Progress Report

Project: Simplify Chinese Even Further

Student Name: Wei Li
Abstract

The main aim of this project “Simplify Chinese Even Further” is to work out a set of simplification principles and word segmentation algorithms for written Chinese language, and then implement the algorithms to create a software that accepts ordinary Chinese texts and transform them into segmented word combinations of simplified characters based on the prebuilt simplification table.

This report firstly detailed explains the related background knowledge that will be used in this project, containing the structure of Chinese characters, the property of Chinese words, the classification and detection of ambiguity, the automatic segmentation techniques, and the basic structure of Chinese segmentation system. Because of the inheritance characteristics of Chinese words [1], the minimum segmentation algorithm [2] cannot be used in word segmentation of Chinese. FMM (Forward Maximum Matching Method) [3] and RMM (Reverse Maximum Matching Method) [4] are the remaining two practical methods that can be used. Besides, A language ambiguity can be classified into two categories, one is real-ambiguity and the other is pseudo-ambiguity, according to whether it can be determined by human. In this report, the ambiguity discussed is pseudo-ambiguity, which can be determined by human and further reduced via the improvement of algorithm. There is another classification method for ambiguity based on the cause of it, containing segmenting ambiguity [5] and combinational ambiguity [6]. There are two methods of ambiguity detection introduced in this report, one of them is word-by-word scanning, which is the prototype of the ambiguity detection algorithm that used in this project.

After the demonstration of background, the research methods is then shown. The four major tasks of this project are study background knowledge that will be used in the simplify process, establish a database for character pairs and word-combination (milestone 1), implement the Java program that can transform the ordinary Chinese characters to the simplified ones with spaces (milestone 2), together with statistics, survey and evaluation (milestone 3). The detailed project plan in Gantt chart form[7] as well as the evaluation plan are also delivered in this part.

After the description of the research methods and plans, the work that I have done in the past three months is shown in the following chapter. I pushed this project forwards strictly according to the Gantt chart. To be mentioned, I have already reached milestone 1 via completing all the sub-tasks in Task 1 and Task 2. Besides, I have also finished the first three sub-tasks of Task 3 and I’m currently implementing the word segmentation algorithm to reach the second milestone.

key words: Chinese, character, simplify, segmentation, ambiguity
# Contents

1 Introduction 4  
1.1 Objectives and content 4  
1.2 Scope of research 5  
1.3 Structure of report 5  

2 Background 6  
2.1 Overview of Chinese language 6  
2.1.1 Structural unit of Chinese characters 6  
2.1.2 Building methods of Chinese characters 6  
2.1.3 The property of Chinese words 7  
2.1.4 Introduction and classification of ambiguity 8  
2.2 Words segmentation techniques 9  
2.2.1 Automatic segmentation methods 9  
2.2.2 Automatic segmentation model (ASM) 9  
2.2.3 Typical segmentation methods based on ASM 10  
2.2.4 Basic structure of Chinese word segmentation system 11  
2.2.5 Detection of ambiguity 11  

3 Research methods 12  
3.1 Tasks involved 12  
3.2 Project plan 12  
3.3 Evaluation plan 14  

4 Progresses have been made 14  
4.1 Task 2: Establishment of the database 15  
4.1.1 Sub-task 1: Design the data structure 15  
4.1.2 Sub-task 2: Determine the simplify principles 16  
4.1.3 Sub-task 3: Build a table of character simplify pairs 16  
4.1.4 Sub-task 4: Collect data of word-combination for space 16  
4.2 Task 3: Implementation of the Java program 17  
4.2.1 Sub-task 1: Connection with database 17  
4.2.2 Sub-task 2: Realize the characters substitution function 17  
4.2.3 Sub-task 3: Determine word segmentation algorithm 19  

5 References 21
1 Introduction

1.1 Objectives and content

Unlike any other western languages, Chinese language turns out to be much more complex due to the fact that the majority of Chinese characters have numerous strokes as well as complicated structures containing a “semantic radical” (usually is not a character) and a “phonetic component” (usually is a character itself) [8]. Therefore, throughout history, the simplification of Chinese language is a considerable measure being mentioned over and over again, to make the written language itself learnt and used by more Chinese people. In 1956, Chinese government launched the simplification campaign for Chinese characters [9], in which the strokes of the most frequently used 2,000 characters were reduced. However, the composite structures still remain and widely appear in modern Chinese characters (or “characters” for short). This project “Simplifying Chinese Even Further” aims at designing a set of more radical simplified Chinese characters (or “simplified characters” for short) and setting up a new Chinese language writing standard by introducing the spaces between words.

The setting up of a series of simplified characters is based on the hypothesis of discarding the semantic radical of a group of characters [10], which have the same pronunciation with their phonetic component. It seems to be a practical method in simplifying characters mainly due to the following three reasons. Firstly, it can lower the complexity of characters by discarding strokes. Secondly, it can reduce the amount of characters by using one “phonetic component” to represent a series of more complex characters based on it. And more importantly, it has controllable impact on understanding because of the same or similar pronunciations between original and simplified characters.

However, this kind of radical simplification of characters may aggravate language ambiguity, which is the wide-spread phenomenon in written Chinese language due to the massive possible combinations of characters. The most typical case of language ambiguity in written Chinese happens when a single character can combine with either previous or later character in a sentence to become a word. These two possible ways of word segmentation lead to totally different meaning of this character, and thus have impact on the understanding of the whole sentence. What’s more, the substitution of a series of characters by using their common “phonetic component” is very likely to increase the possibility of such situation, and further causes more ambiguity in Chinese written language. By contrast, in spoken Chinese, the language ambiguity caused by different possible word segmentation ways can be reduced to a large extend, because of the automatic separation of the words comes with the variation of people’s talking paces. Inspired by this point, introducing the spaces between words (characters’ combination) turns out to be a feasible remedy to deal with the increasing ambiguity. Therefore, the new set of simplified characters together with the new writing standard (with spaces) can make
the written Chinese language enjoy a similar degree of lack of ambiguity with spoken Chinese language.

1.2 Scope of research

In this project, first of all, I plan to design a set of simplify principles, according to which most of the characters can be simplified. There are about 7000 characters still used in modern Chinese, which can be classified into three catalogues, containing 2500 frequently used characters, 1000 less frequently used characters and 3500 rarely used characters [11]. I plan to use less amount of simplified characters with less strokes to represent those considerable huge amount of ordinary Chinese characters [12][13]. Comparing to the first simplify campaign conducted by the Chinese government in 1956, my simplification process should be wider (cover majority of characters in Chinese written material), deeper (scientific and systematic principles) and more radical (discarding more strokes by deleting the “semantic radical”). Meanwhile, the recognizability of this new set of simplified characters should be ensured among Chinese people. Secondly, a complete set of words should be established as a reference table when comes to the segmentation of words, together with an algorithm that tells how to use the words table to separate the words in a sentence. After that, the software will be developed, which accepts articles written in ordinary characters and map them into the simplified ones referring to the pre-built simplified characters matching table. Moreover, it can also introduce spaces between words based on the words table and segmentation algorithm. The output of the software should be a set of separated words of simplified characters based on the input article. At the last stage, the recognizability of the simplified characters with spaces between words will be tested among Chinese people via questionnaires. Furthermore, the intensity of simplification (measured by the ratio of the total amount of characters considered and the amount of their distinct simplified characters, together with the average strokes’ reduce of the characters before and after simplification process), the coverage rate of the simplified characters over the general Chinese written materials (the percentage of the amount of characters that could be simplified according to the matching table out of the total characters’ count in an article) and other important evaluation indexes should also be analyzed during the statistics process.

In the process of simplification, the five tones (the first, second, third, fourth and neutral tone) for 400 Chinese language syllables [14] will not be distinguished, which means a character that has the same syllable and different tone with its “phonetic component” will be regarded as pronunciation unchanged with its phonetic partial. This is because the tone in Chinese language does not play an important part in basic understanding [15]. The tones of mandarin spoken by Chinese people vary from area to area, but people can still understand each other mostly. For example, people from Shaanxi province in China prefer to transform the first tone to the third tone, the third tone to the fourth tone, the fourth tone to the first tone and the second tone remain same when speaking mandarin due to the influence of their own dialect.

1.3 Structure of report
This report introduces the aims and scopes of my final year project “Simplify Chinese Even Further”, and systematically presents the background materials I have learnt related to this project topic, concludes the methods that used in conducting the project, and most importantly, record the related work that I have done in the past three months. There are four chapters in this report. The first chapter is the introduction of project and this report. The second chapter is the background, in which the structure of Chinese characters, the characteristics of Chinese words, and automatic segmentation algorithms are well discussed. The third chapter is about research methods for this project. In this part, the tasks involves, the project plan and evaluation plan are shown. The fourth chapter well presents the progress that has been made up till now.

2 Background

2.1 Overview of Chinese language

2.1.1 Structural unit of Chinese characters

There are two structural units for characters, one is stroke and the other is component [16]. Stroke is the least unit to comprise a character, and all of the characters are made up of strokes. There are five basic strokes in Chinese and twenty five derived strokes based on the basic ones [17]. By contrast, component is a higher level unit based on strokes, which can be further used to form componental characters. According to whether the it is a character or not, the component can be classified into the group of characteristic component or uncharacteristic component. Besides, according to whether it can be cut into smaller components, one component can also be classified into the group of single or composite component. However, the uncomposable characters have only one component, whereas all the composable characters have more than one components.

2.1.2 Building methods of Chinese characters

In general, the building methods of Chinese characters indicate the formation of the characters. According to different building methods, Chinese characters mainly fall into four categories, containing pictograph, self-explanatory, ideograph, and phonogram [18].

Pictograph [19] describes the shape of objects, either their whole structure or partial characteristics. In ancient Chinese language, pictograph makes up a relative large percentage among the whole language system. In the modern Chinese language, the majority of the prior pictographic characters have been transformed into the other three categories of characters. Nevertheless, pictograph is still the basic building method for Chinese language, for many ideographic and phonogram
characters are made up of pictographic characters. However, they are very limited when express the complex or abstract objects, and further they cannot satisfied the need of expression.

Self-explanatory [20] is the character building method that takes use of the pictographic characters as its main part, and adds prompt signs to them, in order to emphasize certain parts of the pictographic characters, or express meaningful circumstances. This kind of building method is also very restrained and the amount of self-explanatory characters is even fewer than the counterpart of pictographic characters in modern Chinese language.

Ideograph [21] uses the building method of combining several components together to make a new character, the meaning of which is the combinational meaning of those components used. Most of the ideographic characters in modern Chinese language are evolved from the ancient ideographic characters.

Phonogram [22] also uses the building method of combining several components together to make a new character. Compared with the ideograph, the meanings of phonogram characters are indicated by their “semantic radicals”, and the pronunciations of them are derived from their “phonetic components” (have either same or similar pronunciations). A series of phonogram characters with different syllables and related meanings can be created by putting a single “semantic radical” together with several different “phonetic components”. Meanwhile, a set of characters with similar syllables and unrelated meanings can also be generated by putting one “phonetic component” together with several different “semantic radicals” respectively. Hence, this characters building method is the most productive one among these four methods in characters building. Phonogram is widely used in modern Chinese language, which is focused on in this project. And it is also an important symbol that indicates the development of the characters’ structure is on its way from ideographic to phonetic.

2.1.3 The property of Chinese words

Word is the smallest language unit that can be separated to express the same meaning with what it serves in a sentence. Chinese is a typical “object-oriented” language with the obvious characteristics of encapsulation, inheritance, and polymorphism [23].

In software programming, encapsulation means the separation of interface and implementation to protect the inner function. Similarly to Chinese, each character has appearance interface (same square appearance), definition interface (mostly one main definition for one character), and syllable interface (mostly one pronunciation for one character).

Polymorphism can be easily find in Chinese because there are many conceptual words in Chinese that can be inherited. Besides, Pinyin (written form of character’s syllable) also shows another presentation of Chinese language.
Most importantly, inheritance is a markable characteristic of Chinese language that cannot be ignored. The majority of Chinese words (comprised by more than one characters) inherit the meaning of the characters it contains. For example, in Chinese, “洗” means “wash”, “发” means “hair”, and “膏” means “cream”, when we put the first two characters together to generate a word, it means “hair-washing”. However, when we put all these three characters together, we can get another new word “洗发膏”, which means “shampoo”. This new word inherits the meaning of all the three characters it contains and become a creamy object that can be used to wash hair. For another example, when we put two characters indicate “electricity”(电) and “brain”(脑) together, they become a single word means “computer”(电脑), which shows the computer works like a electronic brain. When we add a new character indicates “bag” after the former two characters, these three characters appear together and gives a new word “电脑包” (indicates “computer bag”). It should be noticed that most of the characters here are words themselves that can be used alone, however, when add another character after them, they become a new word, and when add one more, they combine and give another new word. Such circumstance is widely existing in Chinese words generation. It shows that when choose word segmentation algorithms, only those focus on longer characters combination and then shorter ones can be applied in Chinese.

2.1.4 Introduction and classification of ambiguity

Ambiguity can be found in Chinese, results from more than one possible ways to conduct the word segmentation for a certain sentence. According to whether the real meaning can be judged by human, a ambiguous sentence fall into either real-ambiguity or pseudo-ambiguity category [24]. In this project, we only focus on the pseudo-ambiguity category, which can be largely avoided through the improvement of segmentation algorithms. This also means that ambiguity circumstances mentioned in the following sections are those can be judged by human.

According to the different cause, ambiguity can be divided into two types, containing segmenting ambiguity and combinational ambiguity [25]. Segmenting ambiguity [26] occurs when a character in the sentence can be either combined with its previous character or latter character. In such cases, it is hard to determine where to segment, specially for the machine. For an example in the phrase “ABC”, suppose both “AB” and “BC” are possible character combinations (words). It is hard to say whether should we put the segmental symbol between “A” and “B”, or “B” and “C”. Combinational ambiguity [27] occurs when two adjoining characters in a sentence can combined to give a word, and each of them can be treated as a word itself. In such cases, whether should we separation these two characters will be a question. For an example in the phrase “ABCD”, suppose “AB”, “CD”, “C”, and “D” are all possible candidates (words). We should firstly insert a segmental symbol between “B” and “C” to separate “AB” and “CD”, but whether should we insert another segmental symbol between “C” and “D” become a question, either decision will give a different expression of the sentence's meaning from each other, and thus cause combinational ambiguity.
2.2 Words segmentation techniques

2.2.1 Automatic segmentation methods

With the development of the technology, machines are always expected to conduct the segmentation of words automatically, which is the base of many advanced applications, such as intelligent search, interactive software and so on. There are mainly three categories of methods to realize this kind of automatic segmentation, containing dictionary driven matching, rules driven, and statistics driven methods [28].

Dictionary driven matching method [29] indicates the segmentation of the given sentences based on the results of matching each possible characters combination (word) in the pre-built words database, which is treated as dictionary. Each time it successfully match a word in the database, the system will remove this word from the undetermined sentence to be segmented and move on, otherwise, it will switch one character and try again.

Rules driven method [30] can be regarded as a supplement to dictionary driven matching method. It usually involves several tables with different types of characters, and then determines a set of regular expressions according to those tables, for the system can know how to segment words based on the regular expressions. For example, suppose table 1 records the numbers in all forms, and table 2 records all the Chinese classifiers. The system will be told one or more characters from table 1 followed by a character from table 2 should be regarded as a word, via a regular expression.

Statistics driven method [31] uses the frequencies of each possible Chinese word and match them into the given sentence from the word with highest frequency to the lowest, rather than match the words into the table. This method is limited because it requires a pre-built large statistics table and also the frequencies from a reliable recourse, which make it not very practical in operation.

2.2.2 Automatic segmentation model (ASM)

Automatic segmentation model provides a basic structure for the segmentation procedure based on dictionary driven matching method. There are three indexes in this model, which can be denoted as ASM (d,a,m). The meanings of “d”, “a”, “m” are shown as below separately [32].

“d” - direction of the matching process, +1 means from the beginning to end (forward), and -1 means from the end to the beginning (reverse);
“a” - the amount of characters added/ subtracted when matching failed, +1 means add one more character after matching failed, -1 means subtract one character from the current matching combination after matching failed;
“m” - indicates whether maximum or minimum matching method is used in the system, +1 means maximum matching, and -1 means minimum matching. For
example, if both “ABC” and “AB” are possible characters combination candidates, the system will choose “AB/C” as segmentation result if minimum matching method applied, otherwise if maximum matching method applied, “ABC” will be the final result, for the word “ABC” has the longer size than “AB”.

2.2.3 Typical segmentation methods based on ASM

According to the different value we choose for these three indexes “d”, “a”, and “m” in ASM(d,a,m), there are mainly six methods. However, as the object-oriented characteristic, especially the inheritance property of the Chinese language, the maximum matching is the only possible choice for Chinese words segmentation (“m” can only be set as +1) [33]. In maximum matching, “a” is better to be set as -1, in order to reach a higher efficiency, for the reason that we want to find the possible word with longest size. If we begin from the largest size combination and subtract one character each time matching failed, we can stop and move on to the next stage whenever find out a word in the table, however, if we started from the smallest size word and added one character after matching failed, we should continue to roundtrip all the possible sizes of words even we find out a matching, because a longer word might exist.

Hence, when comes to the Chinese language segmentation, the only index we can change is the direction of searching “d”. It can be set either +1 or -1, and thus generates FMM (Forward Maximum Matching Method) and RMM (Reverse Maximum Matching Method), respectively.

FMM (Forward Maximum Matching Method) [34] is a method based on ASM which can be represented as ASM(+, -, +). Suppose “D” represents the dictionary, “max” is the length of the longest word in “D”, and the input sentence is “C1C2C3…Cn”. To begin, the string “C1C2…Cmax” will be selected from the input sentence, and checked if it is a word in “D”, if so, the string will be labeled as a word and the pointer switch length of “max”. The next string to be checked is “Cmax+1Cmax+2…C2max”. Otherwise if the string “C1C2…Cmax” is not a word in “D”, it will subtract one character and continue to check whether “C1C2…Cmax-1” is a word in “D”. If all of the possible combinations started from “C1” is not a word in “D”, the next string to be checked will be “C2C3…Cmax+1”, in same rules. In the previous statistics analyze, the error rate of FMM is about 1/169 [34].

RMM (Reverse Maximum Matching Method) [35] is a method based on ASM which can be represented as ASM(-, -, +). Suppose “D” represents the dictionary, “max” is the length of the longest word in “D”, and the input sentence is “C1C2C3…Cn”. To begin, the string “Cn-max+1…Cn-1Cn” will be selected from the input sentence, and checked if it is a word in “D”, if so, the string will be labeled as a word and the pointer switch length of “max”. The next string to be checked is “Cn-2max+1…Cn-max-1Cn-max”. Otherwise if the string “Cn-max+1…Cn-1Cn” is not a word in “D”, it will subtract one character and continue to check whether “Cn-max+2…Cn-1Cn” is a word in “D”. If all of the possible combinations ended with “Cn” is not a word in “D”, the next string to be checked will be “Cn-max…Cn-2Cn-1”, in same rules. In the previous
statistics analyze, the error rate of RMM is about 1/245 [35], which is slightly better than that of FMM.

### 2.2.4 Basic structure of Chinese word segmentation system

There are mainly six stages in Chinese word segmentation system, containing input Chinese text, segmentation with basic method, ambiguity detection, ambiguity segmentation, unregistered word recognition and output of result. A dictionary and a rule base should be built to be referenced in the segmentation process. The basic structure [36] of Chinese word segmentation system is shown as below in Figure 1.

![Figure 1. The basic structure of Chinese word segmentation system](image)

Both FMM and RMM can be used in the basic segmentation stage, however, either of them may contain ambiguity. Hence, a set of algorithms to detect and deal with the ambiguity is needed. Besides, it is impossible to include all the words in the dictionary, thus the method to recognize the unregistered word in the input text should be well considered [37]. After the check of unregistered word, the system will be able to present the final result of segmentation.

### 2.2.5 Detection of ambiguity

The detection and handling of ambiguity are the most important parts of a word segmentation system. There are mainly two typical ways to detect the ambiguity, containing the bi-direction scanning method and word-by-word scanning method.

Bi-direction scanning [38] can find out the most of the segmenting ambiguity in sentences by comparing the segmentation results given by FMM and RMM. If the system got two different results by scanning in two directions, it means there are at least two possible segmentation ways existing for this single sample sentence. For example, if both “AB” and “BC” are possible word candidates, the given string “ABC” will be defined as a segmenting ambiguity string. If we scan this string in both directions (forward and reverse), we will get “AB/C” and “A/BC” respectively.
When compare these two results given by FMM and RMM, the system will find the difference in segmentation and report the ambiguity in this string.

The word-by-word scanning method [39] is another well-known method in ambiguity detection. The basic procedures of this method is shown as below.

1. Select the first string with the size of longest word in dictionary from the beginning of the give sentence;
2. Search this string in the dictionary;
3. If failed to match the string with items in the dictionary, remove the last character in the string and switch to (2) (similar to FMM);
4. If succeed, then segment a word “CiCi+1…Cj”, and compare it with the immediately prior word segmented “CpCp+1…Cq”;
5. If j<=q, then combinational ambiguity occurs here, labeled ambiguity and switch to (8);
6. If j>q and i<=q, then segmenting ambiguity occurs here, labeled ambiguity and switch to (8);
7. If j>q and i>q, there is no ambiguity here, labeled the word and switch to (8);
8. Switch one character forward and operate the new string from step(1).

3 Research methods

3.1 Tasks involved

This project involves four major tasks followed by several sub-tasks. The first major task is to study the background knowledge that will be used in the simplification process, such as the structure of Chinese characters, word segmentation algorithms, and Java programming. The second task is to establish a database which can be connected by java program, to records the character pairs and word-combination. The table-building for simplified character pairs is based on a set of practical and reasonable principles for simplification. It is one of the most important parts of the whole project, and is also the basic of the following tasks. The third task is to implement the Java program that can transform the ordinary Chinese characters to the simplified ones and also introduce the spaces between words. The difficulty in this task is the algorithm for detecting and handling the ambiguity. It is the most important milestone of this project because the most practical outcome of the project will be produced in this stage. The last task is to do some statistics, survey and evaluation related work. A survey will be delivered via questionnaires to check the recognizability of the simplified characters among Chinese people. Moreover, the statistic analysis and data presentation will also be conducted in this stage. Besides these four tasks, report writing is also an important part, which should be set out from halfway of the whole process.

3.2 Project plan

The plan for this project involves in determining the sub-tasks for each major task, and also allocating time periods for each sub-task. The detailed plan in the form of
A gantt chart is shown as below in Figure 2 (the latter half of May is left blank for the final exams).

<table>
<thead>
<tr>
<th>Task 1: Study background knowledge (theories and techniques) that will be used in the simplify process</th>
</tr>
</thead>
<tbody>
<tr>
<td>Study the structure of Chinese characters</td>
</tr>
<tr>
<td>Study word segmentation algorithms</td>
</tr>
<tr>
<td>Study Java basic skills (input and output)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task 2: Establish a database for character pairs and word-combination (Milestone 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design the data structure</td>
</tr>
<tr>
<td>Determine the simplify principles</td>
</tr>
<tr>
<td>Build a table of character simplify pairs</td>
</tr>
<tr>
<td>Collect data of word-combination for space</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task 3: Implementation of the Java program that can transform the ordinary Chinese characters to the simplified ones and also introduce the spaces between words (Milestone 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection with database</td>
</tr>
<tr>
<td>Realize the characters substitution function</td>
</tr>
<tr>
<td>Determine word segmentation algorithm</td>
</tr>
<tr>
<td>Implementation the algorithm</td>
</tr>
<tr>
<td>Testing</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task 4: Statistics, Survey and Evaluation (Milestone 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessing corpus of Chinese material</td>
</tr>
<tr>
<td>Design the questionnaire</td>
</tr>
<tr>
<td>Distribution and collect the questionnaires</td>
</tr>
<tr>
<td>Statistical analysis and data presentation</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task 5: Dissertation Writing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Work out the basic framework</td>
</tr>
<tr>
<td>Complete the draft</td>
</tr>
<tr>
<td>Modification of the draft</td>
</tr>
<tr>
<td>Proof-reading and generating final version</td>
</tr>
</tbody>
</table>

Figure 2. Gantt chart for the project “Simplify Chinese Even Further”
3.3 Evaluation plan

There should be certain criteria according to which one project can be evaluated. Without evaluation, we cannot tell whether the project result is superior to the previous ones or not, besides making adjustments or improvements [40]. Typically in this project, we can know how effective and efficient the substitution and segmentation algorithms are through evaluation process. This project can mainly be evaluated from two aspects, one is intuitive and the other is objective.

From the intuitive point of view, this project can be evaluated according to:
- The effectiveness of simplify principles;
- The effectiveness of segmentation algorithm;
- The effectiveness of ambiguity detecting and handling algorithms;
- The design of the database for character pairs;
- The usability and elegance of the software;
- The quality of final dissertation.

From the objective aspect, this project can be evaluated according to the following indexes:
- The scope of Chinese characters concerned (measured by the amount of ordinary characters considered during the simplified process);
- The coverage rate of the simplified characters over the general Chinese written materials (the percentage of the amount of characters that could be simplified according to the matching table out of the total characters’ count in an article);
- The intensity of simplification (measured by the rate of characters’ amount together with the strokes’ number before and after simplify process);
- The recognizability of simplified characters among Chinese people (acquired by the analysis of questionnaires);
- The completeness of database for character combinations (words);
- The time used in substituting and segmenting every 100 characters;
- The precision of word segmentation (Precision = amount of correctly segmented words / total amount of words segmented) [41];
- The recall of word segmentation (Recall = amount of correctly segmented words / total amount of segmented word in the answer) [42].

4 Progresses have been made

In the past three months, I was pushing this project forwards strictly according to the Gantt chart. To start with, I studied some background knowledge that will be used in the simplification process, containing the overview of Chinese language, word segmentation algorithms for Chinese, basic Java programming skills, and thus have completed Task 1. Besides, I have finished all the sub-tasks of Task 2 and achieved the first milestone. Moreover, the first three sub-tasks of Task 3 have also been accomplished. And I’m currently implementing the word segmentation algorithm to reach the second milestone.
The achievements towards Task 1 have been already shown in previous chapter, hence, I would prefer not to retell them in the following parts. Nevertheless, My progress in conducting Task 2 and Task 3 will be stated as below.

4.1 Task 2: Establishment of the database

4.1.1 Sub-task 1: Design the data structure

In order to collect the relative information of characters substitution pairs, a table should be well designed and built. I use Excel to set up the table rather than SQL, due to the consideration of efficiency and conveniency (a lot of time might be wasted in environmental configuration for SQL). I mainly designed six fields for the table to collect substitution pairs shown in Figure 3.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Original</td>
<td>Subst.</td>
<td>Pinyin</td>
<td>Round</td>
<td>R_Check</td>
<td>Further</td>
<td>Distinct A</td>
<td>1835</td>
</tr>
<tr>
<td>2</td>
<td>啊</td>
<td>阿</td>
<td>a</td>
<td>4</td>
<td></td>
<td></td>
<td>Distinct B</td>
<td>681</td>
</tr>
<tr>
<td>3</td>
<td>暖</td>
<td>爱</td>
<td>ai</td>
<td>4</td>
<td></td>
<td></td>
<td>Distinct C</td>
<td>304</td>
</tr>
<tr>
<td>4</td>
<td>哎</td>
<td>艮</td>
<td>ai</td>
<td>4</td>
<td></td>
<td></td>
<td>Distinct D</td>
<td>6</td>
</tr>
</tbody>
</table>

Figure 3. Data structure of character substitution table

Original (A) - The first field records the original characters before simplification.
Subst. (B) - The second field records the corresponding substituted characters after simplification.
Pinyin (C) - The third field records the syllables of the corresponding original characters.
Round (D) - The fourth field records the round number in which this entry were added in.
R_Check (E) - The fifth field provides the redundancy check for this entry, which can tell whether the original character for this item is distinct among the original column in this table. The function of this field is =IF(COUNTIF(A:A,A2)>1,"REP","") [43], where the number 2 is the row number of the redundancy check function. If only the original character in this row appear somewhere else among the column A, the count will be more than 1, and thus the notice “REP” will appear in the R_Check cell in this row. Otherwise, it will remain blank.
Further (F) - The sixth field serves as a comment column. It can be labeled as “1” when the substitution of corresponding entry need to be considered further.

Besides, I also worked out four functions to calculate the distinct values of the first four columns A, B, C, and D in the first four rows of column H.

Distinct A = SUMPRODUCT(1/COUNTIF(A2:A1836,A2:A1836))
Distinct B = SUMPRODUCT(1/COUNTIF(B2:B1836,B2:B1836))
Distinct C = SUMPRODUCT(1/COUNTIF(C2:C1836,C2:C1836))
Distinct D = SUMPRODUCT(1/COUNTIF(D2:D1836,D2:D1836))
4.1.2 Sub-task 2: Determine the simplify principles

The principles for simplification serves as guidance when conduct the practical substitution work. The reasonable principles is not only the basics, but can ensure the quality of the whole project as well. After the two weeks’ consideration, I worked out the following four basic principles for the further simplification.

1. All of the characters appeared in the “Subst.” field should be able to transformed into the UTF-8 code \[44\], which enables them be referred to by the java program in the next stage. The substitution candidates that is not characters themselves will not be considered;
2. The original characters that have the same syllables with their “phonetic components” will be substituted by using their “phonetic components” (ignore tones);
3. The syllables considered in the simplification process of the multi-syllables characters will be determined as their most commonly used syllables;
4. If there is a much simpler character (less strokes) whose meaning is very similar to the target character, the latter will be substituted by the former one.

4.1.3 Sub-task 3: Build a table of character simplify pairs

Up till now, I have already screened all the modern Chinese characters (about 7000), and built a substitution table of 1835 entries, which means 1835 characters have been mapped into the much simpler characters (their “phonetic component” or synonymous character). The items of the table are listed according to the lexicographical ordering, and the characters that use the same “phonetic radicals” for the substitution are put together. In order to build this large table step by step, I designed six rounds to achieve the goal. All of the substitution determined in the prior five rounds is the “phonetic component” substitution, and the synonymous characters’ substitution is considered in the last round. The statements of these six rounds are shown as below.

Round 0 - Records from the appendix 3 (pictophonetic characters arranged according to their phonetic radicals) of book Chinese Characters [45].
Round 2 - Records added during the scan of Uncomposable Characters list [46].
Round 3 - Records added during the scan of Chinese Character Components list [47].
Round 4 - Records added during the scan of 400 syllables [48].
Round 5 - Records added during the scan of the most frequently appeared synonymous characters in Chinese.

4.1.4 Sub-task 4: Collect data of word-combination for space

In order to introduce the spaces between words, a table of all possible characters combination (words) should be built in advance, only in which circumstance can the system know whether the adjoining characters in given sentence is a word or not. In the past few weeks, I searched, compared and rearranged several Chinese word-lists downloaded from the Internet. And finally, I built a huge characters
combination table of 38,291 entries, containing 23,579 two-characters word, 5,289 three-characters word, and 8,423 four-characters word (most of them are Chinese idioms). The completion of this characters combination indicates that the milestone 1 have been achieved.

4.2 Task 3: Implementation of the Java program

4.2.1 Sub-task 1: Connection with database

After the achievement of milestone 1, I began to implement the java program in class “mapping”. First of all, the substitution table for character pairs in Excel format should be able to accessed by the java class. The piece of codes that realize the function of connection with Excel is shown as follows in Figure 4.

```java
String encoding="GBK";
File file = new File(filePath);
if(file.isFile() && file.exists()){
    //determine whether file exists
    InputStreamReader read = new InputStreamReader(
        new FileInputStream(file), encoding);//encoding format
    BufferedReader bufferedReader = new BufferedReader(read);
    String lineTxt = null;
    int oriNum=0;
    int mapNum=0;
    int total=0;
    Workbook book = Workbook.getWorkbook(new File("mapping.xls"));
    Sheet sheet = book.getSheet(0);
    while ((lineTxt = bufferedReader.readLine()) != null){
        System.out.println(lineTxt);
        //System.out.print(""");
        //System.out.print(""I System.out.print("\n"I System.out.print("--->");
        StringBuffer sb = new StringBuffer(lineTxt);
```  

Figure 4. Java codes for database connection

4.2.2 Sub-task 2: Realize the characters substitution function

After successfully connect the Excel table with the java class, I moved on to realize the substitution function of character pairs. I tried to match the characters accepted from the text into the pre-built substitution table. Every time the java program find a character in testing file appears in the “Original” column of the table, it will automatically substitute this character by using the corresponding simplified character in the “Subst.” column of the same row. It records numbers during the process of substitution for the future statistics use. When it comes to the punctuation, the java program will skip and the number will not be added into the statistics analyze. The java codes of the substitution function is shown as follows in Figure 5.
The output of the substitution process of the input text “Oleander(夹竹桃)" is shown as below in Figure 6. The first lines are characters from original text, and the second lines are the simplified characters of the previous lines’ characters according to the substitution table.

然而，在一墙之隔的大门内，夹竹桃却在那里一声不响，一朵花败了，又开出一朵；一嘟噜
然而，在一墙之隔的大门内，夹竹桃却在那里一声不响，一朵花败了，又开出一朵；一嘟噜
花黄了，又长出一嘟噜。在和煦（xù）的春风里，在盛夏的暴雨里，在深秋的清冷里，看
化黄了，又长出一嘟噜。在和煦（xù）的春风里，在盛夏的暴雨里，在深秋的清冷里，看
不出什么特别茂盛的时候，也看不出什么特别衰败的时候，无日不迎风弄姿。从春天一直到
不出什么特别茂盛的时候，也看不出什么特别衰败的时候，无日不迎风弄姿。从春天一直到
秋天，从迎春花一直到玉簪花和菊花，无不奉陪。这一点韧（rèn）性，同庭院里那些花
秋天，从迎春花一直到玉簪花和菊花，无不奉陪。这一点韧（rèn）性，同庭院里那些花
比起来，不是形成一个强烈的对照吗？
比起来，不是形成一个强烈的对照吗？

Figure 6. Output of the substitution process
In order to calculate the coverage rate of the characters that could be simplified over the whole text (the amount of the characters that could be simplified out of the total amount of characters in certain text), I created two integers named “oriNum” and “mapNum” respectively. The “oriNum” records the number of the characters that remained unchanged during the simplification process, by contrast, the “mapNum” records the number of the characters that changed to a simplified counterpart in this process. Every time it a character cannot be find in the “Original” field in the substitution table, 1 will be added to the “oriNum”. Otherwise if it were found, 1 will be added to the “mapNum”. After the scan of the whole text, a statistics report will be generated based on these two statistics numbers. The codes for the statistic function is shown as below in Figure 7.

```
total=oriNum+mapNum;
double percent=(double)mapNum/total*100;
BigDecimal b = new BigDecimal(percent);
double result = b.setScale(2,BigDecimal.ROUND_HALF_UP).doubleValue();
System.out.println();
System.out.println();
System.out.println("***************Statistics***************");
System.out.println("Total amount of characters(exclude punctuation): "+total);
System.out.println("Substituted: "+mapNum);
System.out.println("Remained original: "+oriNum);
System.out.println("Percentage of substitution: "+result+" ");
read.close();
book.close();
```

Figure 7. Java codes for statistics

The display result of Statistics for a sample text is shown as below in Figure 8. We can know that the coverage rate of the substitution characters (the characters can be simplified according to the table) in this sample text is 23.89%, referring to the “percentage of substitution” row.

```
***************Statistics***************
Total amount of characters(exclude punctuation): 226
Substituted: 54
Remained original: 172
Percentage of substitution: 23.89 
```

Figure 8. Output of statistics results

4.2.3 Sub-task 3: Determine word segmentation algorithm

After the realizing of the substitution function in previous sub-tasks, the algorithm for word segmentation process should be well considered at this stage. Based on the background research stated in the previous chapter, I came up with a new algorithm for word segmentation.

When segment a given sentence, I will firstly decide whether it has ambiguity, if not, I will use the basic FMM to segment the input sentence. However, if yes, both FMM
and RMM segmentation methods will be used. And the total amounts of words segmented by using these two methods will be further compared. The method with more words matched in the dictionary (or segmented into less parts) will be regarded as correct answer. After that, the handling of unregistered words should be achieved by rules driven segmentation method. For example, several characters from the number table that followed by a character from Chinese classifier table should be regarded as a single word. Besides, a table of all Chinese family names will be built to help the system to recognize the more unregistered words related to people’s appellation.

Typically, the ambiguity detection process will be conducted with aid of a two-dimension array (based on the word-by-word scanning method given in the background chapter) shown in Figure 9. Each letter from "A" to "I" represents a character in a sample sentence.

![Figure 9. Array used in ambiguity detection process](image)

Firstly, a method that is very similar to the FMM will be used in this ambiguity detection method. The only one thing that this method different from FMM is that the search in this method begins with the longest size of words in dictionary and end with the smallest size of words in dictionary, by contrast, when find a longer size word, FMM will skip this word and switch to next string. During the process of search, I will put the characters of target sentence on the diagonal, and label the possible combinations of characters in this two dimension array. After that, I will calculate the number of items have been put into the cell in the same row or column with each character. If no items have been put into either its row or column, the character is the unregistered word. If there’s one item in the same row and one item in the same column, the character is the point where segmenting ambiguity occurs. If there are two items in its row, or column, the character is the point where combinational ambiguity occurs. Otherwise, if there’s only one item in either its row or its column, there’s no ambiguity occurs here. For example, in this case, “A” is a unregistered word, “C” encounters the segmentation ambiguity, “E” encounters the combinational ambiguity, and “HI” are normal registered word.
5 References

[26] Feng, Z.W.: The potential of ambiguity (论歧义结构的潜在性), the National Academy of Chinese information, 1995, 4
[27] Long, Q.X.: The overview of ambiguity in Chinese language (汉语歧义研究综述), Chinese learning, 2001,4
[28] Sun, M.S.: The study on automatic segmentation of Chinese words (汉语自动分词词典机制的实验研究), the National Academy of Chinese information, 2000, 14(1), 1-6


[32] Zhai, W.B.: The design of Chinese words segmentation dictionary (汉语分词词典设计), the Engineering and Application of Computer, 2007, 43(1),1-2,26

[33] Sun, M.S.: The function of maximum searching in segmenting ambiguity finding for Chinese (高频最大交集型歧义且分子短在汉语自动分词中的作用), the National Academy of Chinese information, 1999, 13(1), 27-37

[34] Introduction of algorithms for Chinese word segmentation (中文分词的几种算法), http://www.excel-easy.com/functions.html

[35] The compare of there typical Chinese word segmentation algorithms (三种中文分词算法优劣比较), http://www.htmldata.cn/?p=248


[38] Chen, G.L.: An improved Chinese word segmentation algorithm (一种改进的快速分词算法), the Study and Research of Computer, 2000, 14(1), 1-6


[40] Akpodiete, T.: Mind mapping as a Project Management Tool, PMI TPSIG Newsletter, 2007

[41] Luo, G.Q.: The study and evaluation of Chinese word segmentation technology based on reverse dictionary ordering (基于反序词典的中文分词技术研究), The research and development of computer, 2008, 18(1),80-83

[42] Luo, G.Q.: The study and evaluation of Chinese word segmentation technology based on reverse dictionary ordering (基于反序词典的中文分词技术研究), The research and development of computer, 2008, 18(1),80-83


[45] Lists of Modern Chinese Characters (现代汉语通用字表), Chinese Language Press, 1989

[46] Specification of the Undecomposable Characters Commonly Used in the Modern Chinese (现代常用独体字规范), The Education Ministry of China, 2009
