Character based Hangeul search using Location-specific Character Frequency

Jung-Hwa Lee, Jong-Min Lee, Seong-Woo Kim, *Member, KIMICS*

Abstract—Hangul search functionality, including dictionary search is used in many Hangeul applications. Existing research of hangeul search method is the study of using hangeul syllable as a basic unit. However when you consider the characteristics of Hangul, the research of using hangeul character as a basic unit is needed. In this paper we propose the character based hangeul search method using the location-specific frequency information and verify the effectiveness of the proposed method through the experiments.

Index Terms — character based search, search, Hangeul code, Hangeul

I. INTRODUCTION

Search functionality, including dictionary search is used in many applications. Especially, The applications such as database that need to handle large amounts of data requires the quick search algorithms[1][2].

Existing research on how to search hangeul word quickly is being studied as part of Hangeul spelling checker and hangeul information retrieval system, etc, and most of research is the study of hangeul index method using B+ tree or trie[3]. The important thing is that existing hangeul search method is the method of using hangeul syllable like '각', '국' as a basic unit. But when you consider the characteristics of Hangul, the research of using hangeul character like 'ㄱ', 'ㅏ' as a basic unit is needed, because hangeul has the syllable of combined character. Recently, we can find the search function that use hangeul character as a basic unit in some navigation systems[4][5]. A study of character based hangeul search method is needed for such applications.

We have to consider of incomplete syllable as well as complete syllable in case of using hangeul character as a basic unit. Because there are many type of incomplete syllable in hangeul, we must define the type of incomplete hangeul syllable[6]. So we define the type of incomplete hangeul syllable first, and study the method of processing incomplete hangeul syllables in this paper.

In general, when we search words in the database system, we can search the word quickly using the index such as b+ tree or hashing. But in case of using hangeul character as a basic unit, we can not use this mechanism because that use a hangeul syllable as a basic unit, therefore we must use the simple word match algorithm only that has many problems in terms of performance.

II. HANGEUL CODE USED IN CHARACTER BASED HANGEUL SEARCH

Hangeul code is required for representations of the hangeul characters on the computer. There are two types in hangeul code: combined hangeul code and complete hangeul code. Hangeul character is a basic unit of code system in combined hangeul code, so we can representation of hangeul syllable through a combination of the combined hangeul code points. On the other hand hangeul syllable is a basic unit of code system in complete hangeul code.

Until now, KS C 5601 complete code is used mostly, Each of hangeul syllable like '가', '학' has two bytes code in this code system. This hangeul code name was changed to KS X 1001 in 1997. Hangeul combined code that was called KSSM is also national standard now. Finally, Two types of hangeul code are existed in KS X 1001[7]. Fig. 1 shows the representation of hangeul syllable using KS X 1001 hangeul code.

Manuscript received August 1, 2009; revised August 25, 2009.

Jung-hwa Lee is with the Department of Computer software engineering, Dongeui University, Busan, 614-714, Korea (Tel: +82-51-890-1729, Fax: +82-51-890-1724, Email: junghwa@deu.ac.kr)

가 - 0xB0A1(complete) 0x8861(combined)
학 - 0xC7D0(complete) 0xD062(combined)

Fig. 1 Example of the representation of hangeul syllable using KS X 1001 hangeul code

ISO(International Organization for Standardization) and Unicode consortium made a new coded character set, that is ISO 10646-1/Unicode. In this code system, each of characters of all countries occupies 2 bytes, hangeul is the same.

In ISO 10646-1/Unicode, two types of hangeul code are included: combined hangeul code and complete hangeul code. But this code system is completely different from the KS X 1001 hangeul code system.

Fig. 2 shows the representation of hangeul syllable using ISO 10646-1/Unicode.

가 - 0xAC00(complete)
0x1100 0x1161(combined)
학 - 0xD559 (complete)
0x11120 x1161 0x11A8(combined)

Fig. 2 Example of the representation of hangeul syllable using ISO 10646-1/Unicode

ISO 10646-1/Unicode is already in use as a korean standard, that is KS X 1005-1. Especially KS X 1005-1 compete hangeul code has all code point of 11,172 modern hangeul syllables, so we can separate character code points from a syllable code point[8][9].

The follows equation shows the method of separate character code points from a syllable code point in KS X 1005-1[10].

$$Ci = Code\ point - 0xAC00;$$
$$SFi = Ci\ \%\ Nf$$
$$SPi = ((Ci - SFi) / Nf)\ \%\ Np$$
$$SIi = ((Ci - SFi) / Nf) / Np$$

Here, *Nf* is the number of syllable-final character and *Np* is the number of syllable-final character. *SFi, SPi, SIi* is the index of syllable final, peak and initial character

In case of our research, the code point of syllable must be separated character code points as a unit, so we can't use KS X 1001 complete hangeul code although that is used mostly in hangeul application system . Therefore we use KS X 1005-1 complete hangeul code as a basic coded set in this research.

III. DESIGN OF THE CHARACTER BASED HANGEUL SEARCH USING THE LOCATION-SPECIFIC CHARACTER FREQUENCY

A. Define search patterns of the character based Hangeul search

In this paper, Character based hangeul search is defined that hangeul word look up a target word in the set of words such as hangeul dictionary using hangeul character like 'ㄱ', 'ㅏ' as a basic unit.

Hangeul syllable is composed of syllable-initial character and syllable-peak character, and a syllable-final character optionally. That is complete hangeul syllable. Hangeul syllable can be composed of only one character or syllable-initial character and syllable final character we call it incomplete hangeul syllable.

In case of all syllables are complete syllable in target word, just two patterns is considered. The one is the case of including wild cards, the other case is not.

There are many patterns of target word because target word can be composed of incomplete syllables in character based hangeul search, so we have to define search patterns of target word can be used in this method[6].

Fig. 3 shows the search patterns can be used in character based hangeul search

- non-exist syllable-initial character
  /*-ㄹ수록/ -- 할수록, 갈수록, 잘수록, etc
- non-exist syllable-peak character
  /ㄱ-*-ㄴ/ -- 간, 갠, 건, etc
- non-exist syllable-final character
  /가/ -- '가' is matched only
  /가-*/-- 간, 갈, etc
- only exist syllable-initial character
  /ㄱ-*-*/ -- 가, 간, 갈, etc
- only exist syllable-peak character
  /*-ㅏ-*/ -- 가, 나, 간, 날, etc
- only exist syllable-final character
  /*-*-ㄴ다면/ -- 한다면, 간다면, etc

Fig. 3 The search patterns can be used in character based hangeul search

B. Simple character based Hangeul search

When we want to search the target word to source words in based hangeul search, generally, target character is compared to source character in order. For example, we want to find '사랑' from word dictionary, syllable-initial 'ㅅ' is compared with source

words first, and syllable-peak 'ㅏ' is compared with the words that are succeeded first match. And then, Second syllable, 'ㅇ' is also compared with source words in the same method.

Fig. 4 shows example of search order of simple character based hangeul search

ㅅ(syllable initial) -> ㅏ(syllable peak) -> FILL -> ㄹ(syllable initial) -> ㅏ(syllable peak) -> ㅇ(syllable final)

Fig.4 Example of search order of simple character based hangeul search

As you see above, ff the length of target word is n, n times character match have to be succeeded to find a target word. Therefore, Total number of comparison defined as:

$$T_c = C_0 + \sum_{i=1}^{n} C_i$$

T_c is the total number of comparison, C_0 is the number of comparison for first character, that is the same of source dictionary size, and C_i is the number of comparison in each step. Actually, The number of match is follows when we want to search the word '사랑' is the dictionary that is consists of 30,000 words(Fig. 5).

ㅅ(30,000) -> ㅏ(6,895) -> FILL (1,591) -> ㄹ(623) -> ㅏ(52) -> ㅇ(12) = 39,173

Fig. 5 The number of comparison for searching a example word '사랑'.

As shown in Fig. 5, the number of comparison is decreased gradually in accordance with execution of searching. Here, if we can reduce C_i, the total performance of search can be improved. Especially, in case of hangeul, the frequencies of character are many differences with the location of it[11]. Accordingly, if the character that has the lowest frequency is compared first, the number of words for next step will be reduced.

In this paper, we investigate the location-specific frequency of hangeul character and we will use it to determine the search order.

C. The table of location-specific frequency of Hangeul character

We examined the location-specific frequency of hangeul character in the sample dictionary contains 30,000 words.

Following table is the example of location-specific frequency in first hangeul syllables as shown in Table 1.

Table 1. the example of location-specific frequency in first hangeul syllables

In-dex	initial	Freq	peak	Freq	final	Freq
0	ㄱ	874	ㅏ	130	FILL	26990
1	ㄲ	928	ㅐ	419	ㄱ	4646
2	ㄴ	264	ㅑ	612	ㄲ	64
3	ㄷ	376	ㅒ	19	ㄱㅅ	21
4	ㄸ	572	ㅓ	717	ㄴ	8871
5	ㄹ	286	ㅔ	147	ㄴㅈ	16
6	ㅁ	430	ㅕ	338	ㄴㅎ	15
7	ㅂ	549	ㅖ	495	ㄷ	160
8	ㅃ	339	ㅗ	885	ㄹ	6330
9	ㅅ	689	ㅘ	139	ㄹㄱ	139
10	ㅆ	361	ㅙ	129	ㄹㅁ	28
11	ㅇ	947	ㅚ	964	ㄹㅂ	34
12	ㅈ	711	ㅛ	776	ㄹㅅ	2
13	ㅉ	290	ㅜ	769	ㄹㅌ	5
14	ㅊ	324	ㅝ	423	ㄹㅍ	2
15	ㅋ	775	ㅞ	66	ㄹㅎ	24
16	ㅌ	160	ㅟ	853	ㅁ	3193
17	ㅍ	206	ㅠ	817	ㅂ	1484
18	ㅎ	406	ㅡ	326	ㅂㅅ	17
19			ㅢ	337	ㅅ	1203
20			ㅣ	699	ㅆ	8
21					ㅇ	8846
22					ㅈ	233
23					ㅊ	147
24					ㅋ	1
25					ㅌ	315
26					ㅍ	165
27					ㅎ	18

According to above table, In case of searching '닭', if we compare the syllable-initial 'ㄷ' first with example dictionary, 3767 words is remained after first match. But if we do syllable-final 'ㄹㄱ' first, 139 words is remained only. So, the character has the lowest frequency must be compared first to improve the performance of searching.

Generally, whether the syllable has a syllable-final character or not and the syllable-final character is a double character or not is very important in hangeul.

Because the frequency of double syllable-final character is very low in hangeul syllables. So, if target syllable has a double syllable-final character, we had better search the syllable-final character first than the syllable-initial character.

Also, the frequency of character is different according as the location in the word. The more a character is located in the rear of the word is the more the frequency decrease gradually. There are a special case like 'ㄹ'. Syllable initial character 'ㄹ' is located in second syllable has the more frequency than located in first syllable. Therefore, it is very important to consider a location of character as well as a kind of character.

Fig. 6 shows the variation of frequency of character depending on the location.

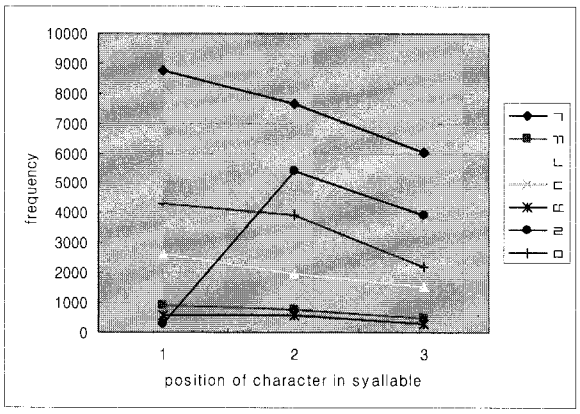


Fig. 6 Variation of frequency of character depending on the location

D. Character based Hangeul search method using location-specific frequency information

Commonly, the word search is performed according to order of location in the simple search method. But that is performed according to ascending order of the frequency of character in proposed method. For example, Fig. 7 shows the search order and the total number of comparison in case of searching '검색' in the sample dictionary using the simple search algorithm.

ㄱ(30,000) -> ㅓ(8748) -> ㅁ(786) -> ㅅ(136) -> ㅐ(26) -> ㄱ(4) = 39,700

Fig. 7 Total number of comparison using the simple search algorithm

The word search is performed according to ascending order of the frequency information in proposed method as shown in Fig. 8.

ㅁ(30,000) -> ㅐ(3193) -> ㄱ(234) -> ㅓ(38) -> ㅅ(12) -> ㄱ(4) = 33,481

Fig. 8 Total number of comparison using the proposed algorithm

When the search word has some incomplete syllables, our method is more efficient, as shown in Fig. 9 and Fig. 10.

ㄱ(30,000) -> ㅓ(8748) -> * (0) -> ㅅ(786) -> *(0) -> *(0) = 39,534

Fig. 9 Total number of comparison using the simple search algorithm (incomplete syllable)

ㅓ(30,000) -> ㅅ(7172) -> ㄱ(303) -> * (0) -> *(0) -> *(0) = 37,475

Fig.10 Total number of comparison using the proposed algorithm.(incomplete syllable)

Algorithm 1 shows the algorithm of character based hangeul search method using location-specific frequency of character.

Algorithm 1 : the algorithm of proposed method

Input : target word(*TW*), source dictionary(*SD*)
Output : a successful match words
Method:

```
1: begin
2:    ascending sort in order to frequency
3:    for end of TW
4:       For bottom of SD_i
5:          if TW_i = SW_i then
6:             SW_i -> { SD_i+1 }
7:       end for
8:    end for
9: end
```

IV. PERFORMANCE STUDY

In this section, we perform a evaluation of proposed method in this paper.

Fig. 11 shows the number of words to be compared in each step of searching when we search the word, '검색' in the sample dictionary using the simple search algorithm and the proposed algorithm.

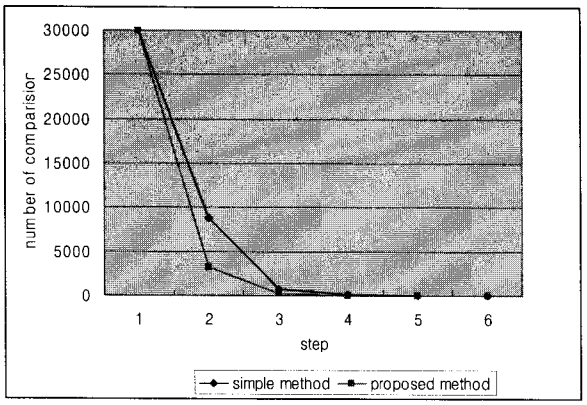


Fig. 11 The number of words to be compared in each step of searching

As shown in the Fig. 11, the number of words being compared in the next step is reduced rapidly in the proposed method more than the simple search. Therefore, we can see that the total number of comparison using the proposed method is less than that using the simple search method.

Fig. 12 shows the average number of comparison in accordance with the number of syllables. For this experiment, first, we chose each 10 word that is made up of 2,3 and 4 syllables from sample dictionary at random and measured the total comparison number in every words.

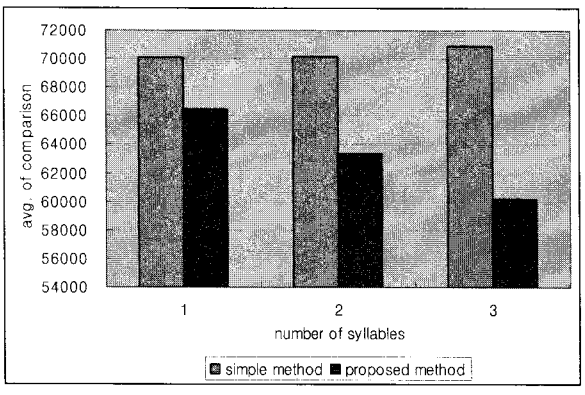


Fig.12 The average number of comparison in accordance with the number of syllables

As shown in the Fig. 12, we can find that every case is efficient in case of using the proposed method more than using the simple method, and also find that the more the length of target word is long, the more the efficiency of the proposed algorithm is better, because the frequency of character at rear position of the word is lower than front position.

V. CONCLUSION

In this paper, we proposed the character based hangeul search method using the location-specific frequency information. For designing this method, we defined search patterns of target word can be used in this method first, and examined the location-specific frequency of hangeul character. Finally, we could find that every case is efficient in case of using the proposed method more than using the simple method through experiments.

Character based hangeul search can be used in a various applications like human name search or firm name search in navigation system.

In the future, we plan to design of new character based hangeul index system using the proposed method

ACKNOWLEDGMENT

This work was supported by Dong-eui University Research Grant(2008AA184).

REFERENCES

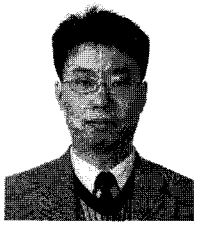
[1] Gollapudi, S. and Panigrahy, R. ,"A Dictionary for Approximate String Search and Longest Prefix Search," CIKM INTERNATIONAL CONFERENCE CD-ROM EDITION, Vol.15, pp.768-775, 2006

[2] Ronnblom, J., "High-error approximate dictionary search using estimate hash comparisons," Software:Practice and Experience, Vol.37 No.10, pp.1047-1059, 2007

[3] Paolo Ferragina and Roberto Grossi. "The String B-Tree: a new data structure for string search in external memory and its applications," Journal of the ACM, vol. 46(2), pp.236-280, 1999

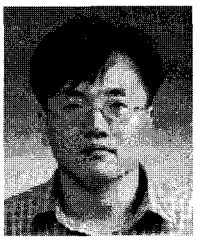
[4] Kyeonghwan Kim, "High-Speed Korean Address Searching System for Efficient Delivery Point Code Generation," The KIPS Transaction, Vol.8, No.3, pp.273-284, 2001

[5] Junho Lee, "A Method of Retrieving Romanized Korean Names,", The Industrial Technology Research, Vol.31, pp.181-189, 2001
[6] Junghwa Lee, "A Study of the framework of search patterns for Hangul characters and its relationship with Hangeul code for Hangeul Character based Index", the Journal of the Korea institute of maritime information and communication sciences, Vol. 11, No. 6, pp. 1083-1088, 2007
[7] KSA, "KS X 1001:2004, Code for information interchange(Hangeul and hanja)" , 2004
[8] The Unicode Consortium, "The Unicode Standard, Version 5.0", Addison-Wesley Professional, 2006.
[9] ISO/IEC 10646-1:2003. "Information technology-Universal Multiple-Octet Coded Character Set (UCS)", 2003.
[10] Gyongsok Kim, "Hangeul Story in Computer second edition", publishing department of Pusan National University, 1999
[11] CheolSu Kim and Yangbeom Kim, "Korean Dictionary Electronic Dictionary Statistical Information Processing Syllable", Journal of the Korea Contents Society, Vol.7, No.6, pp.60-68, 2007



Jung-hwa Lee received his B.S. and the Ph.D. degrees at the Department of Computer Science from Pusan National University, Korea, in 1995 and 2001, respectively. He is a professor at the Department of Computer Software engineering, Dongeui University in Korea. His research interests include database, Hangeul information processing, and semantic web, etc

Jong-min Lee received the B.S. degree in computer engineering from Kyungpook National University, Korea, in 1992, and the M.S. and the Ph.D. degrees in computer science from KAIST in 1994 and 2000, respectively. Since 2002 he has been a faculty member of the Department of Computer Software Engineering, Dong-Eui University. From Feb. 2005 to Feb. 2006, he was a research associate at the University of California, Santa Cruz. His research interests include mobility management in wireless networks, routing in ad hoc networks and sensor networks.



Seong-woo Kim received his B.S., M.S., and Ph.D. degrees in Electrical Engineering from Korea Advanced Institute of Science and Technology (KAIST), in 1991, 1993, and 1999 respectively. He worked at Electronics and Telecommunications Research Institute (ETRI) from 1999 to 2001. Since 2002, he has been a professor at Dongeui University. His research interest is in embedded operating system and sensor network.