice Word Processor. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '12)*. IEEE, Kyoto, Japan. 4973-4976. https://doi.org/10.1109/ICASSP.2012.6289036

[7] Keith Vertanen and David J. C. MacKay. 2010. Speech dasher: fast writing using speech and gaze. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, New York, 595–598. https://doi.org/10.1145/1753326.1753415

[8] Keith Vertanen and Per Ola Kristensson. 2009. Parakeet: A Continuous Speech Recognition System for Mobile Touch-Screen Devices. In *Proceedings of Intelligent User Interface (IUI '09)*. ACM, Sanibei Island, Florida, USA, 237-246. https://doi.org/10.1145/1502650.1502685

[9] Keith Vertanen and Per Ola Kristensson. 2010. Intelligently aiding human-guided correction of speech recognition. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI '10)*. Association for the Advancement of Artificial Intelligence, 1698-1701.

[10] Lijuan Wang, Tao Hu, Peng Liu, and Frank K. Soong. 2008. Efficient Handwriting Correction of Speech Recognition Errors with Template Constrained Posterior (TCP). In *Proceedings of the 15th Annual Conference of the International Speech Communication Association (INTERSPEECH '08)*. International Speech Communication Association (ISCA). 2659-2662.

[11] Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. *Advances in Psychology*, VOL.52, 139-183. https://doi.org/10.1016/S0166-4115(08)62386-9

[12] 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 (INTERSPEECH '14)*. International Speech Communication Association (ISCA). 1194-1198.

[13] Yuan Liang, Koji Iwano, and Koichi Shinoda. 2015. Error Correction Using Long Context Match for Smartphone Speech Recognition. *IEICE Transactions on Information and Systems*, VOL.E98-D, NO.11, 1932-1942. https://doi.org/10.1587/transinf.2015EDP7179

[14] Wikipedia. 2019. Japanese Input Method. 2019. https://en.wikipedia.org/wiki/Japanese_input_methods