Conflict Code Rate Reduction for a New Phoneme-based Chinese Input Method

Cheng-Huang Tung and En-Yih Jean*
Department of Information Technology, National Pintung Institute of Commerce, Pingtung, Taiwan 900
e-mail: chdong@npic.edu.tw tel:886-8-7238700ext2700
*Department of Information Science, University, Tamsui, Taipei, Taiwan 251
e-mail: ey.chien@msa.hinet.net tel:886-2-26212121

Abstract— We have proposed a new phoneme-based Chinese input method with low conflict code rate, 24.7%, where the features we extract are all phonetic symbols. In this paper, we propose a modified phoneme-based Chinese input method to further reduce the conflict code rate significantly. We first propose a new reduction rule that will extract one feature phonetic symbol for each of the selected extended radicals in different manners. Two measurements will be considered to evaluate the performance of the modified input method. First, since users must recall these selected extended radicals all the time, it is required that the number of selected extended radicals must be small. Second, if we apply the new reduction rule on these selected extended radicals, the conflict code rate of the modified phoneme-based input method must be as low as possible. In the experiments, the number of selected extended radicals, n, is defined to be a small number(n=10). In the training phase, we apply an n-stage hill-climbing method to select these n extended radicals. At each stage, the hill-climbing method selects an extended radical under the constraint of minimizing the conflict code rate. After the training phase is finished, we construct a modified phoneme-based input method whose conflict code rate, 13.5%, is much lower than 24.7%.

Keywords: Phoneme-based Chinese input method, effective phonetic sequence, extended radical, conflict code rate, hill-climbing method.

1. Introduction

The research about Chinese characters is very important and practical. There have been a great deal of recent efforts in solving the Chinese input methods [1-13]. Chinese characters are non-alphabetic and two-dimensional in structure[1]. There are thousands of commonly used

Chinese characters, in which each character is ideographically different. Besides being ideographic, Chinese has the homophone property. That is, almost every Chinese character can be pronounced as the phonemes of other characters. In our knowledge, there are more than 10,000 commonly-used characters but only 1,351 effective phonetic sequences[2].

Generally speaking, there are two major categories of methods to code Chinese characters. The first category is the phonetic coding methods[3-6] and the second one is the radical coding methods[7-12]. Like English pronunciation, Chinese has the standard phonetic spelling system. Accordingly, the phonetic coding strategy utilizes a phonetic spelling system to encode Chinese characters. The well-known advantage of the phonetic input method is that it is very natural, but the method obviously suffers the problem of homonyms. That is, the Chinese phonetic input system produces a large number of homonyms for each effective phonetic sequence. In the worst case, there are more than 100 homonyms for a specific phonetic sequence[8,9]. Although the phonetic method is undoubtedly very easy to learn and to use by green hand users, the homonyms problem results in a high conflict code rate and makes the input of Chinese characters inefficient.

The radical coding strategy[7-12] uses the fundamental components of Chinese characters as the basic elements to encode Chinese characters. The encoding methods based on this strategy usually have the advantage of efficiency in encoding characters. A well-designed radical coding method usually results in a low conflict code rate, i.e. a one to one mapping between most of the input codes and the Chinese characters. However, a small set of radicals is not enough to construct thousands of Chinese characters properly. Thus, these methods usually divide a Chinese character into a number of radicals by applying unnatural heuristic rules. When users perform these Chinese input methods, they have to remember not only the radicals heuristically defined by the designer, but also the way how a Chinese character is divided.

To resolve the disadvantages mentioned above, we propose a new phoneme-based Chinese input method with a low conflict code rate, 24.7%, and all input factors are phonetic symbols [13]. First, we retain at most two key phonetic symbols of a character as the first part of features; that is, we reduce an effective phonetic sequence to a reduced phonetic sequence whose length is not more than 2. Second, to overcome the difficulty of decomposing characters, we define an extended radical set, which includes 5,401 frequently-used Chinese characters, radicals, and seven primitive strokes. Third, according to the writing sequence of a Chinese character, we extract two extended radicals which must include the first and last strokes respectively, and then select the first phonetic symbol of each extended radical as the phonetic feature symbol. In this way, we can obtain two phonetic feature symbols from the writing sequence of the character. We append these two phonetic feature symbols to a reduced phonetic sequence to form a new phonetic code, whose maximal length is therefore 4.

In this paper, we will propose a modified phoneme-based Chinese input method to further reduce the conflict code rate significantly. By analyzing characters with the same phonetic code, we find out that we can reduce the conflict code rate by defining a new feature reduction rule for processing some specific extended radicals. We first describe a new reduction rule to extract a feature phonetic symbol for an extended radical in different ways. Then, we will apply the hill-climbing method to identify a set of extended radicals that are processed by the new reduction rule. Two measurements will be considered to evaluate the performance of the modified input method. First, since users must use these specific extended radicals all the time, the number of selected extended radicals must be as small as possible. Second, the conflict code rate of the modified phoneme-based input method must be as low as possible. In the experiments, the number of these selected extended radicals, n, is defined to be 10. At the training phase, we apply the hill-climbing method to select 10 extended radicals under the evaluation of minimizing the conflict code rate. After the training phase is finished, we obtain a

modified phoneme-based input method whose conflict code rate is reduced from 24.7% to 13.5%.

In Section 2, we review the proposed phoneme-based input method. In Section 3, we propose the modification of the input method. Section 4 applies a hill-climbing method to achieve the modified input method with minimal conflict code rate. Section 5 discusses the performance of the modified phoneme-based input method. Section 6 concludes this paper.

2. Review of the Phoneme-based Chinese Input Method

2.1 Reduction of Effective Phonetic Sequences

Let the 5,401 frequently-used Chinese characters be $Ch_1,Ch_2,\ldots,Ch_{5401}$(="一, 乙,…,籲"), and an effective phonetic sequence(EPS) be the phonetic sequence which some Chinese character is pronounced as. The length of an EPS ranges from one to four. For example, the phonetic sequence "ㄐㄧㄣˋ" is an EPS, because it is the pronunciation of character "進." However, "ㄐㄣˋ" is not an EPS, because no characters can be pronounced as "ㄐㄣˋ." The homonym set of EPS EPS_i, $HO(EPS_i)$, is the set of characters with the same EPS EPS_i, and the phonetic sequence set of Ch_i, $PH(Ch_i)$, is the set of phonetic sequences which character Ch_i can be pronounced as. For example, the homonym set of EPS "ㄍㄨㄥˋ" is $HO($ㄍㄨㄥˋ$)$={共,供,貢}, and the phonetic set of character "供" is $PH($供$)$={ㄍㄨㄥ, ㄍㄨㄥˋ}.

To improve the performance of the phonetic input method, we first reduce the length of an EPS to not more than 2; that is, each EPS will be reduced to a reduced effective phonetic sequence(REPS) containing not more than two phonetic symbols. Let $S_1...S_i$ be the EPS of a character, and SET_{tone} be the set of tone symbols, {ˊ, ˇ, ˋ, •}. The reduction rule *RED1* for EPS $S_1...S_i$ is as follows.

Reduction rule $RED1(S_1...S_i)$:

if $i \leqq 2$ then REPS remains $S_1...S_i$

else if i=3

if S_3 is in SET_{tone} then REPS is S_1S_2

else REPS is S_1S_3

else if i=4 then REPS is S_1S_3

Table 1 gives some examples of reducing EPSs by applying the reduction rule. Similarly, we define that the homonym set of $REPS_i$, $RHO(REPS_i)$, is the set of characters whose REPSs contain $REPS_i$, and that $RPH(Ch_i)$ is the REPS set of Ch_i. For example, the homonym set of REPS "ㄙㄢ" is RHO(ㄙㄢ)={三,參,傘,散,酸,算}, and the REPS set of "角" is RPH(角)={ㄐㄠ,ㄐㄝ}.

Table 1. Examples of reducing the length of EPSs.

character	EPS	REPS
醫	ㄧ	ㄧ
遇	ㄩˋ	ㄩˋ
來	ㄌ ㄤˊ	ㄌㄤ
金	ㄐㄧㄣ	ㄐㄣ
進	ㄐㄧㄣˋ	ㄐㄣ

2.2. Decomposition of Chinese Characters

In this section, we first describe the process of extracting two extended radicals from the writing sequence of a character. Following the idea that Chinese characters are constructed recursively, we define that the extended radical set includes:

1. 5401 frequently-used Chinese characters,
2. standard Chinese radicals, listed in a Chinese dictionary,
3. seven primitive strokes used for the characters that can't be decomposed properly.

Table 2 lists the radicals which are not frequently-used characters or are in different writing style. For example, "冖(ㄇㄧˋ)" means that "冖" is pronounced as "ㄇㄧˋ", and "氵(水, ㄕㄨㄟˇ)" means that "氵" is the alternative writing style of character "水" and is pronounced as

“ㄕㄨㄟㄧ.” To decompose characters properly, the seven primitive strokes shown in Table 3 are also included in the extended radical set.

Table 2. The extended radicals which are not frequently-used characters or are in alternative writing style.

# of strokes	radicals with phonetic symbols
2	亠(ㄊㄡˊ), 亻(人,ㄖㄣˊ), 儿(ㄖㄣˊ), 冂(ㄐㄩㄥ), 冖(ㄇㄧˋ), 冫(冰,ㄅㄧㄥ), 刂(刀,ㄉㄠ), 勹(ㄅㄠ),卩(ㄦ,ㄐㄧㄝˊ), 厂(ㄏㄢˇ)
3	氵(水, ㄕㄨㄟˇ), 夂(ㄓㄡˇ), 宀(ㄇㄧㄢˊ), 尢(ㄨㄤ), 屮(ㄔㄠˋ), 巛(川,ㄔㄨㄢ), 广(ㄢ), 廴(ㄧㄣˇ), 彑(ㄧㄣˇ), 彡(ㄕㄢ), 彳(ㄔˋ), 忄(心,ㄒㄧㄣ), 扌(手,ㄕㄡˇ), 犭(犬,ㄑㄩㄢˇ), 阝(阝,儿ˇ)
4	灬(心,ㄒㄧㄣ),攵(ㄆㄨ),下(ㄨㄛˋ),殳(ㄕㄨ), 气(ㄑㄧˋ),灬(火,ㄏㄨㄛˇ), 爫(爪,ㄓㄨㄚˇ), 牜(ㄋㄧㄡˊ), 礻(ㄕˋ), 罒(ㄨㄤˇ), 艹(草,ㄘㄠˇ), 辶(ㄔㄨㄛˋ)
5	疋(ㄆㄧˇ), 疒(ㄔㄨㄤˊ), 癶(ㄅㄛ), 内(ㄖㄡˋ), 氺(水,ㄕㄨㄟˇ), 罒(罒,ㄨㄤˇ), 衤(衣,一)
6	虍(ㄏㄨ), 襾(ㄧㄚˋ)
7	豸(ㄓˋ), 釆(ㄅㄧㄢˋ)
8	隶(ㄉㄞˋ)
10	髟(ㄅㄧㄠ), 鬯(ㄔㄤˋ), 鬲(ㄍㄜˊ)
13	黽(ㄇㄧㄣˇ)
17	龠(ㄩㄝˋ)

Table 3. The seven primitive strokes defined as extended radicals.

primitive stroke	丶	ノ	丶	丨	亅	乛	丿
EPS	ㄋㄚˋ	ㄆㄧㄝˇ	ㄉㄧㄢˇ	ㄕㄨˋ	ㄍㄡ	ㄨㄢ	ㄊㄧㄥˊ

Next, we introduce the method of decomposing a Chinese character according to its writing sequence. We assume that the writing strokes of a Chinese character Ch_i are T_1 , T_2,..., T_k. If strokes T_1 , T_2,..., T_i and T_j , T_{j+1},..., T_k form extended radicals RAD_a and RAD_b, respectively, then extended radicals RAD_a and RAD_b can be extracted from character Ch_i. However, because there may be many alternatives of decomposing a character according to the writing sequence, we briefly define the extraction rules as follows.

Extraction Rule *ER1*(T_1 , T_2,..., T_k):

if strokes T_1 , T_2,..., T_i, $1<i<k$, form extended radical RAD_a

and Tp doesn't intersect with Tq, $1 \le p \le i$, $i+1 \le q \le k$

then RAD_a can be extracted

else primitive stroke T_1 is extracted

Extraction Rule *ER2*(T_1 , T_2,..., T_k):

if strokes T_j , T_{j+1},..., T_k, $1<j<k$, form extended radical RAD_b

and Tp doesn't intersect with Tq, $j \le p \le k$, $1 \le q \le j$-1

then RAD_b can be extracted

else primitive stroke T_k is extracted

Extraction Rule *ER3*(T_1 , T_2,..., T_k):

maximize $i+(k-j+1)$ where $i<j$, and strokes T_1, T_2,..., T_i and T_j , T_{j+1},..., T_k form

extended radicals RAD_a and RAD_b respectively

In Table 4, we give some examples to demonstrate the three extraction rules. *ER1* and *ER2* mean that the extracted extended radicals can't intersect any other stroke. For example, the first extended radical of "中" is not "口", because "口" intersects the last stroke "丨." Therefore, the first extended radical of "中" will be "丨", which is a primitive stroke. Similarly, the last extended radical of "中" is "丨." *ER3* means that we will maximize the stroke count of these two extended radicals. For example, the extraction alternatives of "資" can be "次,貝", "冫,貝", and "次,八", etc. If we apply rule *ER3*, "次,貝" will be the best extraction alternative for "資." Accordingly, we will apply *ER1, ER2*, and *ER3* simultaneously to extract extended radicals from Chinese characters.

Table. 4. Some examples of extracting extended radicals for Chinese characters.

character	first extended radical	second extended radical
中	丨	丨
資	次	貝

Next, we define reduction rule RED2 as follows to extract the phonetic feature for each extended radical.

Reduction rule *RED2*(ER_i):

initialize feature set *FS* to be empty

for each EPS $S_1S_2...S_k$ in $PH(ER_i)$

if S_1 is not in feature set *FS* then add S_1 into *FS*

output *FS*

Generally, the size of the phonetic feature set of an extended radical is equal to one. For example, the phonetic feature set of extended radical "弓"("ㄍㄨㄥ") is {ㄍ}. Merely a small part of extended radicals own phonetic feature sets whose sizes are larger than one. For example, the feature set of extended radical "虫" ("ㄏㄨㄟˇ", "ㄔㄨㄥˊ") is {ㄏ, ㄔ}. For character Ch_i, two extended radicals RAD_a and RAD_b can be extracted from its writing sequence by applying rules *ER1*, *ER2* and *ER3* simultaneously. Then, we apply *RED2* to obtain one phonetic feature set for each of RAD_a and RAD_b. If $REPS_j$ is in $RPH(Ch_i)$ and phonetic features S_p and S_q are in the phonetic feature sets for RAD_a and RAD_b, the phonetic code of Ch_i is $S_p+S_q+REPS_j$. Table 5 gives some examples of extracting phonetic codes for characters.

Table. 5. Some examples of extracting phonetic code for characters.

character	first extended radical	second extended radical	character	phonetic code
虹	虫 (ㄏㄨㄟˇ, ㄔㄨㄥˊ)	工(ㄍㄨㄥ)	虹(ㄏㄨㄥˊ)	ㄏㄍㄏㄥ, ㄔㄍㄏㄥ
他	亻(ㄖㄣˊ)	也(ㄧㄝˇ)	他(ㄊㄚ)	ㄖㄧㄊㄚ
事	一(ㄧ)	亅(ㄍㄡ)	事(ㄕˋ)	ㄧㄍㄕˋ
恩	因(ㄧㄣ)	心(ㄒㄧㄣ)	恩(ㄣ)	ㄧㄒㄣ

3. The Modified Phoneme-based Input Method

In Section 2, we have described a new Chinese input method, in which all input features are phonetic symbols, and the conflict code rate is low, 24.7%[13]. In this section, we further

propose a modified phoneme-based Chinese input method to reduce the conflict code rate as much as possible. By analyzing characters with the same code, we find that some modification can be performed for resolving the characters with the same code. For example, characters "餃" and "攪" are mapped to the same phonetic code "ㄕㄐㄐㄠ" as follows.

- 餃→食(ㄕˊ)交(ㄐㄧㄠ)餃(ㄐㄧㄠˇ)→ㄕㄐㄐㄠ
- 攪→扌(ㄕㄡˇ)覺(ㄐㄩㄝˊ)攪(ㄐㄧㄠˇ)→ㄕㄐㄐㄠ

The first extended radicals of characters "餃" and "攪" are "食" and "扌", and the second extended radicals are "交" and "覺". We extract "ㄕ" for both "食" and "扌", and "ㄐ" for both "交" and "覺." Characters "食" and "扌" are pronounced differently, but the extracted feature symbols are the same, "ㄕ." In this case, if we define that the feature symbol of extended radical "扌"(ㄕㄡˇ) is "ㄡ" instead of "ㄕ", the codes for characters "餃" and "攪" shown below will be different.

- 餃→食(ㄕˊ)交(ㄐㄧㄠ)餃(ㄐㄧㄠˇ)→ㄕㄐㄐㄠ
- 攪→扌(ㄕㄡˇ)覺(ㄐㄩㄝˊ)攪(ㄐㄧㄠˇ)→ㄡㄐㄐㄠ

Accordingly, we propose a new reduction rule *RED3* to extract a feature symbol for each element of the selected extended radical set, $SER=\{ER_{i_1}, ER_{i_2}, \ldots, ER_{i_n}\}$, where these extended radicals will be determined in the Section 4.

Reduction rule *RED3*(ER_i):

initialize feature set *FS* to be empty

for each EPS $S_1S_2...S_k$ in $PH(ER_i)$

if S_k is in SET_{tone} then add S_{k-1} into feature set *FS*

else add S_k into feature set *FS*

output *FS*

For example, we assume that the set of selected extended radicals is {水,木,金}. After we apply

reduction rule *RED3* to process extended radicals “水”, “木”, and “金”, we obtain feature sets {ㄟ},{ㄨ} and {ㄣ} respectively. Before we can apply the modified phoneme-based input method, the codes of 5,401 frequently-used Chinese characters must be learned in template database *TDB* in advance. Then, we can match an input code with the codes in database *TDB*, and output the matched characters. The algorithm of creating template database *TDB* is shown as follows.

Algorithm 1: create template phonetic database *TDB*

Input: 5,401 frequently-used Chinese characters

Output: database *TDB* containing the codes of 5401 frequently-used Chinese characters

begin

TDB={}

for i=1 to 5401

apply *ER1, ER2, ER3* to obtain RAD_a and RAD_b from Ch_i

if RAD_a is in *SER* then $FS_1=RED3(RAD_a)$

else $FS_1=RED2(RAD_a)$

if RAD_b is in *SER* then $FS_2=RED3(RAD_b)$

else $FS_2=RED2(RAD_b)$

for each $REPS_j$ in $RPH(Ch_i)$

for each S_p in FS_1

for each S_q in FS_2

if code $S_p+S_q+REPS_j$ is not in database *TDB* then

add $S_p+S_q+REPS_j \rightarrow \{Ch_i\}$ to *TDB*

else add Ch_i to referred character set $\{Ch_r,\ldots,Ch_s\}$ of $S_p+S_q+REPS_j$

end.

Let $Code_i$ be a code in database TDB, and the referred character set be $CREF(Code_i)$. The

occurrence probability, $P(Code_i)$, will be $P(Code_i) = \sum_{Ch_j} P(Ch_j, Code_i)$. Conflict code rate of the modified phoneme-based Chinese input method will be:

$$CCR(TDB) = \sum_{Code_i} P(Code_i) \text{ where } \left|CREF(Code_i)\right| > 1.$$

4. The Hill-Climbing Method for Minimizing Conflict Code Rate

In Sectin 3, a modified phoneme-based Chinese input method has been proposed. In this section, to determine the set of selected extended radicals $SER=\{ER_{i_1}, ER_{i_2}, \ldots, ER_{i_n}\}$, we propose a hill-climbing method with the constraint of minimizing the conflict code rate approximately. We assume that there are totally m extended radicals, ER_1, ER_2,..., ER_m. Let the function of evaluating the conflict code rate of an input method be CCR', where $CCR'(ER_{i_q})$ means that the conflict code rate of the input method where $SER = SER \cup \{ER_{i_q}\}$. Initially, the set of selected extended radicals is empty; that is, $SER=\{\}$. At stage 1 shown in Fig. 1, there are m alternatives to select the first element of *SER*. The best alternative is to add extended radical ER_{i_q} into *SER* where $CCR'(ER_{i_q})$ is minimal; that is, we have $SER=\{ER_{i_q}\}$. Similarly, we can select the extended radical resulting in the lowest conflict code rate from m-1 alternatives at stage 2. Repeating this process, we can determine the n elements of *SER*.

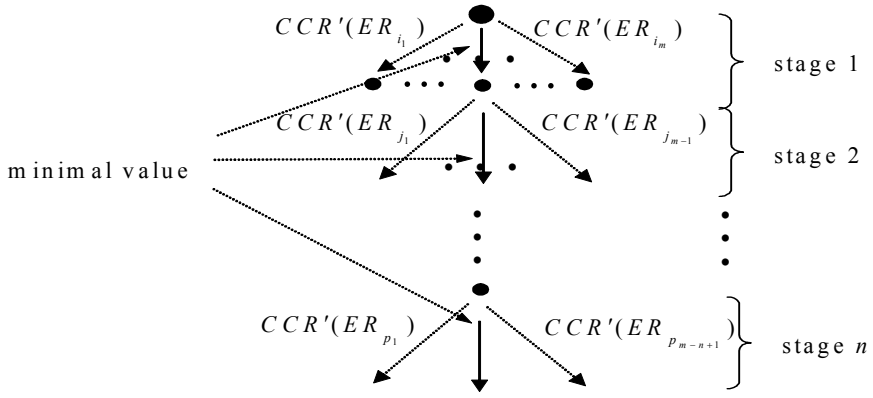

Fig. 1. The hill-climbing method of selecting n extended radicals out of m extended radicals.

Because the number of extended radicals is large(m>1000), it will be very time-consuming to perform the n-stage process. To speed up the process, we will merely measure the effects of the extended radicals with large occurrence probabilities in conflict codes. At each stage of the hill-climbing process, the k candidate extended radicals with the largest probability values in conflict codes will be evaluated. As shown in Fig. 2, an extended radical is determined from k candidates at each stage.

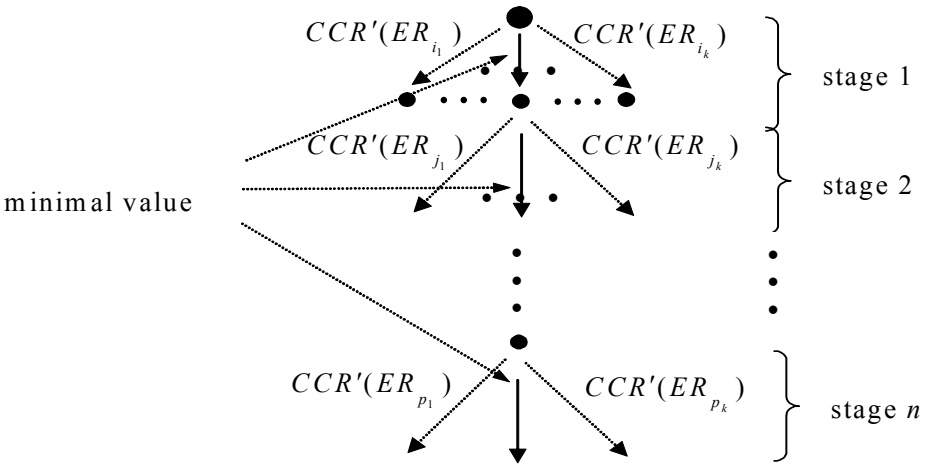

Fig. 2. The hill-climbing method of selecting n extended radicals out of m extended radicals where we have k candidate extended radicals with the largest probability values in conflict codes at each stage.

To determine the k candidate extended radicals at each stage, we will define the way of calculating the occurrence probability of extended radicals. In the following, we show the probability measurement by examples. In Fig. 3(a), we show some code mappings in template database *TDB*. For example, code "ㄕㄐㄐㄢ" refers to two characters "捲" and "揀." We assume that the occurrence probability of each of 5,401 frequently-used characters is $\frac{1}{5401}$; i.e. P(螫)=P(捲)= P(揀)=$\frac{1}{5401}$. In Fig. 3(b), "ㄕㄏㄓㄜ→{(螫,赦虫,$\frac{1}{5401*2}$)}" means that "赦" and "虫" are the extended radicals extracted from character "螫." We have P(ㄕㄏㄓㄜ,

螯)=$\frac{1}{5401*2}$, since character “螯” has two EPSs “ㄕㄏㄓㄜ” and “ㄕㄔㄓㄜ.” Then, we define the accumulative probability functions P_{First} and P_{Last} concerning probability values of first and second extended radicals respectively. In Fig. 3(c), we show the values of P_{First} and P_{Last} calculated from Fig. 3(b). For example, we have P_{First}(ㄕㄐㄐㄢ,手)=$\frac{2}{5401}$, since “手” is the first extended radical of characters “捲” and “揀.”

ㄕㄏㄓㄜ→{螯}

ㄕㄔㄓㄜ→{螯}

ㄕㄐㄐㄢ→{捲揀}

(a)

ㄕㄏㄓㄜ→{(螯,赦虫,$\frac{1}{5401*2}$)}

ㄕㄔㄓㄜ→{(螯,赦虫,$\frac{1}{5401*2}$)}

ㄕㄐㄐㄢ→{(捲,手卷,$\frac{1}{5401}$), (揀,手柬, $\frac{1}{5401}$)}

(b)

P_{First}(ㄕㄏㄓㄜ,赦)=$\frac{1}{5401*2}$, P_{Last}(ㄕㄏㄓㄜ,虫)=$\frac{1}{5401*2}$,

P_{First}(ㄕㄔㄓㄜ,赦)=$\frac{1}{5401*2}$, P_{Last}(ㄕㄔㄓㄜ,虫)=$\frac{1}{5401*2}$,

P_{First}(ㄕㄐㄐㄢ,手)=$\frac{2}{5401}$, P_{Last}(ㄕㄐㄐㄢ,卷)=$\frac{1}{5401}$, P_{Last}(ㄕㄐㄐㄢ,柬)=$\frac{1}{5401}$

(c)

Fig. 3. (a) Codes referring to characters. (b) Code mappings involving probability occurrence values and extended radicals. (c) Accumulative probability values of code and extended radical.

In the following, we propose the algorithm of determining k candidate extended radicals at each stage according to the occurrence probability of extended radicals in conflict codes.

Algorithm 2: determine *k* candidate extended radicals with the largest probability values

Input: selected extended radical set *SER*, template database *TDB*

Output: *k* candidate extended radicals with the largest probability values

begin

initialize $P_{First}(ER_i) = 0$ and $P_{Last}(ER_i) = 0$ for all ER_i

for each code $Code_i$ in template database *TDB*

if size of referred set *RS* of code $Code_i$ in *TDB*>1

for each character Ch_j in referred set *RS*

apply *ER1, ER2, ER3* to extract ER_a, ER_b from Ch_j

$$P_{First}(ER_a) = P_{First}(ER_a) + P_{First}(Code_i, ER_a)$$

$$P_{Last}(ER_b) = P_{Last}(ER_b) + P_{Last}(Code_i, ER_b)$$

for each ER_j that is not in the selected extended radical set *SER*

$$P(ER_j) = P_{First}(ER_j) + P_{Last}(ER_j)$$

output the *k* extended radicals with the highest probability

end.

After algorithm 2 is performed, the *k* candidate extended radicals of each stage can be determined. In the following, we propose the algorithm of determining *n* selected extended radicals and the code-mapping database.

Algorithm 3: obtain *SER* containing *n* selected extended radicals, and the final code-mapping database *TDB*

Input: 5401 frequently-used Chinese characters

Output: *SER* and code-mapping database *TDB*

begin

$SER = \{\}$

call Algorithm 1 to create *TDB*

for *i*=1 to *n*

call Algorithm 2 to obtain the k candidate extended radicals

for each candidate extended radical ER_j

$SER = SER \cup \{ER_i\}$

call Algorithm 1 to obtain database TDB'

evaluate conflict code rate $CCR(TDB')$

$SER = SER - \{ER_i\}$

$SER = SER \cup \{ER_k\}$ where ER_k results into the lowest conflict code rate

call Algorithm 1 to create the final code-mapping database *TDB*

output *SER* and *TDB*

end.

5. Experimental Results

To make the input system easy to learn, we define that the number of selected extended radicals is a small number, n=10. In the training phase, we define that the number of candidate extended radicals at each stage of the hill-climbing process is k=80. The experimental results are shown in Table 6. After the 10 extended radicals have been determined, the conflict code rate can be reduced from 24.7% to 13.5%. Therefore, the proposed Chinese input method can be more effective and practical. At each stage, the first row shows the top-15 candidate extended radicals, the second row shows the accumulate probability in conflict codes, the third row shows the extracted features for the extended radicals, and the fourth row shows the conflict code rate if the extended radical is contained in the set of selected extended radicals. At each stage, the marked extended radical results in the lowest conflict code rate and is added to the set of selected extended radicals. Even though the top 80 candidate extended radicals are considered, the experimental results show that the largest rank of selected extended radicals is 11, which is still a small number. It means that k=80 will be large enough to determine the best choice at each stage. We summarize the selected extended radicals in Table 7.

To evaluate whether the proposed input method is easy to learn or not, we perform the following experiments. In Taiwan, the most popular radical coding input methods include “Tsang-jye”(“倉頡”), “Dah-yih”(“大易”) and “Wu-shia-mii”(“無蝦米”). Forty volunteers are required to learn each of these input methods and ours for half an hour. These volunteers are asked to write down the codes for a testing text consisted of 200 Chinese characters, and then we measure the accuracy of these codes. Table 8 lists the average code accuracy. The code accuracy values for “Tsang-jye”, “Dah-yih” and “Wu-shia-mii” are not more than 14%, and that for our method is higher than 71%. It shows that our method is easy to learn even though some modification has been involved in our input method.

6. Conclusion

Previously we have proposed a new phonetic-based Chinese input method with low conflict code rate, 24.7%, where the features we use are all phonetics. In this paper, we propose a modified phoneme-based Chinese input method to further reduce the conflict code rate significantly. The paper proposed a new feature extraction rule to extract features of extended radicals in different ways. However, which extended radicals should be processed by the new feature extraction rule will be determined by a hill-climbing method. At the training phase, each time we measured the first k extended radicals with high occurrence probability in the conflict characters. Then, an extended radical is determined from these k extended radicals. Repeating the process for n times, we can determine a selected extended radicals set, where these extended radicals are processed by the new feature extraction rule. After the training phase is finished, the conflict code rate can be reduced to 13.5%, which makes the input method more efficient.

Table 6. The top-15 candidate extended radicals with the conflict code rates at each stage, where the selected extended radicals are marked.

stage 1	水	手	人	木	口	草	糸	言	金	月	女	山	虫	目	疒
Prob.(%)	3.52	2.54	2.04	2.02	1.67	1.61	1.5	1.45	1.34	1.09	1.05	1.01	0.97	0.88	0.88
Feature	㇇	ㄡ	ㄣ	ㄨ	ㄡ	ㄠ	ㄧ	ㄢ	ㄣ	ㄝ	ㄩ	ㄢ	ㄥ	ㄨ	ㄤ
CCR(%)	21.8	22.5	23.9	24	24.7	24.7	24.8	23.6	24.2	24.3	24.4	24	23.8	24.2	24.2
stage 2	人	木	手	口	草	糸	言	金	月	女	虫	山	疒	目	心
Prob.(%)	1.94	1.91	1.8	1.57	1.51	1.4	1.35	1.24	0.95	0.95	0.87	0.84	0.78	0.75	0.72
Feature	ㄣ	ㄨ	ㄡ	ㄡ	ㄠ	ㄧ	ㄢ	ㄣ	ㄝ	ㄩ	ㄥ	ㄢ	ㄤ	ㄨ	ㄣ
CCR(%)	21	21.1	20.3	21.8	21.8	22	20.7	21.3	21.4	21.5	21	21.2	21.4	21.4	21.5
stage 3	人	木	口	草	糸	言	金	月	女	虫	疒	目	心	火	馬
Prob.(%)	1.84	1.81	1.47	1.41	1.3	1.21	1.14	0.85	0.85	0.77	0.66	0.65	0.62	0.59	0.56
Feature	ㄣ	ㄨ	ㄡ	ㄠ	ㄧ	ㄢ	ㄣ	ㄝ	ㄋ	ㄥ	ㄤ	ㄨ	ㄣ	ㄛ	ㄚ
CCR(%)	19.4	19.5	21.4	20.3	20.4	18.9	19.7	19.8	19.5	19.4	19.8	19.8	19.9	19.8	19.8
stage 4	人	木	口	草	糸	金	月	女	虫	疒	目	心	火	馬	竹
Prob.(%)	1.84	1.79	1.45	1.41	1.3	1.12	0.85	0.85	0.77	0.66	0.65	0.6	0.57	0.56	0.55
Feature	ㄖ	ㄨ	ㄡ	ㄠ	ㄧ	ㄣ	ㄝ	ㄩ	ㄥ	ㄤ	ㄨ	ㄣ	ㄛ	ㄚ	ㄨ
CCR(%)	18.9	18.2	20.1	18.9	18.7	18.4	18.5	18.6	17.9	18.5	18.5	18.6	18.6	18.5	19.1
stage 5	人	木	口	草	糸	金	月	女	目	心	馬	竹	王	疒	山
Prob.(%)	1.79	1.74	1.4	1.33	1.23	1.07	0.8	0.8	0.6	0.55	0.51	0.5	0.49	0.45	0.45
Feature	ㄖ	ㄨ	ㄡ	ㄠ	ㄧ	ㄣ	ㄝ	ㄩ	ㄨ	ㄣ	ㄚ	ㄨ	ㄤ	ㄤ	ㄢ
CCR(%)	17.9	17	19	17.9	17.6	17.3	17.4	17.5	17.4	17.5	17.4	18.1	17.7	17.6	17.9
stage 6	人	口	草	金	糸	月	女	王	心	目	竹	馬	疒	山	足
Prob.(%)	1.77	1.4	1.33	1.07	0.87	0.8	0.8	0.73	0.55	0.51	0.5	0.48	0.45	0.45	0.45
Feature	ㄣ	ㄡ	ㄠ	ㄣ	ㄧ	ㄝ	ㄩ	ㄤ	ㄣ	ㄨ	ㄨ	ㄚ	ㄤ	ㄢ	ㄨ
CCR(%)	16	18.1	17	16.5	17.1	16.6	16.7	16.5	16.7	16.9	17.5	16.5	16.8	17	17.4
stage 7	口	草	金	糸	月	王	心	女	竹	目	馬	疒	山	足	ㄱ
Prob.(%)	1.4	1.33	1.07	0.87	0.8	0.73	0.55	0.53	0.5	0.49	0.46	0.45	0.45	0.45	0.42
Feature	ㄡ	ㄠ	ㄣ	ㄧ	ㄝ	ㄤ	ㄣ	ㄩ	ㄓ	ㄨ	ㄚ	ㄤ	ㄢ	ㄨ	ㄢ
CCR(%)	17.2	16	15.5	16.2	15.6	15.3	15.7	16.2	16	16	15.6	15.8	16.1	16.4	15.9
stage 8	口	草	金	糸	月	心	女	竹	目	疒	山	馬	足	ㄱ	門
Prob.(%)	1.4	1.33	1.07	0.87	0.8	0.55	0.53	0.5	0.49	0.45	0.45	0.45	0.45	0.4	0.38
Feature	ㄡ	ㄠ	ㄣ	ㄧ	ㄝ	ㄣ	ㄩ	ㄨ	ㄨ	ㄤ	ㄢ	ㄚ	ㄨ	ㄨ	ㄣ
CCR(%)	16.5	15.3	14.6	15.5	14.9	15	15.5	15.8	15.2	15.4	15.4	14.9	15.6	15.3	15
stage 9	口	草	糸	月	心	女	竹	目	疒	山	馬	ㄱ	門	足	宀
Prob.(%)	1.35	1.28	0.82	0.75	0.5	0.48	0.45	0.44	0.4	0.4	0.4	0.35	0.33	0.31	0.3
Feature	ㄡ	ㄠ	ㄧ	ㄝ	ㄣ	ㄩ	ㄨ	ㄨ	ㄤ	ㄢ	ㄚ	ㄢ	ㄣ	ㄨ	ㄢ
CCR(%)	15.8	14.6	14.8	14.1	15.1	14.9	15.1	14.6	14.7	14.7	14.2	14.5	14.5	15.1	14.5
stage 10	口	草	糸	心	竹	女	目	疒	山	ㄱ	門	足	土	頁	宀

Table 7. The selected extended radicals with the phonetic feature symbols.

Extended radicals	水	手	言	虫	木	人	王	金	月	門
Phonetic sequence	ㄕㄟˇ	ㄕㄡˇ	ㄧㄢˊ	ㄔㄨㄥˊ ㄏㄨㄟˇ	ㄇㄨˋ	ㄖㄣˊ	ㄨㄤˊ	ㄐㄧㄣ	ㄩㄝˋ	ㄇㄣˊ
Feature symbol	ㄟ	ㄡ	ㄢ	ㄥ，ㄟ	ㄨ	ㄣ	ㄤ	ㄣ	ㄝ	ㄣ

Table 8. The average accuracy values for codes extracted by 40 volunteers, who learn every input method for half an hour.

Input method	Tsang-jye	Dah-yih	Wu-shia-mii	our method
code accuracy	11.2%	13.5%	12.4%	71.2%

References

1. S. Kwong, H. Wong and Y.S. Yu(1991), An effective method for storing and retrieval of Chinese characters, Proc. of Int. Conf. on Information Engineering, Singapore, pp. 368-376.

2. Wellington C. P. Yu et al.(1988), A historical advancement of Chinese language computing, Computer Processing of Chinese and Oriental Languages, vol. 4, No. 1, pp. 57-81.

3. S.K. Wan, H. Saitou and K.I. Mori(1982), Experiment on PINYI-HANZI conversion Chinese word processor, Computer Processing of Chinese and Oriental Languages, Vol. 1, No. 4, pp.213-224.

4. S.P. Zhou(1983), A proposal on Chinese PINYI encoding scheme, Proc. of Second National Conf. on Chinese Information Processing (in Chinese).

5. X. Cao and C.Y. Suen(1987), A new phonetic and ideographic coding Technique for Chinese information processing, Computer Processing of Chinese and Oriental Languages,

vol. 3, No. 2.

6. H.Y. Gu(1994), A Chinese-character inputting system using a new type of phonetic keyboard and a compound Markov language model, Proc. of ROCLINGC, pp.253-262.

7. S. Kwong, Y.K.Chan and E. Lee(1994), A Postprocessing to reduce conflict code rate for Chinese input methods, Proc. of Computer Processing of Chinese and Oriental Languages, pp.82-85.

8. C.K. Chen and R.W. Gong(1984), Evaluation of Chinese input methods, Computer Processing and Oriental Language, Vol. 1, No. 4, pp.236-247.

9. T.Y. Kiang and T.H. Cheng(1982), A New Chinese indexing system based on separation of character into main and subordinate components, Proc. of Int. Computer Symposium, Vol. 1, pp.131-141.

10. C.Y. Suen and E.M. Huang(1984), Computational analysis of the structural compositions of frequently used Chinese characters, Computer Processing of Computer Processing of Chinese and Oriental Languages, vol. 1, No. 3.

11. C.C. Shieh(1987), Evaluation report of Chinese input methods and input devices, Institute for Information Industry, Taipei, Taiwan.

12. K.J. Chen(1993), A mathematical model for Chinese Input, Computer Processing of Chinese and Oriental Language, vol. 7, pp.75-84.

13. E. Y. Jean and C. H. Tung(2000), “A Phoneme-based Chinese Input Method with Low Conflict Code Rate”, International Journal of Computer Processing of Oriental Languages, Vol. 13, No. 4, 333-349.