A progress report of solving the missing character problem at Academia Sinica

Translated by Mr. Hsueh-hsiang Lu, Computing Centre, Academia Sinica

Abstract

There are characters whose shapes can not be found with interchange code while manipulating Chinese characters, especially in ancient books, with computers. To preserve these characters, a commonly used approach is to build up their shapes in user defined area. But it is not cost-effective, neither did it solve the problem, for instance, the workload greatly increased by recording the hundreds of thousands of new characters; the difficulty of searching these characters by index; data error or unreadable document caused by possible duplicate code during information sharing. This is what we called ``missing character'' problem.

The missing character problem is a nightmare currently in the area with Chinese cultural heritage. Whenever names, places, history related to Chinese characters occur, there's serious missing character, an international issue everyone cares about. With fourteen-year experience of electronic book development, Academia Sinica(Taipei) has accumulated more than 9,600 missing characters currently, showing the seriousness and urgency of missing character problem. This article introduces the solution to this problem both theoretically and pragmatically.

The approach we adopt is to express the structure and attribute of characters based on character shape database in which the character is expressed by the combination of components and roots. We has built up the structure formulation equation for more than 40,000 character shapes. Usually, there are no input codes, interchange codes, or shapes stored in computers when characters does not exist, and thus they can not be processed. However, character shape database can be utilized to provide the manipulation of input, description, recognition, and lookup for missing characters.

Theoretically, what the character shape database expresses is the character derivation, meaning that each character root generates a character family tree. Among this family tree, the description for the structure formulation equation is expressed by one of three combination methods, i.e. vertical, horizontal, or inclusive. The character on the non-terminal node of family tree is called ``component,'' from which all character variations are constructed. There are 2,370 components in our system; 625 of them are roots, and 1,745 are non-roots (1,429 belongs to Big-5 code, and 326 can excluded.) Based on this architecture, 96% plus of 9,600 non-existing characters described before can be constructed. So are the rest of them, provided more ``components'' are added. More details are described in this article for character insufficiency analysis.

In practical application, we will base on this approach to modify data input and full-text indexing mechanism in our system and this will be explained in this article as well.

Preface

There are characters whose shapes can not be found with interchange code while manipulating Chinese characters with computers. This is what we call ``missing characters" problem and extremely difficult to handle with mass data processing in national database and ancient books. To display these missing characters, a commonly used approach is to build up their shapes in users defined area. But it is not cost-effective, neither did it solve the problem such as the workload greatly increased by entering the hundreds of thousands of new characters; the difficulty of searching these characters by index; data error or unreadable documents caused by possibly duplicate code during information sharing; even data sharing problem caused by prohibition of changing characters with user defined area in local area network.

Academia Sinica has 14-year experience of electronic book development and is famous for its Full-Text Database System, totally more than 140,000,000 characters in operation(refer to PNC paper written by 黃寬重 and 劉增貴.) The techniques used in this system are all designed by staff in Academia Sinica including structure of Full Text Database, content mark-up, data entry management, missing characters management, etc. However, more than 9,600 missing characters have been accumulated in the database currently, showing the seriousness and urgency of missing character problem. This article introduces the solution to this problem both theoretically and pragmatically.

Approach to expressing missing characters

The theory for solving missing character problem is complete(1,2,3,4). The major part of this theory deals with expressing missing characters in computers by giving each missing character an identification code, and solving the problem concerning input, output, and manipulation of missing characters to reduce the occurrences of missing characters in documents. The codes we developed is compatible with all the ones current used. All we need to do is add about 600 appropriate components and operating symbols in current user defined file and use character construction method to express missing characters such as structure formulation equation, component ordering and missing character serial number.

Components

When an entity is a part of a character, we call that entity is component. For example, `日' and `京'

are components of `**景**'; `**景**' and `**頁**' are components of `**顯**'; `**顯**' is a component of `**灝**'. `**口**' and `**韋**' are components of `**圍**'. Therefore, components are construction units of Chinese characters. Components are hierarchical. For example, `**顯**' is composed of `**頁**' and `**景**', which is composed of `**日**' and `**京**'. Some components are with meaning, some are not. An entity that can not be composed of other components further is called a root.

Three methods are commonly used in character construction for Chinese characters: horizontal combination(` '), vertical combination(` ') and inclusive combination(` '). For instance, ``**灝=　顯**", ``**顯=景　頁**", ``**景=日　京**", ``**圍=口　韋**". All these constructions are called structure formulation equations. In our approach, each structure formulation equation can have only one composition symbol, and it thus looks compact. For missing characters, structure formulation equation expresses the unique character shapes, distinguished from each other. Therefore, it is a good idea for missing characters to have structure formulation equation as their identifiers.

The character ``**金　本**" in sentence ``**爾時世尊食時, 著衣持金　本, 入舍衛大城乞食, 於其城中, 次第乞已, 還至本處。飯食訖, 收衣金　本, 洗足已, 敷座而坐**", quoted from ``**金剛經**", is expressed by structure formulation equation; `**金　本**' is not only a code for this missing character, but it also indicates the character structure.

We break apart more than 400,000 characters and put all these structure formulation equations in character database. Statistically speaking, 70% plus are horizontal combinations, 21% plus are vertical combinations, and 9% minus are inclusive combinations. Structure formulation equation can express character derivation, meaning that each character root generates a character family tree.

There are 2,150 components in component set acquired by forest set analysis(thereafter forest component set), from which 1,452 are characters(already existing characters in forest component set), and 698 are not characters(4), which are added in Big-5 code. However, the usage frequencies for these 698 characters are not consistent(see table 1).

Accumulated Frequency	no. of Components	Accumulated Components	user defined Characters	Accumulated User Defined Characters
0-90%	628(29.21%)	628(29.21%)	182(26.07%)	182(26.07%)
90%-99%	662(30.79%)	1290(60.00%)	193(27.65%)	375(53.72%)
99%-99.9%	345(16.05%)	1635(76.05%)	89(12.76%)	464(66.48%)
99.9%-100%	515(23.93%)	2150(100%)	234(33.52%)	698(100%)

If 99.9% of characters can be composed of structures formulation equations(0.1% of missing characters are solved by other approach for system performance consideration), then 207 non-character components can be discarded(27 of 234 non-character components are roots which should be reserved, thus 234-27=207). Of 464 non-character components, 22 has duplicate roots:

All the above characters of duplicate roots can be expressed by some easy symbols without constructing new characters. There are 464-22+27=469 characters need to be constructed in forest component set. They are listed in table 2.

Based on forest component set, the character set can expand to Big-5 code set. Accompanied by 214 origins from **康熙** Dictionary and 869 vowels and 265 **形母** from **周何**(5). The so-called Big-5 component set extends forest component set to 2,370 characters, of which 1,801 are included in Big-5 code, and 569 are non-characters(exclusive of Big-5 code). All characters in Big-5 component set are listed in Appendix A. As for 179 components needed by simplified characters are not included in Table 2 because their character shapes are not built yet.

Table 2. 469 non-character components in forest component set

Component Ordering

Let's take a look at two examples first. The character '**牖**' can not be expressed by structure formulation equation, in which only one operating symbol can be used. The left part of '**牖**' is '**片**' and the right is '**戶　甫**', which is not in Big-5 code. The structure formulation equation is ''**牖=片(戶　甫)**", and two operating symbols are used. That is a conflict of structure formulation equation we allow(Actually '**戶　甫**' is a component, but it is excluded because of low usage frequency). The other one example is the character '**奭**', composed of '**大**', '**百**', '**百**' components. That character can not be expressed by either horizontal, vertical, or inclusive combination.

Component ordering is used to express the characters such as '**牖**', which is lack of appropriate components, or '**奭**', which is lack of appropriate operating symbol. We then express those characters by the order of writing them(it is fine with the order of writing because they are identified by root equation in computer), and tag the prefix ' ', and suffix ' '. For example, ''**牖 = 片戶甫** ". We recommend the writing order of '**片**', '**戶**', '**甫**' for the convenience of correcting the missing characters in following processing.

Easy-of-use is the major concern in the design approach. That is why we get rid of 207 components

which are least frequently used(only 0.1%). Users may not be familiar with their shapes and input codes. In next section, we will propose some methods to lessen the usage difficulty for the same reason as why we use component ordering to express missing characters.

Using easy symbols and missing symbols

Not all the missing characters can be expressed by structure formulation equation. In cases of missing characters where some components does not exist or are hard to input, we create a missing symbol ` ' indicating a missing component, for example, ``竊= 穴釆 ". Only one missing symbol is permitted to exist in structure formulation equation.

We also create an easy symbol for users to input missing characters easily. The easy symbol is put as prefix of a component, for example, `oo' indicates horizontal combination of duplicate components, for example, ``兢= 克"; `8' indicates vertical combination of duplicate component, for example, ``戔=8 戈"; `ooo' indicates horizontal combination of triple components, for example, `ooo 去', No. 3171 in Chinese Dictionary); ` ' indicates vertical combination of triple components, for example, `戶', No. 12029 in Chinese Dictionary); `oo' indicates triangle position of triple components, for example, ``轟=oo 車''; `oooo' indicates horizontal combination of quadruple components; `8' indicates vertical combination of quadruple components; `oo' indicates rectangle position of quadruple components, for example ``燚=oo 火".

Missing symbol and easy symbol can be used in structure formulation equation to extend its scope, transforming character construction from component ordering to structure formulation equation. For example, ``瞿=oo 目 隹", ``俎=8 人 且", ``桑=oo 又 木", ``啜=口 oo 又". Easy symbol is used in component ordering to make structure formulation equation easy. For example, ``歠= oo 又酉欠 ". Easy symbol also makes structure formulation equation easy by discarding horizontal, vertical, inclusive symbols and prefix and suffix tags, for example, `oo 魚' (No. 47603 in Chinese Dictionary).

Missing character serial number

There are about 2,000 characters in Chinese Dictionary which are something like pictures, thus being unable to be constructed with components. Instead, they must be expressed by missing character serial number. The format for missing character serial number is similar to that for component ordering, i.e. using prefix and suffix tags and numbering for component. For example, `5' is the fifth missing character that can not be expressed by all rules described.

Other expressing rules

We are designing some other expressing rules for variants and derivation(6). Basically, variants do not have a corresponding code. We have database for variants management, and font library for variant shapes.

Manipulation and Analysis for Missing Character in Chinese Full-Text Database

For solving missing characters at Academia Sinica, we acquire information about missing characters from Computing Centre. Through collecting, processing, and analyzing those data, we realize the situation and start to design and implement a system for solving missing characters(thereafter ``the system"). Next, we will explain the result of our study.

Missing characters at Academia Sinica

Missing characters managed by Computing Centre are divided into two categories: those built in Big-5 code user defined area(4,553 characters), and those not built yet(5,174 characters), totally 9,727 characters up to March, 1998. We can build up 5,809 characters in Big-5 code user defined area and 4,553 character are already built and 1,256 character are still left blank and distributed as follows:

1. 702 characters in FAB5-FEFE section
2. 480 characters in 9DF6-A0FE section
3. others dispersed separately

We will use the first section(FAB5-FEFE) to add non-character components with the consideration of not disrupting the existing user defined area and with intent of smoothly transforming from old system to new one.

Analysis of 4,553 characters in user defined area}

1. Excluding 13 duplicate characters in Big-5 code, 6 variants and 9 symbols, the total number of missing characters is 4,525
2. Of 4,525 missing characters, 3,903 ones(86.25\%) can be expressed by structure formulation equation, 515 characters(11.38\%) by component ordering, and 107 ones(2.37\%) by adding new roots.
3. According to on-line database totally summed up 138,000,000 characters, the number of usage of 4,525 missing characters is 517,891, of which 411,698(79.50\%) for 3,903 missing characters, 89,638(17.31\%) for 515 missing characters, and 16,555 for 107

missing characters.

Analysis of 5,174 missing characters not built yet

1. Excluding 7 duplicate characters in Big-5 code, 10 self-duplicate characters, 14 blank characters, and 199 symbols, the total number of missing characters is 4,944.
2. Of 4,944 missing characters, 3,760(76.05\%) can be expressed by structure formulation equation, 864(17.48\%) characters by component ordering, and 320 ones(6.47\%) by new roots.
3. The number of usage of 4,944 missing characters is 16,598, of which 13,375(80.58\%) for 3,760 missing characters, 2,409(14.51\%) for 864 missing characters, and 814(4.91\%) for 320 missing characters.

Missing characters manipulated by our system

According the statistics depicted in previous section, the number of missing characters at Academia Sinica is 9,727. But if excluding duplicates, variants, ancient characters, and symbols, we have 9,469 missing characters, and its number of usage is 517,891+16,598=534,489. For the database containing 138,500,000 characters, the probability of missing characters is 0.038% with respect of Big-5 code. If we build up 4,525 missing characters in user defined area, then there are still 16,598 characters which can not be expressed(0.0012%). If we use the suggested approach, then 517,120 occurrence of missing characters can be solved, approximately the same percentage as old system does(0.00125%). However, if we replace the existing user defined character file with 427(107+320) missing characters that can not be handled directly, then the missing character problem can be solved completely.

Solution and Management for Missing Characters

The production procedure for Chinese Full-Text Database is taken by 5 stages, i.e. data-entry, correction, missing character management, mark-ups, and database construction. These stages are processed step-by-step. But missing character management can happen at either data-entry or correction stages. Next, we describe the missing character management currently used, then explain the solution for missing characters, followed by the operating procedure adopted.

Current missing character management

In current data-entry process, we use `。' symbol to denote a missing character. We then list one by one a table for missing characters indicating where they occurs and their shapes in final correction

process.

Missing character management is more complicated. First, we filter the duplicate occurrence of missing characters, mark it as new character, and count the number of occurrence of these new characters. We worked this out manually and costly. Although the efficiency is improved by means of auxiliary tools, the approach we suggests is much better because no lookup of new characters is needed.

When new characters are inspected and marked, we then create them. Limited by 5,809 user defined character area in Big-5 code, it is impossible to create nearly 10,000 new characters. The missing characters that are most frequently used are created first.

When creation of new characters is complete, we engage in backward data entry of missing characters. According to the table of missing characters, we can fill in its new character where it's found before. Some errors may happen in the backward data-entry process. For example, missing characters are still not corrected or they are corrected with wrong characters or using incorrect command in editing, rendering data damage and lost.

Our missing character management

In reference article(2), a data-entry system is discussed. It comprises of character database, network, compiler with character construction ability, and Chinese features, and it is expected to solve the missing characters by providing missing character shapes in time. But the techniques embedded in that system are complicated and hard to grasp in a short of period. However, the missing character problem must be solved immediately. Before transforming from the current system to ideal system, we must take into consideration the intermediate process concerning personnel, technology, equipment difficulties. We will explain this next.

The missing character management currently used is more efficient in that there are only combination components, operating symbols and new characters, which can not be expressed by structure formulation equation. In data entry stage, characters not in standard character set are expressed either by structure formulation equation or directly by characters in user defined area. Unless characters can not be expressed by structure formulation equation, or they can not be found in user defined area, there's chance for us to express them with missing symbols. If no missing symbols are used, the backward data-entry process is not needed, decreasing the key-in errors and promoting the efficiency. If those characters do appear in documents, we can look up some character database for their missing character serial number, then put it in data file. If those rare characters are not found in character database, we create them in user defined file and change missing character serial

number to internal code.

In the correction stage, we check if the structure formulation equation for missing character is correct or not. Furthermore, we can display the missing characters by mapping structure formulation equation to character font using font file, which is generated with the help of character database and relevant programs. It is feasible because the character database can connect to large character library such as Kanji Base. We use font file to store shapes of missing characters expressed by structure formulation equation. By means of structure formulation equation-character font corresponding relationship, we solve the problem of duplicate codes between missing characters in font file and standard character set, and we can display the missing characters correctly.

The programs for missing character processing are as follows:

1. combination collection module is used to scan the data file, search structure formulation equation for missing characters, and report the location it appears at the very first time.
2. character database inquiry module is used to check the structure formulation equation found by combination collection module. According to structure formulation equation, together with easy symbols introduced, there are two or more structure formulation equations for a missing character, but these equations can be concluded with the same root ordering. We then use root ordering to find information about missing character in character database. If the missing character is not in standard character set, we write down its serial number, root ordering, structure formulation equation and character address; otherwise, we get its internal code. If root ordering does not exist, the correction inspector would decide whether it is a new character or its structure formulation equation is wrong. If it is a new character, we put it in character database, and run the character database inquiry module again.
3. Font file management module is used to process the results from previous module. If the file does not exist, we get the font for missing characters, put it in font file, and create structure formulation equation-character corresponding table. The fields in corresponding table include serial number, structure formulation equation, root ordering and font information(font filename/internal code). The internal code is assigned either automatically or manually. If the font file and corresponding table do exist, some programs are used to insert that missing character into corresponding table and font file. When lots of missing characters are found. there are more font files, but only one corresponding table is kept.
4. Structure formulation equation-character convert module is used to map the structure formulation equation via corresponding table into character or standard character internal code. Then it is used in correction stage with the help of font file. In the correction stage,

we must check and correct, if necessary, the structure formulation equation and change the internal code into structure formulation equation. Finally, we use programs to check if these structure formulation equations are correct or not.

All these modules are not complicated and can be implemented shortly. Once all the character font are built in character database, the correction steps can be summarized as follows:

1. Run programs with single command to generate or update font file, and produce data file in which character font has been transformed.
2. Set up font file, if it is changed.
3. Correction
4. Check structure formulation equation with single command

Solution for Missing Characters and Full-Text Database Tools

Solution used for missing characters involves the internal structure and program tools used in Chinese Full-Text Database. The problem we encounter first is whether structure formulation equation is put in database directly or it needs to be transformed? Considering that the structure formulation equation is not unique, its length is variant, and its format is somewhat complicated, we decide to define a format, called combined code, with constant length for the searching performance. The syntax is:
<combination code tag><character database serial number>

Combination code tag is one-byte long and its value is distinguished from values in both Chinese code and ASCII code. We can define its value to 255 in the Big-5 code context. Character database serial number is 3-byte long, and we use printable codes in ASCII code for their values. There are 94 printable characters in ASCII code, thus we have 830,584(94*94*94) different values for character database serial number. Now that character database is centrally managed, and each character is well-defined, combination code has the data-sharing potential, although it is used interchangeably.

The program tools for Chinese Full-Text Database are database built-up and database indexing subsystems as core. These tools have the abilities of processing structure formulation equation and combination codes. In addition, structure formulation equation solves the problem of variants. The difficulties are increase by indexing when searching short sentences with different variants. For example, `菸葉' and `火　因葉' should be included in the results when searching `煙葉'. However, distinguishing variants is not necessarily done by the same way. We collect characters commonly used as the same meaning as a group called common character group used for solving some

problems. Database built-up subsystem is used to read in marked up data files, map structure formulation equation in corresponding table into combination or internal code, generate full-text database, and produce index. Combination code influences the index built-up. The index is constructed by character reverse structure, recording the addresses in database for Chinese and foreign characters. Chinese characters and foreign characters are manipulated separately because of the attributes of length of internal code and number of characters. Combination code is of constant length, but of unknown number of character. As for variants, it is better to locate adjacent area for common character group with intent of fast access of that group.

Database indexing subsystem provides users with data resulting from indexing conditions. Structure formulation equation influences the aspects of input, index and output. Structure formulation equation, when being typed with command in input, is transformed to combination or internal code. Most users are not familiar with the components, operating symbols in structure formulation equation. Thus we need input user interface, in which components and operating symbols are listed, and users can construct structure formulation equation from the interface. We also need the relevant mechanism for selecting missing character serial number in case of void structure formulation equation.

Two methods are introduced in database indexing subsystem for searching: by index or by characters. The former is explained in previous paragraph. For the latter method, a short sentence can be extended as a variant group and be searched. For example, `**煙葉**' is extended as `**煙葉**', `**菸葉**' and '**火　因葉**'.

There are characters which can be used in most situation interchangeably, for example, `**饑**', and `**飢**'; Some are used interchangeably in special context, e.g. `**煙**' and `**菸**' can not be used interchangeably if not in `**煙草**' context. Generally speaking, it is effective and without side effect when searching short sentence consisting of two or more characters. For example, when searching for `**煙囪**', the searching engine would search for `**火　因囪**', and `**菸囪**', too. `**火　因囪**' will be found, but '**菸囪**' us not because it does not exist. In current stage, users are persuaded to adjust common equivalent group for special request.

To display missing characters is to map the structure formulation equation into appropriate character shapes. That is fine for reading documents. However, more attentions should be paid to the missing characters for cut-and-past processing. For example, if the cut content is about to be pasted in the searching field, the index program is able to transform character shapes back to structure formulation equation by combination font corresponding table. Further, what happened if user get the missing character font file and would like to proceed indexing search or statistics report? Except displaying missing characters for reading only, users are given data files for structure formulation equation, combination corresponding table and relevant programs for their usage of missing

characters.

Conclusion

This article proposes an approach to the missing character problem at Academia Sinica and this approach is feasible and provides a comprehensive solution. The upgrade of existing system at Academia Sinica is now in progress, and the first version of the user interface prototype for character database is released. It will be demonstrated in an appropriate time, if possible.

The study keeps going. The system can be extended as character auxiliary tool for Chinese research by adding linguistic data and procedure. In application, the first priority is to develop a mechanism for Chinese document sharing with intent to sharing and manipulation of missing character documents.

You are mostly welcome to use this system. People who are interested this system can either contact us or visit our web site at http://www.sinica.edu.tw/~cdp