Effectiveness of Ambiguity-aware Web Search Algorithm for Mobile Phone

Masafumi Matsuhara

***Abstract*—We have proposed a fast Web search method for mobile terminals such as mobile phones. We need to input some terms for Web searching usually. A full QWERTY keyboard is used for input of terms. It is not easy to press the intended key because the key size is small on a mobile phone. Moreover, a user needs to press a few keys per *Kana* character since one *Kana* character generally consists of a few alphabets in Japanese. A flick input method has been developed and is used for input of *Kana*-characters on a touch panel of a smart phone. Because the method needs only about 12 keys, the key size is larger than a full QWERTY keyboard. However, the flick operation is not easy. In our proposed method, a user is able to easily input terms because our proposed method needs only one keystroke per character using only 12 keys without flick operations. Character-strings inputted by the user are ambiguous because each string corresponds to not only the intended term but also other terms. However, the ambiguous strings are not translated into the intended terms in our proposed method. The system based on our proposed method directly searches Web pages by the ambiguous strings and outputs the searched result based on co-occurrence. Thus, we are able to find the intended Web page on a mobile phone rapidly and easily. In the result of the evaluation experiment, the Web search accuracy was 80[%] and the system was able to search Web pages by the correct terms at 90[%]. It was proved that our proposed method was effective for mobile terminals.**

***Index Terms*—Web Search, Mobile Phone, Co-occurrence, Number-string, Disambiguation.**

I. Introduction

Recently, performance of mobile terminals such as mobile phones is greatly increasing. Most mobile phones are smart phones, e.g. iPhone, Nexus and so on. A smart phone has an advanced mobile operating system and is able to connect the Net. We are able to search Web pages since it has a Web browser. In Web search procedure, we need to input search-terms and choose the intended page generally. The processes have to be performed on a small device such as a mobile phone.

We usually input some terms for the Web searching. A full QWERTY keyboard is used for input of terms. It is not easy to press the intended key because the key size is small on a small device such as a mobile phone. Moreover, a user needs to press a few keys per *Kana* character since one *Kana* character generally consists of a few alphabets in Japanese.

Ordinary Japanese sentences are expressed by two kinds of characters: i.e. *Kana* and *Kanji*. *Kana* is Japanese phonogramic characters and has about fifty kinds. *Kanji* is ideographic Chinese characters and has about several thousand kinds. Therefore, we need to use not only a *Kana* input method but also a *Kanji* input method in order to input Japanese sentences into computers including mobile phones. A typical *Kanji* input method is based on the *Kana-Kanji* conversion algorithm of non-segmented Japanese sentences. This method translates non-segmented *Kana*-sentences into *Kanji-Kana* mixed sentences. A *Kana*-string inputted by a user is ambiguous because it may correspond to several *Kanji*-words in Japanese. For example, the *Kana*-string "きかい (*kikai*)" corresponds to "機械 (a machine)", "機会 (an opportunity)", "奇怪 (strange)" and so on. A user needs to choose the intended one in the word-candidates. They are weighted and are shown to the user in descending weight order. It is troublesome for the user to choose the intended word when its rank is low. Therefore, we need an efficient *Kana* input and *Kana-Kanji* conversion algorithm to realize rapid Japanese input on mobile phones.

We focus on 12 keys layout on mobile phones. We are able to input characters easily since each key size on 12 keys layout is larger than that of a full QWERTY keyboard. However, the number of keys is less than the kinds of characters. English has 26 alphabet characters and Japanese has about 50 *Kana* characters and so on. Therefore, it is difficult to input the terms and search Web pages rapidly and easily on mobile terminals such as a mobile phone.

A flick input method has been developed and is used for input of *Kana*-characters with 12 keys layout on a touch panel of a smart phone. In the input method, the chosen key represents a consonant and a user decides the vowel by a direction of a key-flick, e.g. the left key-flick means the vowel "*i*", the upward key-flick means "*u*" and so on. However, the flick operation is not easy.

To input characters into a mobile phone, the letter cycling input method is also used. The inputted character is decided by the chosen key and the number of pressing it. For example, a user chooses the key "2" and presses three times in order to input the alphabet "c". It is troublesome for users since the input method needs several keystrokes per character. Therefore, a method is demanded which enable to promptly and easily input terms for Web searching.

Some input methods for mobile phones have been proposed [1],[2]. The methods enable us to input one alphabet per key press on the keyboard of 9 keys. Since three or four letters are assigned to each key of 9 keys, the specific letter intended by one key press is ambiguous. This system disambiguates the pressed keys on word-level. However, the system is for English mainly. Some input methods have been proposed for Japanese [3]-[8]. The methods enable us to

Masafumi Matsuhara, Department of Software and Information Science, Iwate Prefectural University, JAPAN.

input one *Kana* character per key press. Since about five *Kana* characters are assigned to each key on a mobile phone, the specific character intended by one key press is ambiguous. The methods disambiguate by dictionaries. Therefore, they are not able to translate the number-strings into words not included into the dictionary. Some predictive input methods have been proposed [9]-[12]. The methods output word-candidates by prediction or completion. The number of key presses increases to select the intended word when there are many word-candidates. Therefore, we focus on a text input method without prediction.

These input methods enable us to search Web pages by translation on a mobile phone. We are able to search Web pages to translate the inputted number-strings into the intended terms by the input methods for a mobile phone: i.e. number-*Kanji* translation in Japanese. However, it is troublesome for users to choose the intended terms in some candidates for each number-string.

We have proposed a fast Web search method for mobile phones by input of ambiguous character-strings without translation. In our proposed method, a user is able to input one character per keystroke using only 12 keys. We consider that the ambiguity of number-strings is able to be resolved by co-occurrence among search-terms in Web pages without translation. The system based on our proposed method is able to disambiguate the number-strings by the co-occurrence and enables us to find the intended Web pages on a mobile phone rapidly and easily.

This paper shows the processes of our proposed method and the result of evaluation experiment for our proposed method.

II. Web Search with Translation

We are able to search Web pages by translation on a mobile phone without our proposed method. However, the Web searching with translation is troublesome for a user. Fig. 1 shows the procedure of the Web searching with translation.

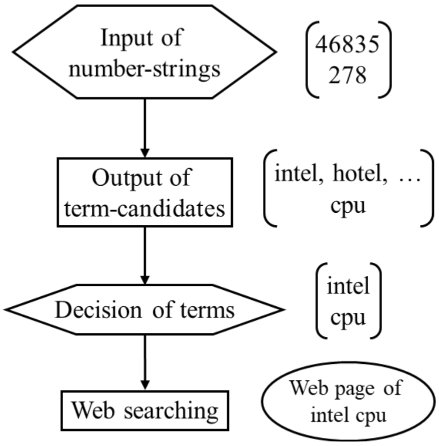

Fig. 1 Procedure of Web search with translation

A user inputs the number-strings for the intended terms. In Fig. 1, the search-terms intended by the user are “intel” and “cpu”. TABLE I shows how to input alphabet characters on a mobile phone.

TABLE I
Correpondence of Number to Alphabet

1: . ? - '	2: A B C	3: D E F
4: G H I	5: J K L	6: M N O
7: P Q R S	8: T U V	9: W X Y Z
*:	0:	#: (space)

The assignment of alphabet characters is commonly used and the system needs only one keystroke per alphabet character. The inputted number-strings are “46835” and “278” in this case. The system translates each number-string into term-candidates. In Fig. 1, the term-candidates for “46835” are “intel”, “hotel” and so on. The user needs to choose the intended one in some candidates. The user chooses “intel” in Fig. 1. Then, the system searches the Web page by the query of the terms chosen by the user. In Fig. 1, the query used for Web search by the system is “intel AND cpu”. Although the user is able to input the number-strings for the search terms rapidly, it is troublesome for the user to choose the intended term in many candidates for each number-string.

We consider that ambiguity of number-strings is able to be resolved by co-occurrence among the search-terms in Web pages without translation. A term for a string co-occurs with a term for another one of the number-strings in Web pages. Therefore, the system based on our proposed method is able to disambiguate the number-strings by the co-occurrence and enables us to find the intended Web page on a mobile phone rapidly and easily.

III. Outline of Our Proposed Method

Fig. 2 shows the procedure of our proposed method.

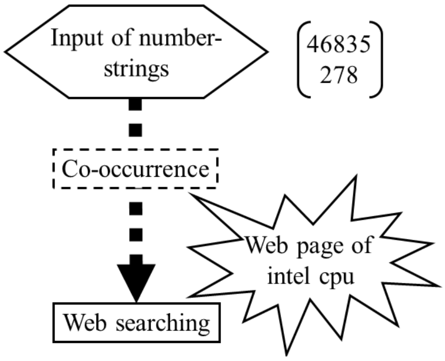

Fig. 2. Procedure of Web search without translation

A user inputs the number-strings for the intended terms. TABLE I shows how to input alphabet characters on a mobile phone. The assignment of alphabet characters is commonly used. Moreover, the system based on our proposed method needs only one keystroke per alphabet character. Therefore, the user is able to input characters rapidly and easily.

In Fig. 2, the user inputs the number-strings “46835” and “278”. The inputted number-string has ambiguity because the number-string “46835” corresponds to not only “intel” but also “hotel” and so on. However, the system based on our proposed method does not need to translate the number-strings into the terms intended by the user. The system directly searches Web pages by OR-query of the terms for each number-string.

In Fig. 2, the system based on our proposed method performs Web search by the query “(intel OR hotel) AND cpu”. The search term “intel” co-occurs with the term “cpu” in Web pages because the searched result has many pages. On the other hand, the search term “hotel” does not co-occur with the term “cpu” in Web pages because the searched result has a few or no pages.

Our proposed method is able to disambiguate the number-strings on Web searching. The searched result is shown as a list of Web pages related to the search terms, with the most relevant page appearing first, then the next, and so on. Thus, the user is able to find the intended page on mobile terminals such as mobile phones rapidly and easily.

Then, the system extracts the terms from the Web page selected by the user. The system uses the extracted terms for generating the search-query in the next search. Because the terms are learned, the system is able to adapt to the user gradually.

We are able to find the intended page by the same method in Japanese also. TABLE II shows the correspondence of the number-keys with *Kana*-characters: e.g. the key “4” represents “た(*ta*)” or “ち(*ti*)” or “つ(*tu*)” or “て(*te*)” or “と(*to*)” of *Kana*-characters. Then, a number-character of 12 keys generally corresponds to a consonant. Vowel information is degenerated in our proposed method.

TABLE II
Correspondence of Number to Kana

1:あいうえお	2:かきくけこ	3:さしすせそ
4:たちつてと	5:なにぬねの	6:はひふへほ
7:まみむめも	8:やゆよ	9:らりるれろ
*:゛゜	0:わをん	#: (space)

The user presses the key “*” for a voiced consonant and a p-sound in our proposed method. For example, the user inputs the number-string “4*12” for the Japanese word “大工 (a carpenter)” of which the pronunciation is “だいく(*ta*iku*)”.

The assignment of *Kana* characters is commonly used in Japanese. Moreover, the system based on our proposed method needs only one keystroke per Japanese *Kana* character. Therefore, a user is able to input characters rapidly and easily in Japanese.

When the terms intended by the user are “北海道 (*hotukaito*u*; a proper noun)” and “大学(*ta*ika*ku*; the university)”, the user inputs the number-strings “64214*1” and “4*12*2”. The inputted number-string has ambiguity because the number-string “64214*1” corresponds to not only “大学(*ta*ika*ku*)” but also “同額(*to*uka*ku*; the same amount)”, “土井垣(*to*ika*ki*; a person's name)” and so on. Our proposed method disambiguates the number-strings on Web searching. The search-term “北海道(*hotukaito*u*)” co-occurs with the term “大学(*ta*ika*ku*)” in Web pages because the searched result has many pages. On the other hand, the search-term “北海道(*hotukaito*u*)” does not co-occur with the term “同額(*to*uka*ku*)” in Web gages because the searched result has a few or no pages.

Our proposed method uses OR-queries in order to realize the Web search. In this case, the system based on our proposed method performs Web search by the query “北海道(*hotukaito*u*) AND {大学(*ta*ika*ku*) OR 同額(*to*uka*ku*) OR 土井垣(*to*ika*ki*) }”. The searched result is shown as a list of Web pages related to the search terms, with the most relevant page appearing first, then the next, and so on. Thus, the user is able to rapidly and easily find the intended page on mobile terminals such as mobile phones in even though Japanese and so on.

IV. Processes

Fig. 3 shows the processes for our proposed method. The procedure consists of the number-strings input, the search-query generation, the Web searching, the Web page choice and the term extraction in this order.

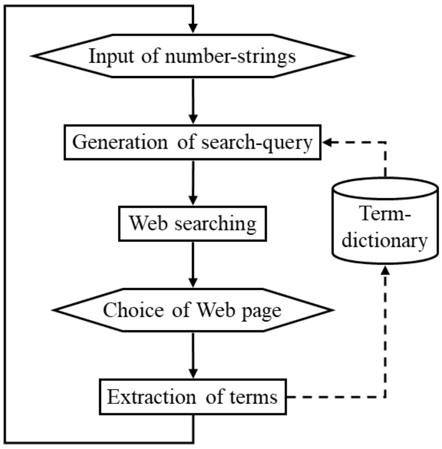

Fig. 3 Processes of our proposed method

A. *Number-Strings Input Process*

In this process, a user inputs the number-strings for the intended terms. The key assignment is shown in TABLE I for English and in TABLE II for Japanese. The assignment of characters is commonly used. Moreover, the system based on our proposed method needs only one keystroke per character. Therefore, the user is able to input terms as number-strings rapidly and easily.

B. *Search-query Generation Process*

The number-strings inputted by the user have ambiguity because each number string corresponds to some term-candidates. However, the system based on our proposed method does not translate the number-strings into the terms intended by the user. The system directly searches Web pages by OR-query of the terms for each number-string. The system looks up each number-string in the term-dictionary. The example of the dictionary is shown in TABLE III.

TABLE III
Example of Term-Dictionary

Number-string	Term
:	:
278	cpu
:	:
46835	hotel
46835	intel

:	:
4*12*2	大学
4*12*2	同額
:	:

When the number-string agrees with the one in the term-dictionary, the term which corresponds to the number-string is added to the search-query. Then, the search-query is generated by the term-dictionary.

C. Web Searching Process

The system searches Web pages by the query generated in the previous process. The searched result is shown as a list of Web pages related to the search terms, with the most relevant page appearing first, then the next, and so on.

D. Web Page Choice Process

The user needs to choose the intended Web page in the result because the searched result has many pages ordinarily. The user is able to immediately find the intended Web page when the rank of the Web page is high in the searched result.

E. Term Extraction Process

In this process, the system extracts the words in the chosen page as the terms. Then, the extracted terms are registered into the term-dictionary with the number-strings. The system is able to find the number-strings for the terms by the key assignment for the input in English.

However, it is difficult to find the number-strings for the terms in Japanese because Japanese has a lot of kinds of characters and Japanese sentences are not segmented ordinarily. The system uses “ChaSen” for the segmentation. “ChaSen” is a Japanese morphological analysis system [13]. The system is able to extract the terms and the number-strings because “ChaSen” segments the sentences and outputs the words and their *Kana* characters in Japanese. The system is able to find the number-strings for the *Kana* characters by the key assignment for the input in Japanese. The extracted number-strings and terms are registered into the term-dictionary.

The terms in the dictionary are used for the next searching. The selected Web page is suitable for the user. We consider that the words in the page are also suitable for the user. Therefore, it is effective to use the extracted words for the next searching. Because the words are learned, the system is able to adapt to the user gradually.

V. Evaluation Experiment

The system based on our proposed method has been developed for the experiment. The system searches Web pages which are in Japanese mainly.

A. preliminary Experiment

We evaluated the queries in the Web search history before Web searching. The system based on our proposed method generates the search-query by the term-dictionary. When the user chooses the intended Web page in the searched result, the number-strings and the terms are extracted from the Web page and are registered into the term-dictionary. The terms in the dictionary are used for the next searching. Therefore, the term whose frequency is 2 and over is able to be used for Web search in our proposed method. However, the inputted number-string is able to correspond to not only the intended term but also other terms: i.e. the number-string has ambiguity. Then, we evaluated how much the number-string is ambiguous in the Web search history.

The evaluation data for the preliminary experiment is the actual Web search history in the year 2014 of the author of this paper. The number of queries is 2,690. The queries have 4,600 terms. The kinds of terms are 2,300. The result of the evaluation is shown in TABLE IV.

TABLE IV
Evaluation Result

Kinds of terms	2,300
Kinds of number-strings	2,156
Whose frequency >=2	713 (33.1%)
Ambiguous number-strings	123 (17.3%)

The kinds of the number-strings extracted from the terms are 2,156 in TABLE IV. The number-strings whose frequency is 2 and over are 713. The rate is 33.1[%] for the kinds of the number-strings. The number-strings include the ambiguous ones of 123. The rate is 17.3[%] for the number-strings whose frequency is 2 and over.

In the ambiguous number-strings, the ambiguities of 59 strings are caused by difference between a small letter and a capital letter: e.g. “International” and “international” and so on. These are able to be considered to the same terms in Web searching. Then, only 64 number-strings are ambiguous and the rate is less than 10[%] for the number-strings whose frequency is 2 and over. Therefore, the system based on our proposed method is able to perform Web search using the intended terms whose accuracy is more than 90[%].

B. Data and Procedure

The data for extracting terms has 20 pages. They are included in the bookmark of the author of this paper. The system extracts the terms from the 20 pages. The result of the extraction is shown in TABLE V.

TABLE V
Learned Dictionary

Kinds of terms	3,561
Kinds of number-strings	2,297
Average of perplexity per number-string	1.55
Entropy [bit]	10.74

In TABLE V, the kinds of terms are 3,561 and the kinds of number-strings are 2,297 in the term-dictionary. The average of perplexity is 1.55 per number-string. Entropy H of the dictionary is calculated as follows:

$$H(x) = -\sum_{i=1}^{N} P(x_i)\log_2 P(x_i)$$

$$P(x_i) = \frac{C(x_i)}{T}$$

where N expresses the kinds of number-strings in the term-dictionary. N is 2,297 in the experiment. T is the number of terms in the dictionary. T is 3,561 in the experiment. x_i means a number-string in the dictionary. $C(x_i)$ is the appearance frequency of the number-string x_i in the dictionary. $P(x_i)$ is the probability of the number-string x_i.

The system searches 20 Web pages by the learned dictionary. The 20 pages are included in the author's bookmark and are different from the 20 pages for extracting terms. We evaluate the search results by each title of the 20 pages. Each title is segmented by “ChaSen”. The system searches Web pages by the query based on the number-strings which corresponds to each segmented title. The search-query is generated by the terms which are looked up by the number-strings in the term-dictionary.

The average of the terms included into a query is 7.67 because the average of words in a segmented title is 4.95 and the average of perplexity per number-string is 1.55 in TABLE V. The system needs to disambiguate the perplexity for correctly Web searching.

C. *Results and Considerations*

TABLE VI shows the experiment result. The searched result is shown as a list of Web pages related to the search terms, with the most relevant page appearing first, then the next, and so on.

TABLE VI
Search Result

	Rate[%]
First place	65.0 (13/20)
First page	80.0 (16/20)
Others	20.0 (4/20)
Search by the correct terms	90.0 (18/20)

When there is the intended Web page at the top of the list, it is “First place” in TABLE VI. When the intended page is in the first 10th places of the list, it is “First page” in TABLE VI. The list at the first page of the searched result has the links to 10 pages. We were able to find the intended Web pages at 80[%] at the first page in the searched result. It was proved that the system based on our proposed method was able to disambiguate number-strings by co-occurrence among search-terms in Web pages without translation and enabled us to find Web pages rapidly.

In the experiment, the intended pages at 20[%] were not found. The titles of 2 pages in the not found pages had proper nouns which were not included in the dictionary of “ChaSen”. The proper nouns were not registered into the term-dictionary and the search-query did not have them as the terms. Therefore, the system was not able to find the Web pages. It is necessary to improve a Japanese morphological analysis system.

The ranks of 2 pages in the not found pages were low in the searched result. However, the queries of the 2 pages were correct. The system was able to disambiguate the inputted number-strings because the queries consisted of the terms intended by the user. Thus, the system was able to search the Web pages by the correct terms at 90[%]. The search system needs to more adapt to the user for searching the intended page.

VI. Conclusion

In this paper, we have proposed a fast Web search method on mobile terminals such as mobile phones and evaluated Web search performance in the system based on our proposed method.

Mobile phones such as smart phones enable us to search Web pages. We usually input some terms for Web searching. A full QWERTY keyboard is used for input of terms. It is not easy to press the intended key because the key size is small on a mobile phone. Then, we focus on 12 keys layout on mobile phones. We are able to input characters easily since each key size on 12 keys layout is larger than that of a full QWERTY keyboard. However, it is less than the kinds of alphabet characters. Therefore, it is difficult to input the search-terms and search Web pages on mobile terminals such as mobile phones. Our proposed method needs only one keystroke per character on 12 keys layout without flick operations or without several keystrokes. Then, the user is able to rapidly input terms. The key assignment is much the same as the commonly used one. Therefore, the user is able to easily input terms.

The terms inputted by the user are expressed as number-strings. The system based on our proposed method searches Web pages by the number-strings for the search terms. Each number-string inputted by the user corresponds to many terms and has ambiguity. However, the system is able to disambiguate the inputted number-strings by co-occurrence among the terms in Web pages. Therefore, we are able to find the intended Web page on a mobile phone rapidly and easily.

In the result of the evaluation experiment, the Web search accuracy was 80[%] and the system was able to search Web pages by the correct terms at 90[%] even though the inputted number-strings for Web searching were ambiguous. It was proved that our proposed method was effective for Web searching on mobile terminals such as mobile phones.

One of future work is to use more data for experiment. The data was small in the experiment in this paper. We will show effectiveness of our proposed method by the experiment using large data. Then, we will apply our proposed method to other languages. Because our proposed method is fundamentally independent of language property, our proposed method is able to apply to other languages. Then, we need to evaluate the system based on our proposed method in other languages.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number 15K00155.

References

[1] Cliff Kushler, “AAC USING A REDUCED KEYBOARD,” Proceedings of CSUN98, Los Angeles, USA, March 1998.

[2] Sasa Hasan and Karin Harbusch, “N-Best Hidden Markov Model Supertagging to Improve Typing on an Ambiguous Keyboard,” Proceedings of Seventh International Workshop on Tree Adjoining Grammar and Related Formalisms, pages 24--31, Vancouver, British Columbia, Canada, May 2004.

[3] Masafumi Matsuhara, Kenji Araki, Yoshio Momouchi and Koji Tochinai, “Evaluation of Number-Kanji Translation Method of Non-Segmented Japanese Sentences Using Inductive Learning with

Degenerated Input," Lecture Note in Artificial Intelligence 1747, pages 474-475. Springer-Verlag, December 1999.

[4] Masafumi Matsuhara, Kenji Araki and Koji Tochinai, "Evaluation of Number-Kanji Translation Method Using Inductive Learning on E-Mail," Proceedings of Third IASTED International Conference on Artificial Intelligence and Soft Computing (ASC'2000), pages 487-493, Banff, Alberta, Canada, July 2000.

[5] Masafumi Matsuhara and Satoshi Suzuki, "An Efficient Context-Aware Character Input Algorithm for Mobile Phone Based on Artificial Neural Network," Proceedings of The 3rd International Conference on Awareness Science and Technology (iCAST 2011), pp.314-318, Dalian, China, September, 2011.

[6] Masafumi Matsuhara and Satoshi Suzuki, "Effectiveness of Context-Aware Character Input Method for Mobile Phone Based on Artificial Neural Network," Applied Computational Intelligence and Soft Computing, Volume 2012 (2012), Article ID 896948, 6 pages.

[7] Masafumi Matsuhara, Miki Itoh, Goutam Chakraborty, Hiroshi Mabuchi, "An Efficient Pressure-Aware Character Input Algorithm for Mobile Phones," Proc. of 2013 International Joint Conference on Awareness Science and Technology & Ubi-Media Computing (iCAST 2013 & UMEDIA 2013)}, pp.191-196, University of Aizu, Aizu-Wakamatsu city, Japan, November 2-4, 2013.

[8] Masafumi Matsuhara, Taichi Sugawara, Goutam Chakraborty and Hiroshi Mabuchi, "An Efficient Image-Aware Kana-Kanji Conversion Algorithm for Twitter on Mobile Phones," The IEEE 7th International Conference on Awareness Science and Technology (iCAST 2015)}, Session 3-2, Qinhuangdao, China, Sep.22-24,2015.

[9] Kumiko Tanaka-Ishii, Yusuke Inutsuka and Masato Takeichi, "Personalization of text entry systems for mobile phones," Proceedings of Sixth Natural Processing Pacific Rim Symposium, pages 177--184, Japan, November 2001.

[10] Kumiko Tanaka-ishii, "Word-based predictive text entry using adaptive language models," Natural Language Engineering, Volume 13 Issue 1, p.51-74, March 2007.

[11] Antal van den Bosch and Toine Bogers, "Efficient context-sensitive word completion for mobile devices," Proceedings of the 10th international conference on Human computer interaction with mobile devices and services, pp.465-470, Amsterdam, the Netherlands, September 2008.

[12] Mark D. Dunlop and Michelle Montgomery Masters, "Investigating five key predictive text entry with combined distance and keystroke modelling," Personal and Ubiquitous Computing, Volume 12 Issue 8, Springer, November 2008.

[13] Yuji Matsumoto, Akira Kitauchi, Tatsuo Yamashita, Yoshitaka Hirano, Hiroshi Matsuda, Kazuma Takaoka and Masayuki Asahara, "Japanese Morphological Analysis System ChaSen version 2.2.1," Nara Institute of Science and Technology, December 2000.