USE OF RADICAL FEATURES IN CHINESE MEDICAL TEXT MINING

A thesis submitted to The University of Manchester
for the degree of Doctor of Philosophy
in the Faculty of Science and Engineering

2021

Yifei Wang

Department of Computer Science

Contents

Word Count: 28,263

List of Tables

List of Figures

Abstract

The radical is an important feature of Chinese characters. It can represent the meaning or the pronunciation of the character. However, the use of radical features in text mining in Chinese is less concerned. The study will focus on the use of radical features in both Named Entity Recognition task and Terminology Extraction task.

By reviewing the structure of Chinese characters, phono-semantic characters show the close relationship with the radicals. A phono-semantic character has a primary radical representing its meaning and a phonetic radical representing its pronunciation. A new method is proposed to identify a phono-semantic character by looking for the phonetic radical. The test is made on *Shuowen Jiezi*, an ancient dictionary, and the F-measure at 0.802 shows it can correctly identify most of the phono-semantic characters.

Experiments using radical features have been made on both the basic machine learning method and deep learning method on named entity recognition. In deep learning method, three different embedding models using radical features are proposed. The result shows that the model uses primary radical and pinyin performs best with an F-measure at 0.709.

An advanced version of C-value, RC-value, is proposed for terminology extraction task. RC-value beats C-value with higher F-measures by testing them on two different sets of data.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright

i. The author of this thesis (including any appendices and/or schedules to this thesis) owns certain copyright or related rights in it (the "Copyright") and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.

ii. Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made **only** in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.

iii. The ownership of certain Copyright, patents, designs, trade marks and other intellectual property (the "Intellectual Property") and any reproductions of copyright works in the thesis, for example graphs and tables ("Reproductions"), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.

iv. Further information on the conditions under which disclosure, publication

and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see http://documents.manchester.ac.uk/DocuInfo.aspx?DocID=24420), in any relevant Thesis restriction declarations deposited in the University Library, The University Library's regulations (see http://www.library.manchester.ac.uk/about/regulations/) and in The University's policy on presentation of Theses

Acknowledgements

I would like to thank my supervisor, Sophia Ananiadou, and co-supervisor, Jun'ichi Tsujii, who gave me guidance for my research. I would like to thank all other staffs and students in the National Centre for Text Mining (NaCTeM), who give me academic support during my PhD programme. I would also like to thank to my parents, Changsuo Wang and Dongming Guo, who give me financial and mental support during these years. Finally, I want to thank my roommate, Jingsi Xu, who has kindly fed me for years, and his marine fish tank gave me much fun in the free time.

Chapter 1

Introduction

1.1 Research Motivation

Chinese, Spanish, English, Hindi and Arabic are the top five spoken languages in the world[lan]. Among the five languages, Chinese shows its uniqueness by the writing system. All other four languages as well as most of the other languages are using an alphabet with limited amount of letters, however, the amount of the basic element in Chinese, the Chinese characters is far greater than that of any other letters. *Shuowen Jiezi*, the dictionary published in 1st or 2nd century, collected 9,353 characters[Zho03]; *Kangxi Zidian*, published in 18th century, collected 47,035 characters;[Yip00] a recent published dictionary, *Zhonghua Zihai*, included 85,568 characters[LW94].

One of the reason why the amount of characters is quite large is that Chinese characters are logograms that represent meaning of the word or phrase, but in many other languages, the individual letters simply suggest the pronunciation. Machines that are able to comprehend the meaning of Chinese characters will make text mining tasks such as terminology extraction easier.

A radical is the basic graphical element of the Chinese character, which

usually suggest either the meaning or pronunciation. Characters in the same domain usually share the same radical. Some examples are shown in Table 1.1. Further information about radical will be discussed in the next chapter.

Character	Shared Radical
病 (illness)	疒 (sickness)
痨 (tuberculosis)	疒 (sickness)
痛 (pain)	疒 (sickness)
肝 (liver)	月 (moon)/肉 (meat)
胸 (chest)	月 (moon)/肉 (meat)
脑 (brain)	月 (moon)/肉 (meat)
江 (river)	氵 (water)/水 (water)
海 (sea)	氵 (water)/水 (water)
湖 (lake)	氵 (water)/水 (water)

Table 1.1: Shared Radicals in Chinese Characters

In the medical domain, many characters share the same radical, such as characters describing diseases and characters describing organs. Knowing the name of a disease is straightforward if the name of the radical is understood. Using radicals in text mining will help medical text mining.

Apart from radicals presenting meaning, some radicals suggest the pronunciations of the characters as well. In the Chinese biomedical domain, many terms come from other languages like English and German. Most these terms are transliterations, which are pronounced similarly to their original pronunciation. The use of a radical might help a machine to understand how terms are originally pronounced in their original language.

Therefore, in this thesis, we will explore how radical could help in terminology extraction and named entity recognition in Chinese medical texts.

1.2 Research Aim and Objectives

The main aim of the research is to learn how radical features can be used in text mining of Chinese medical text. Because the radical is a basic feature of Chinese characters, so named entity recognition and terminology extraction will be focused on the research. These two tasks are highly related to the characters and their meaning.

To better address the problem, the following research questions are proposed:

Q1: Do all Chinese characters have radicals and are their radicals highly related to their meaning? If not, what is the percentage of such characters among all Chinese characters?

Q2: Do all radicals represent the meaning of the characters? If there are other kinds of radicals, can they be used in the text mining tasks?

Q3: How can radical features be applied to named entity recognition?

Q4: Do radical features improve the performance of named entity recognition?

Q5: How can radical features be applied to terminology extraction?

Q6: Do radical features improve the performance of terminology extraction?

Before starting the research, the following hypotheses are made:

H1: Although not all Chinese characters have radicals that represent their meaning, the majority of Chinese characters do have radicals that represent their meaning.

H2: Some radicals that are not related to the meaning still have value in text mining.

H3: Radical features can improve the tasks of Named Entity Recognition and Terminology Extraction.

Based on the research questions and hypotheses, the following objectives are established:

O1: To review the special features of Chinese characters and determine the categories of Chinese characters.

O2: To determine the percentage of Chinese characters that have radicals that represent meaning.

O3: To propose a method using radical features for Named Entity Recognition.

O4: To propose a method using radical features for Terminology Extraction.

1.3 Thesis Structure

The structure of the whole thesis is listed below.

- Chapter 1 is the introduction about the research;
- Chapter 2 will focus on **Q1** and **Q2**, covering **O1** and **O2**, it will introduce and discuss some Chinese character features, and will also talk about some Chinese processing problems and how they could be solved;
- Chapter 3 will focus on **Q3** and **Q4**, covering **O3**, address named entity recognition and show how the radical feature can be used in basic machine learning and deep learning methods;
- Chapter 4 will focus on **Q5** and **Q6**, covering **O4**, go through terminology extraction and show how the radical feature can be used in classic statistical methods;

- Chapter 5 will be the conclusion of the whole work.

1.4 List of Translations

This thesis will focus on Chinese language processing. For some Chinese language terms, there are different translations in different papers and articles. For better clarification, Table 1.2 will show the Chinese terms and the translations used in this thesis.

Chinese	English	Meaning
偏旁	radical	basic graphical element of characters
部首	primary radical	radical used for sorting and ordering in the ancient dictionary
形声字	phono-semantic character	characters containing a radical representing the meaning and another radical representing the pronunciation
形旁	phonetic radical	radical representing the meaning in a phono-semantic character
声旁	semantic radical	radical representing the pronunciation in a phono-semantic character
笔画	stroke	a mark that can be written without lifting the pen from the paper
声母	initial	similar to the consonant in English
韵母	final	similar to the vowel in English
介音	prenuclear glide	sound between initial and final

Table 1.2: Translation of Chinese terms used in this thesis

Chapter 2

Chinese Processing

2.1 Chinese Character Features

2.1.1 Radical

A radical is the basic graphical element of Chinese characters; most Chinese characters are made up of more than one radical. In the ancient times, the radical was the only possible feature to sort the characters in a dictionary. Characters were sorted according to their primary radical, which is usually the one that represents the meaning of the character. Table 2.1 shows some characters with their primary radicals. It is clear that the characters related to sickness share the same primary radical, as do the characters represent the organs.

Character	Primary Radical
病 (illness)	疒 (sickness)
痨 (tuberculosis)	疒 (sickness)
痛 (pain)	疒 (sickness)
肝 (liver)	月 (moon)/肉 (meat)
胸 (chest)	月 (moon)/肉 (meat)
脑 (brain)	月 (moon)/肉 (meat)

Table 2.1: Primary Radicals in Chinese Characters

The last three characters in Table 2.1 show the simplification of the radicals. The shape of their primary radical is 月, which means moon. But they are all names of organs, which have no relationship with moon. It is because the primary radicals were originally 肉, which means meat. During the simplification of the characters, their primary radicals finally came to look like 月 [Li05]. Fig 2.1 shows the process of simplification of 肝 (liver), 月 (moon) and 肉 (meat).

(a) 肝

(b) 月

(c) 肉

Figure 2.1: The simplification process of 肝 (liver), 月 (moon) and 肉 (meat)

Besides the primary radical providing semantic information, some other radicals may provide the phonetic information. These are called phonetic radicals. Table 2.2 shows the phonetic radicals of the characters in Table 2.1 and their pronunciation in pinyin. Pinyin is the Romanisation system for Chinese representing the pronunciation of the Chinese characters, which will be introduced in detail in the following section. Those characters all share similar pinyin with their phonetic radicals, which are also the characters themselves.

There are some other phonetic radicals that do not have similar pronunciations with the characters, but the characters sharing the same radical have the similar pronunciation. Table 2.3 shows some examples of them.

Character	Pinyin	Phonetic Radical	Pinyin of Phonetic Radical
病 (illness)	bìng	丙 (third)	bǐng
痨 (tuberculosis)	láo	劳 (labour)	láo
痛 (pain)	tòng	甬 (path)	yǒng
肝 (liver)	gān	干 (do)	gàn
胸 (chest)	xiōng	匈 (ancient form of 胸)	xiōng
股 (thigh)	gǔ	殳 (pike)	shū

Table 2.2: Phonetic Radicals in Chinese Characters

Character	Pinyin	Phonetic Radical	Pinyin of Phonetic Radical
脔 (sliced meat)	luán	亦 (also)	yì
峦 (hill)	luán	亦 (also)	yì
孪 (twin)	luán	亦 (also)	yì
脾 (spleen)	pí	卑 (low)	bēi
啤 (beer)	pí	卑 (low)	bēi
裨 (vice)	pí	卑 (low)	bēi

Table 2.3: Other Phonetic Radicals in Chinese Characters

As said, there are many Chinese characters that have a primary radical that presents the semantic information and a phonetic radical that shows the phonetic information. But not all Chinese characters follow this rule; the characters that follows this rule made of a primary radical and a phonetic radical are called phono-semantic characters.

Actually, the formal name of the radical presenting the semantic information in a phono-semantic character is semantic radical. Most of the characters will have their primary radicals as the semantic radicals; only few of them have a semantic radical different from the primary radical. As it is difficult to get the semantic radical of a character, to simplify the problem, primary radical will be used as the semantic radical in this thesis.

It is believed that more than 90% of Chinese characters are phono-semantic[Bol94]. For a phono-semantic character, which is made of a primary radical and a phonetic radical, it is possible to get the information that this character contains by viewing its primary radical and phonetic radical, which suggests its meaning and

pronunciation.

2.1.2 Indicating Pronunciation

To better get the phonetic information of phonetic radicals, it is necessary to know the pronunciation of a character. As Chinese characters are logograms, which presents the meaning instead of the pronunciation, so other system are required to indicate how to pronounce a character.

A simple system already shown in Tables 2.2 and 2.3 is pinyin. Pinyin is a Romanization system for Chinese, which represents the pronunciation of a Chinese character in Latin letters. The pinyin of a character contains three parts: initial, final and tone. Initials and finals are similar to the consonants and vowels in English except that there can be only one initial and one final in the pinyin of a character. There are five different tones in the pinyin: flat tone (¯), rising tone (´), falling-rising tone (ˇ), falling tone (`) and neutral tone ( ).

As shown in Table 2.2, the pinyin of 病 is bìng, where b is the initial, ing is the final and ì shows that the tone is falling tone. 丙, the phonetic radical of 病, has its pinyin as bǐng. In this case, only the tones are different in the phonetic radical and its original character. Actually all the characters in Table 2.2 share at least the same finals with the phonetic radicals.

All initials and finals in pinyin are shown in Figure 2.2 and Table 2.4 [oEotPRoC15].

p m f d t n l g k h j q x z c s zh ch sh r *y* *w*

Figure 2.2: Initials in Pinyin

In Fig. 2.2, *y* and *w* are the special cases. In *Scheme for the Chinese Phonetic Alphabet* published by Ministry of Education of the People's Republic of China[oEotPRoC15], *y* and *w* are not in the initial list. But *y* and *w* will be

	i	u	ü
a	ia	ua	
o		uo	
e	ie		üe
ai		uai	
ei		uei	
ao	iao		
ou	iou		
an	ian	uan	üan
en	in	uen	ün
ang	iang	uang	
eng	ing	ueng	
ong	iong		

Table 2.4: Finals in Pinyin

considered as the initial when there are no initial and the final is *i* and *u*, respectively.

Table 2.4 including the first row and first column shows all the finals in pinyin. Among them, finals at the last 5 rows are special. They are nasal finals, which means the process of pronouncing these finals involves a nasal sound produced through the nose. There are two kinds of nasal finals in pinyin: front nasal finals and back nasal finals. Finals in the rows starting with *an*, *en* are front nasal finals; the pronouncing of these finals use the front part of the nose. Finals in the rows starting with *ang*, *eng*, *ong* are back nasal finals; the pronouncing of these finals use the back part of the nose.

Besides initial and final, another important concept of Chinese pronunciation is the prenuclear glide, which is between initial and final, helping to transfer from initial to final. But not all the pronunciation of characters involve prenuclear glides. 病, the example mentioned before, does not have the prenuclear slide. The pinyin of 胸 is xiōng, as shown in Table 2.2. In this example, x is the initial, ōng is the final with the flat tone, and i between x and ōng is the prenuclear glide. *i*, *u*, *ü* shown in the first row in Table 2.4 are the only three prenuclear glides

in pinyin. Table 2.4 shows that even in the official file of pinyin, the prenuclear glide is considered as part of the final. It makes it difficult to get the prenuclear glide of a given pinyin.

Besides pinyin, there are also some alternative candidates as the pronunciation schemes to pinyin. Zhuyin (also called Bopomofo) is another transliteration system, which is widely used in Taiwan. Table 2.5 shows the zhuyins of the characters in Table 2.2.

Character	Pinyin	Zhuyin
病	bìng	ㄅㄧㄥˋ
痨	láo	ㄌㄠˊ
痛	tòng	ㄊㄨㄥˋ
肝	gān	ㄍㄢ
胸	xiōng	ㄒㄩㄥ
背	bèi	ㄍㄨㄟ

Table 2.5: Zhuyins of Chinese Characters

Unlike pinyin that uses Latin letters, the zhuyin system uses a set of unique symbols to annotate the pronunciation. There are also initials, finals, prenuclear glides and tones in the zhuyin system. In the zhuyin system, prenuclear glides are separated from finals. ㄧ, ㄨ and ㄩ are the only prenuclear glides in zhuyin system, and they can be found at the zhuyin of 病, 痛 and 胸 in Table 2.5, respectively. It should be noted that the tones in the zhuyin system are separated from the final, which are quite different from those in pinyin. When the tone of a character is flat tone (¯), it will be ignored.

Fanqie is another system to annotate the pronunciation. Fanqie is widely used to annotate the pronunciation in the ancient dictionaries. In this system, the pronunciation of a character is annotated by two other characters A and B. This annotation shows that the pronunciation of the character is the combination of the pronunciations of A and B. Usually A will provide the initial and B will

provide the final. Table 2.6 shows the fanqie in *Kangxi Zidian* of the characters in Table 2.2.

Character	Pinyin	Fanqie	Pinyins of Fanqie	Combination
病	bìng	皮 命	pí mìng	pìng
痨	láo	郎 到	láng dào	lào
痛	tòng	他 贡	tā gòng	tòng
肝	gān	古 寒	gǔ hán	gán
胸	xiōng	许 容	xǔ róng	xóng
股	gǔ	公 戶	gōng hù	gù

Table 2.6: Fanqie of Chinese Characters

Following the rule of fanqie, it is possible to get the predicted pronunciation of a character in the last column of Table 2.6. These pronunciations are quite similar to the real pronunciations. However, from the table, we can find the problem of fanqie. There is not a rule about which characters should be used for annotating. Even in the same dictionary, the same initial or final may be presented by different characters, and different dictionaries could cause more problems. An example can be found in Table 2.6, where both 公 and 谷 are used to annotate the initial *g*.

All three pronunciation features have advantages and disadvantages.

- Pinyin is the most widely used system in the world, and using Latin letters makes it easy to understand.

- The format of pinyin makes it difficult to separate the tone with finals and to get the prenuclear slides, if there are any.

- The format of zhuyin makes it easy to get the initials, prenuclear glides, finals and the tones.
- Zhuyin uses unique symbols that are not used in anywhere else.
- Fanqie uses the Chinese characters to annotate the pronunciation, which makes the processing only focus on Chinese characters.
- There is no standard for which characters should be used for annotating, even in the same dictionary.

Although the advantages make it beneficial to use zhuyin and fanqie for processing, it is almost impossible to use them because the symbols of zhuyin can only be understood by Taiwanese and there is no standard of using fanqie. As a result, pinyin will be used for Chinese processing in this thesis.

2.2 Categories of Chinese Characters

The phono-semantic characters are the characters that have a primary radical representing the meaning and a phonetic radical representing the pronunciation. Phono-semantic character is just one of the categories of Chinese characters. In the most famous dictionary in ancient China, *Shuowen Jiezi*[Xury], the characters are classified into six different categories: ideograms (指事), pictographs (象形), phono-semantic compounds (形声), compound ideographs (会意), derivative cognates (转注) and phonetic loan characters (假借).

All the description and examples below comes from the preface of *Shuowen Jiezi*[Xury], so that all the meaning annotated is the meaning in ancient Chinese, although some of the characters have quite different meaning in modern Chinese.

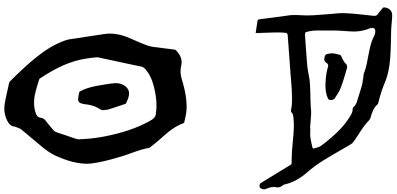
Ideograms are the characters showing the meaning by the graphs. The meaning of ideograms are usually concepts. Fig. 2.3 shows the ancient form of 上 (up)

and 下 (down), two examples of pictographs.

(a) 上 (up) (b) 下 (down)

Figure 2.3: Examples of ideograms in ancient form

Pictographs are believed to be the most ancient characters. Most of the pictographs are nouns representing the things in the real world, and the writing of these characters are graphs showing what they mean. Fig. 2.4 shows the ancient forms of 日 (sun) and 月 (moon), two examples of pictographs.



(a) 日 (sun) (b) 月 (moon)

Figure 2.4: Examples of pictographs in ancient form

Phono-semantic compounds have been introduced before. They are formed by a primary radical representing the meaning and a phonetic radical representing the pronunciation.

Compound ideographs are the characters formed by two or more other characters, and the meanings of compound ideographs highly relate to meaning of the characters formed. For example, 武 (military) is formed by 戈 (an ancient weapon like axe) and 止 (foot); 信 (truthful) is formed by 亻, the simplification of 人 (human), and 言 (speech).

Derivative cognates are the pairs of characters sharing same meanings, and

one of them changes the meaning. The example given by *Shuowen Jiezi* is 考 (elder) and 老 (elder), where 考 no longer means elder anymore in modern Chinese. It is the most confused category among the six and there are still discussion of the exact definition of derivative cognates now. One of the opinions is that there is no need to care about the exact definition of derivative cognates in the research of Chinese characters, as there are only few of them now[Wan11].

Phonetic loan characters are the characters formed by borrowing the characters having the same pronunciations. The examples given by *Shuowen Jiezi* are 令 (order) and 长 (officer or long). Why these characters are identified as phonetic loan characters is another confusing problem. Another example given by Liu [Liu05] is 我 (I). 我 originally means a kind of sow or weapon, pronounced the same as the first person singular (in English, *I*). However, at that time, there was not a character that represented the first person singular, so 我 was used because it had the same pronunciation as the first person singular.

Among all six categories, derivative cognates and phonetic loan characters are less frequently used, so that they can be ignored. The ideograms and the pictographs usually cannot be separated into different radicals. The phono-semantic compounds and compound ideographs have their radicals representing their meanings. In phono-semantic compounds, the semantic radicals suggest the meaning of the characters, while in compound ideographs, all of its radicals suggest parts of the meanings of the characters. In *Shuowen Jiezi*, a primary radical is assigned to each character, and all characters are sorted by their primary radicals. For phono-semantic characters, their semantic radicals are used as the primary radicals, and for compound ideographic characters, their most meaningful radicals are used as the primary radicals. Because the ideograms and the pictographs cannot be separated, the characters themselves work as their primary radicals.

In the preface of *Shuowen Jiezi*, it also states that with the development of the Chinese characters, most new characters are phono-semantic. As it is the dictionary written almost 2000 years ago, so most currently used Chinese characters should be phno-semantic characters.

2.3 Previous Work on Graphical and Semantic Features

As stated, most Chinese characters are phono-semantic characters, which express their meaning by the primary radicals and pronunciation by their semantic radicals. This property should be useful in Chinese text mining task. This section will explore the previous work on the graphical features (such as primary radical) and semantic features (such as pinyin).

2.3.1 Graphical Feature

As Chinese characters are logograms, there have many attempts to use the graphical information contained by Chinese characters in Chinese language processing.

Part-Of-Speech

It is easy for people to guess the part-of-speech (POS) of a given character through its primary radical. For example, most of the characters related to illness, which share the primary radical 疒, are nouns, and most of the characters related to hand, which share the primary radical 手/扌, are verbs. So, the early research about radical features mostly focus on POS tagging.

Wang et al.[WCL09] apply the primary radical feature on POS tagging using

the SVMTool, the overall accuracy is improved to 97.90%. By applying the primary radical, the accuracy of pos on out-of-vocabulary words becomes 85.27%, which is much higher than using other features such as suffix, where the accuracy is 84.32%.

Other works also show the importance of primary radical on unkown word. In the work of Zhang et al.[ZSFH09], the primary radical feature is applied to Conditional Random Fields to guess the POS tag of the unknown words, and get a precision at 94.67%. The work of POS tagging on unknown words by Tseng et al.[TJM05] use the primary radical feature in maximum entropy Markov model, and proves that the primary radical helps most on identifying nouns.

Character Embedding

With the use of deep learning, the radical feature, which shows much information about the character itself, is applied in more research than POS tagging alone.

In the work of Shao et al.[SHTN17], the primary radical feature and graphical features extracted from the picture of characters are used for joint word segmentation and POS tagging, and show that the use of the radical feature in the RNN-CRF model performs best.

Shi et al.[SZY+15] introduce Radical Embedding in their work, where a character is treated as a bag of radicals to train the embedding. The model helps in the task of Short-Text Categorisation (STC) and Chinese Word Segmentation (CWS). The same embedding model is improved by dealing with the simplification of radicals and tested in named entity recognition task by Dong et al.[DZZ+16], and the performance is improved.

In the embedding model of Sun et al.[SLY+14], the primary radical feature is captured in C&W model[CWB+11], and the enhanced embedding improves the

performance of CWS.

Li et al.[LLSL15] modify the existing embedding models to accept the radical feature for word similarity and text classification tasks.

One special use of the radical feature is made by Zhang et al.[ZM17] in the task of machine translation between Japanese and Chinese, both of which use characters sharing the radical feature.

In the task of Named Entity Recognition (NER), many attempts using graphical feature have been made. In the model proposed by Yu et al.[YJXS17], the primary radicals are used. The models proposed by Cao et al.[CLZL18] and Wu et al.[WMH+19] show the uniqueness by using other graphical features rather than the primary radicals only like other works stated above.

The model of Cao et al. uses strokes, the most basic graphical component of the Chinese characters. The model of Wu et al. uses the pictures of the character and split each of them into four parts for embedding. These three models will be explained and described in the next chapter.

2.3.2 Semantic Feature

There is some research about using pinyin, the pronunciation feature. Some studies, such as the work of Jin et al.[JCGL14], tries to correct the misspelled pinyin during inputting Chinese, a problem mentioned by Chen et al.[CL00]. Others, such as the work of Lin et al.[LZ08], try to improve the performance in pinyin-Chinese conversion, which benefits the Chinese input method. Zhang et al.[ZZL15] convert a Chinese corpus to pinyin form to use the convolutional network designed for English on Chinese text.

Unlike radical, which is a graphical feature, there are few attempts to use pinyin as part of embedding. It is possibly because pinyin completely destroy the

logographic structure of characters. However, most Chinese characters are phono-semantic characters, containing one primary radical and one phonetic radical. It should be easy to extract the information contained in the phonetic radical through the pinyin of the character.

2.3.3 Limitation

Most of the research on graphical features use only primary radicals, however, as stated before, most of the Chinese characters are phono-sematic characters, containing not only the primary radical but also the semantic radical.

The pronunciation information provided by the semantic radical could help in text mining tasks by limiting the effect of typos in the documents, understanding the phonetic loan characters in the documents, understanding the transliteration words and so on.

But for semantic features, most of the works focus on some Chinese only tasks, such as misspelled pinyin correction and pinyin-Chinese conversion stated above. Few attempts have been made on some general task such as NER.

Based on the limitation stated above, an experiment should be made on exploring how the graphical and semantic feature can be involved in a general task to verify the hypothesis that the property of the phono-semantic characters is useful in Chinese text mining. The experiment will be introduced in Chapter 3.

Another limitation is that except the obvious POS tasks and the character embedding model in the deep learning methods, there are few attempts on statistical methods using the radical feature. For example, statistical methods were the main methods for Terminology Extraction(TE) tasks, different measurements have been proposed, such as C-value[FAM00].

But none of the research has tried to use graphical or pronunciation feature in statistical methods. Although compared to the performance of the deep learning methods, the performance of the statistical methods are not so good. The statistical methods are still useful when lacking suitable annotation documents. It is worth exploring if graphical feature is useful in statistical methods. The experiment will be introduced in Chapter 4.

There is one more limitation that none of the research tries to figure out the difference between the performance on Simplified Chinese and Traditional Chinese when using the graphical features of the Chinese characters. The graphical features may be changed when converting between Simplified Chinese and Traditional Chinese, so that when using graphical features, the performance should be slightly different. This problem will be discussed in the following section and focused in each experiment.

2.4 Problems in Chinese Processing

2.4.1 Simplified and Traditional Chinese

Introduction

It is well-known that two kinds of Chinese characters are used in the world today: Simplified Chinese and Traditional Chinese.

Simplified Chinese is mainly used in mainland China and some countries in Southeast Asia, while Traditional Chinese is mainly used in Hong Kong and Taiwan.

Usually, the variant will not cause any problems in processing because the text will be converted during preprocessing. However, when radicals are involved, things become different.

As the name suggests, Simplified Chinese simplifies a huge number of characters in Traditional Chinese. As shown in Figure 2.1, during the process of simplification, the radical information may be destroyed. Table 2.7 shows some examples of the loss of the semantic information that primary radical contains or the phonetic information contained by phonetic radical. Table 2.7 shows seven pairs of characters In each pair, the top character is in Simplified Chinese and the bottom character is the corresponding character in Traditional Chinese.

Pair	Character	Pinyin	Primary Radical	Phonetic Radical	Pinyin
1	产 (produce)	chǎn	立 (stand)	N/A	N/A
	產 (produce)	chǎn	生 (born)	N/A	N/A
2	颅 (skull)	lú	页 (head)	卢 (cottage)	lú
	顱 (skull)	lú	頁 (head)	盧 (cottage)	lú
3	庐 (cottage)	lú	广 (wide)	户 (door)	hù
	廬 (cottage)	lú	广 (wide)	盧 (cottage)	lú
4	广 (wide)	guǎng	广 (wide)	N/A	N/A
	廣 (wide)	guǎng	广 (wide)	黄 (yellow)	huáng
5	肮 (dirty)	āng	肉 (meat)	亢 (high)	kàng
	骯 (dirty)	āng	骨 (bone)	亢 (high)	kàng
6	脏 (dirty)	zāng	肉 (meat)	庄 (village)	zhuāng
	髒 (dirty)	zāng	骨 (bone)	葬 (bury)	zàng
7	脏 (viscera)	zàng	肉 (meat)	庄 (village)	zhuāng
	臟 (viscera)	zàng	肉 (meat)	藏 (hide)	cáng
8	舰 (warship)	jiàn	舟 (boat)	见 (see)	jiàn
	艦 (warship)	jiàn	舟 (boat)	監 (supervise)	jiān
9	灭 (extinguish)	miè	大 (big)	N/A	N/A
	滅 (extinguish)	miè	水 (water)	威 (extinguish)	miè
10	邮 (mail)	yóu	邑 (district)	由 (cause)	yóu
	郵 (mail)	yóu	邑 (district)	N/A	N/A

Table 2.7: Simplified Chinese and Traditional Chinese

Primary Radical

Jia[Jia03] states the opinion that the simplification of the characters cause six different cases of primary radical changes:

1. 25% of the characters still hold their primary radicals in Traditional Chinese, which is the case in Pair 3, 7 and 8;

2. 1% of the characters delete the other parts and have only their primary radicals left, which is the case in Pair 4;

3. 12% of the characters delete their primary radicals and have something meaningless as their new primary radical, which is the case in Pair 1;

4. 24% of the characters have their primary radicals simplified, which is the case in Pair 2;

5. 27% of the characters change their primary radicals to another existing primary radical, which is the case in Pair 5 and 6;

6. 11% of the characters do not have any existing sensible primary radical, and it is difficult to determine the new primary radicals, which is the case in Pair 9.

Case 1, 2 and 4 cannot influence the Chinese processing, because the characters still have the same primary radical, although some of them change their appearances. Case 5 might affect the processing of radicals, as the meaning of the character may be related to the new primary radical. In Case 3 and 6, the primary radicals of the characters are changed to something meaningless and completely destroy the rule that the primary radical should contain the semantic information.

Phonetic Radical

For phonetic radicals, the simplification of Chinese characters causes the following cases:

1. Non-phono-semantic characters remain as they are, which is the case in Pair 1;

2. Non-phono-semantic characters are simplified into phono-semantic characters, which is the case in Pair 10;

3. Phono-semantic characters are simplified into non-phono-semantic characters, which is the case in Pair 4 and 9;

4. Phonetic radicals in Simplified Chinese are the same as they are in Traditional Chinese, which is the case in Pair 5;

5. Phonetic radicals are simplified into other phonetic radicals, which is the case in Pair 2;

6. Other characters are used as phonetic radicals in Simplified Chinese, which is the case in Pair 3, 6, 7 and 8.

Case 1 will definitely not influence the processing of the phonetic radical, as the characters are not phono-semantic characters and thus have no phonetic radicals. In Case 2, some non-phono-semantic characters become phono-semantic characters, and so will be useful when processing phonetic radicals. In Case 3, although most phono-semantic characters abandon their compound rules because their phonetic radicals are almost never used anymore, machine learning can be taught the pronunciation of rarely-used characters. In Case 4, the phonetic radicals are unchanged, so it will not affect the result. Case 5 is similar to Case 4 in primary radicals, which should not cause any problems to processing with phonetic radicals. In Case 6, some examples use a more accurate phonetic radical like pair 8, and some make the pronunciations of the characters and phonetic radicals more different. It is hard to tell how case 6 will affect the processing with phonetic radicals.

Discussion

As the changes of primary radicals and phonetic radicals, when a new method using radical features is proposed, it is worth testing the method in both Simplified Chinese and Traditional Chinese to see if any different results will come.

As stated above, in the process of simplification, some characters lose their original sensible primary radicals or semantic radicals (Case 3 and 6 of the primary radicals and Case 3 of the phonetic radicals), and some get more sensible primary radicals or semantic radicals (Case 5 of the primary radicals and Case 5 and 6 of the phonetic radicals). It is difficult to tell whether the performance of the model using the property of the phono-semantic characters is better on Simplified Chinese or Traditional Chinese.

The conversion between Simplified Chinese and Traditional Chinese is also a problem in Chinese processing. The difficulties in conversion have been categorised into three situation by Wang et al.[WW08]:

1. It is difficult to deal with many-to-one or one-to-many problem. Some different characters in Traditional Chinese have the same form in Simplified Chinese, and vice versa. In this situation, the simplification should be made based on the meaning of the character.

2. The words used in mainland China, Hong Kong, Taiwan and other different areas are different due to the dialect. It is like the situation of American English and British English; fries in American English is actually chips in British English.

3. Some characters, usually place names in mainland China, should not be converted.

The opinion of Wang et al. only focuses on the conversion problems between

mainland China, Hong Kong and Taiwan. However, just as the name suggests, Traditional Chinese is also the version used in the ancient time, and the dictionaries mentioned before as *Shuowen Jiezi* and *Kangxi Zidian* are both in Traditional Chinese. The problem of conversion between modern Simplified Chinese and non-modern Traditional Chinese is identified by Xiamen University[SCH11]. *Simplified and Traditional Chinese Intelligence Converting System*[CSZ11], the system invented by Xiamen University, can cover the situation of conversion between Traditional Chinese in classical literature and modern Simplified Chinese. This system will be used when conversion of ancient Traditional Chinese is required.

In modern Chinese, Traditional Chinese in Taiwan is also different from that in Hong Kong. In the conversion between modern Chinese, Traditional Chinese in Taiwan should be used because Traditional Chinese used in Hong Kong includes some representation of Cantonese. And a open-source package called OpenCC[1] will be used to convert Simplified Chinese to Traditional Chinese in Taiwan.

2.4.2 Chinese Word Segmentation

Introduction

It is known that in the writing of Chinese, there are no spaces between words. The example shown in Figure. 2.5 is a typical word segmentation ambiguity of the phrase 南京市长江大桥 in Chinese.

It can be found that the different segmentation of a phrase leads to two quite different meanings, the first one refers to a bridge, while the second one is a person. So a unique task for Chinese is generated, Chinese Word Segmentation (CWS). Chinese Word Segmentation is one of the focus of Chinese Text Mining, there are many works done on the task. Xue et al.[XS03] first transfer the task to

[1] `https://github.com/BYVoid/OpenCC`

Segmentation:

南京市 \ 长江 \ 大桥

Translation of each segment:

Nanjing City\ Yangtze River\ Bridge

Translation of the phrase:

Nanjing Yangtze River Bridge

(a) Segmentation A

Segmentation:

南京 \ 市长 \ 江 \ 大桥

Translation of each segment:

Nanjing\ Mayor \ Jiang (Surname) \ Daqiao (Given name)

Translation of the phrase:

Mayor of Nanjing, Daqiao Jiang

(b) Segmentation B

Figure 2.5: Word Segmentation ambiguity in Chinese

a character tagging problem. Special Interest Group on Chinese Language Processing (SIGHAN) held some bakeoffs on CWS [SE03][Eme05]. Several different approaches has been done on CWS task, however, there is not a perfect standard in CWS task, most of the current works use the dataset of the bakeoffs as the standard.

In the work of Sproat et al.[SGSC96], six Chinese native speakers are asked to segment the same test corpus, and the result is shown in Table 2.8, where *M1*, *M2* and *M3* are the native speakers from mainland China, and *T1*, *T2* and *T3* are the native speakers from Taiwan, China.

It can be found that even for Chinese native speakers, they will segment the text in different ways.

Similarity	M1	M2	M3	T1	T2	T3
M1		0.77	0.69	0.71	0.69	0.70
M2			0.72	0.73	0.71	0.70
M3				0.89	0.87	0.80
T1					0.88	0.82
T2						0.78

Table 2.8: Similarity of the word segmentation by different native speakers

In 1992, General Administration of Quality Supervision, Inspection and Quarantine of China publish a standard about the rules of Chinese Word Segmentation, *GB13715*[GAoQSQ92]. SIGHAN also follows this standard in the preparation of the dataset.

There are many controversies about this standard after publishing. According to Song[Son97], Li et al.[LCJ+07] and Huang et al.[HGL03], several problems are listed about the standard:

1. Segmentation about reduplication is meaningless. In Chinese, there are some words in the form like *AAB*, *ABB*, *AABB* and *ABAB*, those words usually have the same meaning of *AB* but are more artful or are used for emphasising. These words are called reduplication (叠词) in Chinese. For example, 开开心 (have fun), whose meaning is same as 开心 (have fun). However, following the standard in *GB13715*, 开开心 would be segmented as 开开 (meaningless, *open open* by literal translation)\ 心 (heart), causing one word segmented is meaningless and the meaning of the whole phrase is difficult to explain.

2. Segmentation about special suffixes is meaningless. For example, 教育部长 (Chief of the Department of Education) would be segmented as 教育 (education) \ 部长 (chief of the department) following the standard in *GB13715* instead of 教育部 (Department of Education) \ 长 (chief). Such segmentation also makes whole phrase difficult to explain the meaning.

3. No standard on abbreviation. There is no rules about how abbreviation should be segmented in *GB13715*, such as 中英关系 (relationship between China and the UK), where 中 is the abbreviation of 中国 (China) and 英 is the abbreviation of 英国 (the UK).

4. Place names cannot be segmented well. If a place name is a combination of other place names, it will be considered as a whole rather than different segments. For example, 河南省郑州市中原西路 233 号 (No. 233, Zhongyuan West Road, Zhengzhou City, Henan Province) will not be segmented as 河南省 (Henan Province) \ 郑州市 (Zhengzhou City) \ 中原西路 (Zhongyuan West Road) \ 233 号 (No. 233), the whole phrase will be considered as a word.

5. Some segmentations about numeral break the meaning of the word. For example, 百分之零点八三 (0.83 percent) would be segmented as 百 (hundred)\ 分之 (fraction of)\ 零点八三 (0.83).

6. Some words in *GB13715* is not clear. For example, it is written that *the tight-coupling and stable-used two-character compounds and three-character compounds are all the basic segmented units* in *GB13715*. However, there is not a definition of *tight-coupling* and *stable-used*, which makes different persons will segment differently. It also makes the use of the standard difficult.

The imperfect standard of word segmentation will obviously cause problems to other tasks. For example, based on different segmentation of 河南省郑州市中原西路 233 号 mentioned above, different place names will be found in Named Entity Recognition task.

Solutions

There are two possible methods to avoid the problems of Chinese Word Segmentation. The first one is to use a character-based method rather than word-based method. There are always two kinds of methods in the processing of Chinese text, the character-based methods and the word-based methods. Word segmentation is usually the first step of the word-based methods, and the use of character-based methods can avoid CWS.

Another benefit in this research of using character-based method is that radical features are the basic features of characters. The features should be better presented in the character-based method than that in the word-based method.

Besides that, applying character embedding will not meet out-of-vocabulary problem. Although there are still some new invented characters. However, if a text is readable by machine, which means all the characters are encoded by some standard, which is most probably Unicode, so that the information about the characters could be found in the database provided by Unicode Inc..

The second one is to use the text that do not require word segmentation at all, one of the options is Classical Chinese. Classical Chinese is a widely-used writing language in Confucianism culture sphere, including China, Japan, Korea and Vietnam. Classical Chinese has been used as the official writing language in China until the early 20th century. Compared with Modern Chinese, it has slightly different grammars and different meanings for a same character.

In Classical Chinese, most words are one-character word. In the work of Huang et al.[HPWW02] of POS tagging, word segmentation is not applied at the prepossessing, because the words in Classical Chinese are written in one-character form. However, as Classical Chinese has been used from the 20th century B.C. until the early 20th century A.D., the writing styles are quite different in different

periods. The work of Huang et al. is applied on *Lunyu*, *Daodejing* and other books written in the 5th century B.C., in these text, one-character words are very common. While in the work of Xiong et al.[XLL+13] of name entity recognition on novels of Ming and Qing dynasties (around 15th century to 19th century), word segmentation is necessary because one-character words become less frequent in these text. It means that when using Classical Chinese text for avoiding CWS problems, the text should be written in some early ages (e.g., before 10th century A.D.) to make sure one-character words are still the majority of the words in the text.

It is obvious that biomedicine has not been developed before 10th century A.D., so when using Classical Chinese to avoid CWS, the data will be focused on Traditional Chinese Medicine.

In order to avoid CWS problem, these two methods have been applied. In Chapter 3, the use of character-based method will be applied to Named Entity Recognition. In Chapter 4, the text in Classical Chinese will be used for Terminology Extraction.

2.5 Identifying Phono-semantic Characters

2.5.1 Introduction

Phono-semantic character is just one of the six categories of characters summarised by *Shuowen Jiezi*. Not all characters are phono-semantic characters with a phonetic radical. Boltz et al.[Bol94] stated that more than 90% of Chinese characters are phono-semantic characters, Hoosain et al.[Hoo13] believed that over 80% of them are phono-semantic. Although there is some disagreement about exactly how many phono-semantic characters are there among all the characters,

the majority of the characters are phono-semantic characters.

It is necessary to know the proportion of the phono-semantic characters among the text when processing the text using primary radical and phonetic radical features. However, there is not a proper method to check whether a given character is phono-semantic currently. In daily life, people will only consider a character as a phono-semantic character if there is a radical having the pronunciation similar to the pronunciation of the character or there are some other characters with the same radical sharing the similar pronunciations. It is also possible to look up the character in *Shuowen Jiezi*, as this dictionary first summarised the six categories and all phono-semantic characters are written in the similar format. But *Shuowen Jiezi* was written in the 1st or 2nd century. Most new characters are not included in this dictionary, so it is not a good source for looking up phono-semantic characters, but it is a good test dataset for any algorithm of identifying phono-semantic characters.

It is necessary to identify a method of identifying phono-semantic characters to understand how phono-semantic characters affect the result.

2.5.2 Methodology

A new method is proposed to identify the phono-semantic characters.

First, it is necessary to indentify all radicals of a given character to find the possible phonetic radical candidates.

Ideographic Description Sequence (IDS) is a method designed for presenting characters by the graphical components (i.e., radicals)[AAB$^+$12]. In IDS, an Ideographic Description Character (IDC) will be used to show the structure of the character. Table 2.9 shows all 12 IDCs, and Table 2.10 shows some examples of IDSs.

Type	IDC
Binary Operator	⿰⿱⿲⿳⿴⿵⿶⿷⿸⿹⿺
Trinary Operator	⿲⿳

Table 2.9: 12 Ideographic Description Characters in Unicode

There are only 12 IDCs in Unicode at the moment, which can be classified into two categories. Binary operators are used to show the position relationship of two combining parts, while trinary operators are used for three parts.

	Character	IDS
1	病	⿸疒丙
2	病	⿸疒⿱一⿻冂人
3	肝	⿰月干
4	胸	⿰月匈
5	匈	⿱勹凶
6	胸	⿰月⿱勹凶
7	徒	⿰彳走
8	徒	⿰彳⿱土止

Table 2.10: Examples of Ideographic Description Sequences

In Table 2.10, examples 1 & 2, 4 & 6 and 7 & 8 show that there is not an unique IDS for a character, because some radicals are also compounds of other radicals, examples 4, 5 and 6 show the process.

After breaking down the characters by IDS, all radicals of a character are obtained. In the previous introduction of phono-semantic characters, a phono-semantic character usually has a phonetic radical whose pronunciation is similar to the character. So the algorithm shown in Algorithm 1 should be able to identify the phono-semantic characters by finding phonetic radical. In the algorithm, IDS() is the function to get the IDS of a character; if the character cannot be divided anymore, the character itself will be returned. Similar() is a function to check if the pinyins of two characters/radicals are similar or not. The definition of similarity in pinyin will be discussed in detail below.

The algorithm performs a search among all the radicals of a character to find

Algorithm 1 Pseudo code of the algorithm for identifying the phono-semantic character

```
c is the character to be tested
ids = c
newids = ids
repeat
  ids = newids
  newids = ""
  for all i in ids do
    j = IDS(i)
    if Similar(c,j) then
      return TRUE
    end if
    newids+ = j
  end for
until newids == ids
return FALSE
```

if the pinyin of a radical is similar enough to the pronunciation of the character, because some phonetic radicals do not appear in the simplest IDS. For example, the phonetic radical of the character 徒 (tú) shown in example 7 and 8 in Table 2.10 is 土 (tǔ), which does not appear in the simplest IDS (i.e., example 7 in Table 2.10).

For the Similar() function, it is known that not all characters share exactly the same pinyin with their phonetic radicals, so it is not possible to define the function to return true only if the pinyin of two inputs are the same. One reasonable definition of the Similar() function could be, *return true if the finals of the pinyins of them are the same*. But this is not perfect. For example the Simplified Chinese character 脏 (zāng) in pair 6 in Table 2.7 slightly changed its final (i.e., ang) compared to the final (i.e., uang) of its phonetic radical 庄 (zhuāng).

To address this problem, the fuzzy tones should be considered when comparing the pinyin. Fuzzy tone is a feature of most Chinese pinyin input method, which aims to help people who have difficulty identifying similar pinyins, such

as initial *z* and *zh*, and final *an* and *ang*. In the use of fuzzy tones, some similar initials and finals will be treated same. For example, the pinyins of the characters 脏 (zāng), 张 (zhāng), 簪 (zān) and 占 (zhān) will be treated the same.

In the experiment, two different sets of fuzzy tones are used. The first one will ignore the prenuclear glides in the final, which is shown in Table 2.11; and the second one will further ignore the difference between front nasal finals and back nasal finals, which is shown in Table 2.12.

Group	Finals
1	a, ia, ua
2	e, ie, uo, ue
3	ai, iai, uai
4	ei, ui
5	ao, iao
6	ou, iu
7	an, ian, uan
8	en, in, un
9	ang, iang, uang
10	eng, ing, ueng
11	ong, iong

Table 2.11: Fuzzy tones set 1 which ignores the prenuclear glides

Group	Finals
1	a, ia, ua
2	e, ie, uo, ue
3	ai, iai, uai
4	ei, ui
5	ao, iao
6	ou, iu
7	an, ian, uan, ang, iang, uang
8	en, in, un, eng, ing, ueng
9	ong, iong

Table 2.12: Fuzzy tones set 2 which ignores the prenuclear glides and difference of nasal finals

So three different versions of the Similar() function will be tested:

1. Return true only if the finals of two inputs are the same

2. Return true when the finals of two inputs are same or they are in the same group in Table 2.11

3. Return true when the finals of two inputs are same or they are in the same group in Table 2.12

2.5.3 Experiment

Because there is not a method of identifying the phono-semantic characters, there is also no test data listing the phono-semantic characters and non-phono-semantic characters available now.

However, as mentioned before, the ancient *Shuowen Jiezi* summarised the six categories of Chinese characters. This dictionary identifies which of the six categories each character in the dictionary belongs to. The examples of how *Shuowen Jiezi* labels phono-semantic characters and non-phono-semantic characters are shown in Fig. 2.6.

1. 口：人所以言食也。象形。凡口之属皆从口。
2. 呼：外息也。从口乎声。
3. 吸：内息也。从口及声。
4. 君：尊也。从尹。发号，故从口。

Figure 2.6: Example of *Shuowen Jiezi*

从 (follow) is frequently used in *Shuowen Jiezi*; it is usually followed by another character, which suggests the meaning of the character it describes.

Sentence 1 describes the character 口 (mouth), which is the character before ：. And then 人所以言食也 is the meaning of 口 (mouth), explaining that 口 (mouth) is the thing that humans used for speaking and eating. After that, 象形 (pictogram) shows the category of the character 口 (mouth), which is pictogram.

As 口 (mouth) is also a primary radical, the final part 凡口之属皆从口 means that every character whose primary radical is 口 (mouth) has the meaning related to mouth.

Sentence 2 describes the character 呼 (exhale, hū). 外息也 explains the meaning of 呼 (exhale), which is breathe out. 从口乎声 means that its meaning is following 口 (mouth), and the pronunciation is following 乎 (an interrogative particle, hū), showing that 呼 (exhale) is a phono-semantic character by telling the semantic radical and phonetic radical.

Sentence 3 describes the character 吸 (inhale, xī). 内息也 explains the meaning of 吸 (inhale), which is breathe in. 从口及声 means that its meaning is following 口 (mouth), and the pronunciation is following 及 (reach, jī), showing that 吸 (inhale) is a phono-semantic character by telling the semantic radical and phonetic radical.

Sentence 4 describes the character 君 (ruler). 尊也 explains the meaning of 君 (ruler), which is someone respected. 从尹 means its meaning is following 尹 (officer). 发号，故从口 means that ruler needs to announce and give orders, and that is how its meaning related to mouth explaining the reason why the primary radical of 君 (ruler) is 口 (mouth). By using two 从 (follow), it shows that the meaning of 君 (ruler) is combined by two other characters, so that 君 (ruler) is a compound ideograph.

From the examples, it can be found that in the description of phono-semantic characters, there will always be a phrase in the format of 从 XY 声, where 从 means following, 声 means sound or pronunciation, X is the primary radical and Y is the phonetic radical.

Using this feature makes it easy to know whether each of the characters recorded in *Shuowen Jiezi* is phono-semantic or not. So that the test data could be obtained by viewing each character with its description in *Shuowen Jiezi*.

Shuowen Jiezi used in this experiment is taken from CJKVI Dictionary Database[2]. All characters are classified as phono-semantic characters or non-phono-semantic characters based on their descriptions, and 7710 characters are phono-semantic characters and the other 1716 characters are non-phono-semantic characters.

The IDS data are taken from the CHISE project[3]. The pinyin data are taken from the Unihan database provided by Unicode, Inc.[4]. When the character is a heteronym with different pinyins, the most common one will be used.

2.5.4 Result

The results are shown in Table 2.13 and Table 2.14. The result is shown in the form of precision, recall and F-measure, which can be calculated by the following the equations shown in Formulas 2.1, 2.2 and 2.3.

$$precision = \frac{true\_positive}{true\_positive + false\_positive} \tag{2.1}$$

$$recall = \frac{true\_positive}{true\_positive + false\_negative} \tag{2.2}$$

$$F - measure = \frac{2 * precision * recall}{precision + recall} \tag{2.3}$$

It can be easily seen that all three methods have a high precision and a slightly lower recall. One reason for this is that the data is unbalanced; the phono-semantic characters are about four times more common than the non-phono-semantic characters. The other reason is that even using fuzzy tone set 2, it will still not cover the case shown in Table 2.3, where the finals of characters

[2] http://kanji-database.sourceforge.net/

[3] http://www.chise.org/

[4] http://www.unicode.org/charts/unihan.html

	Real phono-semantic	Real non-phono-semantic
Predicted phono-semantic	4274	413
Predicted non-phono-semantic	2415	2324

(a) Without fuzzy tone

	Real phono-semantic	Real non-phono-semantic
Predicted phono-semantic	4785	502
Predicted non-phono-semantic	1904	2235

(b) Fuzzy tone set 1

	Real phono-semantic	Real non-phono-semantic
Predicted phono-semantic	4833	524
Predicted non-phono-semantic	1856	2213

(c) Fuzzy tone 2

Table 2.13: Result of identifying phono-semantic characters in *Shuowen Jiezi*

Method	Without fuzzy tone	Fuzzy tone set 1	Fuzzy tone set 2
Precision	0.912	0.905	0.902
Recall	0.639	0.715	0.723
F-measure	0.751	0.799	0.802

Table 2.14: F-measure of different methods

and phonetic radicals are completely different.

Comparing the method using fuzzy tones with the method without fuzzy tones, it can be seen that when the fuzzier tones are used, the result will have a lower precision and a higher recall. It is because the fuzzy tones will classify some non-phono-semantic characters as phono-semantic characters.

From Table 2.14, it can be found that the method using fuzzy tone set 2 performs best, so it should be used in the subsequent experiments when it is necessary to check if a character is phono-semantic.

As *Shuowen Jiezi* is an ancient dictionary, so the characters in it are all in Traditional Chinese. It is necessary to compare the result when converting the characters into Simplified Chinese. The tool used to convert Traditional Chinese into Simplified Chinese is the *Simplified and Traditional Chinese Intelligence Converting System* provided by Xiamen University.

Three methods are applied to the data of *Shouwen Jiezi* in Simplified Chinese, the results are shown in Table 2.15.

Method	Without Fuzzy tone	Fuzzy tone set 1	Fuzzy tone set 2
Precision	0.909	0.903	0.900
Recall	0.623	0.700	0.707
F-measure	0.739	0.789	0.792

Table 2.15: Result of identifying phono-semantic characters in Simplified Chinese

Compared to the original Traditional Chinese version, the performance in Simplified Chinese data is slightly worse. It may be explained by the fact that the original data is in Traditional Chinese, and some phono-semantic characters will become non-phono-semantic characters after conversion, and vice versa. The results in Simplified Chinese are still acceptable with a highest F-measure at 0.792, which means the proposed method could be used on both Simplified Chinese and Traditional Chinese. Similar to the results in Traditional Chinese, the method using fuzzy tones set 2 performs best, which is additional evidence in supporting of using this method in the further experiment.

2.5.5 Conclusion

This section introduces a new problem about how to identify a phono-semantic character. A new algorithm is proposed based on the definition of the phono-semantic characters. In the algorithm, all the radicals of the given character are checked whether they have a similar pinyin with the given character. If the final of the radical and that of the given character are in the same fuzzy tones group, the algorithm will match the pinyin of the radical and that of the given character as similar. The algorithm is tested on *Shuowen Jiezi*, the first known Chinese dictionary annotated the categories of the characters, and the algorithm shows a high F-measure at 0.792.

2.6 Summary

This chapter introduces the graphical features (primary radical and phonetic radical) and the pronunciation indicating systems (pinyin, zhuyin and fanqie) of Chinese characters. Three indicating systems are compared and the reasons for selecting pinyin are stated. Previous studies about using radicals and pinyin have been reviewed and analysed. The limitations of these works have been discussed, and new experiment have been proposed to overcome the limitations. Problems of processing with radicals are discussed and one method of identifying phono-semantic characters is proposed and tested using *Shuowen Jiezi* data; it achieve a F-measure at 0.802.

Chapter 3

Named Entity Recognition

3.1 Previous Work

Named Entity Recognition (NER) is a task to find the named entities in the given text. The named entities are usually the names of persons, places and other specified objects. In the biomedical domain, the named entities usually refer to diseases, symptoms and so on.

An NER task is usually regarded as a sequence tagging task where each word will be assigned a label. A typical tagging format BIO is shown in Table 3.1, where **B-X** stands for the beginning of a named entity whose type is **X**, **I-X** means that this word is inside a named entity with type **X**, and **O** presents outside named entities.

In the early time, some rule-based methods are applied to the NER task. In rule-based methods, a set complex rules of are used to identify the named entities. For example, if a person's title such as *Mr.* is found, then the following word with the first letter capitalised is supposed to be one named entity of person name. LaSIE-II[HGA+98] and IsoQuest[KH98] are two examples of rule-based system. Rule-based methods usually require complex preprocessing in a natural

Word	Tag
Donald	B-Person
Trump	I-Person
and	O
Xi	B-Person
Jinping	I-Person
reached	O
agreement	O
at	O
the	O
G20	B-Organisation
summit	O
in	O
Japan	B-Location

Table 3.1: BIO tagging of sentence *Donald Trump and Xi Jinping reached agreement at the G20 summit in Japan.*

language, such as POS tagging and syntax analysis. The setup of the rules usually requires much time and the involvement of experts in the domain, which makes the method less effective on a different domain.

3.1.1 Basic Machine Learning

After rule-based methods, basic machine learning methods are then used for the NER tasks. The most common models used include Support-vector Machine (SVM), Hidden Markov Model (HMM) and Conditional Random Field (CRF).

Support-vector Machine

Support-vector Machine is a model advanced by Corinna et al.[CV95], which is used to solve classification problems. SVM will construct an binary classifier and classify a given vector x into 1 or -1 by Formula 3.1. In the formula, $w \cdot x + b = 0$ is the hyper-plane used to separate the data. So the output of a given vector is decoded by the side of the hyper-plane that the data will be located.

$$c(x) = \begin{cases} 1 & \text{if } w \cdot x + b > 0, \\ -1 & \text{otherwise} \end{cases} \tag{3.1}$$

When given the training set $[(x_1, y_1), (x_2, y_2), ...(x_N, y_N)]$, where x_i is the data vector and y_i is the classification (i.e. 1 or -1), SVM will try to find a hyper-plane that will maximise the margin. Margin is defined as the distance between hyper-plane and the nearest data vectors, and those vectors are the support vectors. By maximising the margin, it will make sure that the hyper-plane is located at the midpoint between two different classes of data.

The training of SVM can be expressed as a mathematical problem to minimise $\frac{1}{2}\|w\|^2$ subject to $y_i(w \cdot x_i + b) \geq 1, i = 1, 2, ..., N$.

Considering that in most cases, there is not a perfect hyper-plane to separate the data with no errors, a soft margin is often used. It can be express by a non-negative variable ξ. And the problem becomes minimising $\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \xi_i$ subject to $y_i(w \cdot x_i + b) \geq 1 - \xi_i, i = 1, 2, ..., n$, where C is a constant.

In the use of NER, words are required to be classified into more than two classes, so a multi-class classifier is required. To build a multi-class classifier based on SVM, two different methods are proposed[KMOT02] :

- **One-vs-rest** In the one-vs-rest approach, assuming there are K classes, K different SVM classifiers will be constructed. Each of the classifier is used to decide if the vector should be classified into this class or not.

- **Pairwise** In the pairwise approach, assuming there are K classes, $K(K-1)/2$ different SVM classifiers will be constructed. Each of the classifier is used to vote if the vector should be in class i or class j. $K(K-1)/2$ classifiers will cover all possible pairs of two classes (i, j). The final class of the vector will be the class that most voted.

In the SVM method, a set of features of a word is required to make a word become a high-dimensional data point for classification. Features of a word include the part-of-speech of the word, whether special symbols are included, whether the first letter is capitalised, and whether the previous word has its first letter capitalised. Then NER task becomes a classification problem to classify these words into different classes, each of which represent one tag[JWZ11].

Hidden Markov Model

Hidden Markov Model[BJM83][LRS83] is a directed graphical model defined as a 5-tuple (S, V, Π, A, B), representing the set of possible hidden states, the set of possible observations, the initial probabilities, the state transition probabilities and the omitting probabilities, respectively[TSL87].

One of the use of the HMM is to predict the hidden states sequence of a given observations sequence. In the use of NER task, with a given sequence of tokens $G_1^n = g_1 g_2 ... g_n$, the HMM will try to find a corresponding sequence of tags $T_1^n = t_1 t_2 ... t_n$, which maximise $P(T_1^n | O_1^n)$[MCF+98]. In the sequence of tokens, each token t_i is defined as (f_i, w_i), where w_i is the word and f_i is the features of the word w_i. Each tag t_i has three parts: the boundary category, the entity class and the feature set. Boundary category shows if the word is at the beginning, middle or the end of a named entity, which is the same as B and I in BIO tagging. Entity classes are the names of classes with an extra NOT-NAME class. The feature set is used to enhance the accuracy of the model because there are only few combinations of the boundary category and the entity class[ZSZ+04].

In the model, the maximising of $P(T_1^n | O_1^n)$ is often represented by maximising[ZS02]:

$$\log P(T_1^n | G_1^n) = \log P(T_1^n) + \log \frac{P(T_1^n, G_1^n)}{P(T_1^n) \cdot P(G_1^n)} \tag{3.2}$$

And by assuming the independence of T_1^n and G_1^n, the formula can be written as[ZS02]:

$$\log P(T_1^n|G_1^n) = \log P(T_1^n) - \sum_{i=1}^{n} \log P(t_i) + \sum_{i=1}^{n} \log P(t_i|O_1^n) \tag{3.3}$$

In the formula, the first item can be computed by applying chain rules, each tag is assumed to be independent on the previous tags. The second item is the sum of log probabilities of all tags. The third item can be computed by a two-level back-off model, where the first level will be based on the word features and words themselves, and the second one will be based on different combinations of word sub-features[ZS02].

In the use of HMM in NER task, word features can be capitalisation, digitalisation and important prefix or suffix.

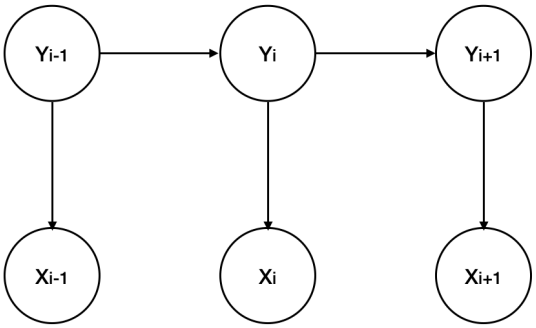
Conditional Random Field

Conditional Random Field is a sequence modelling framework that can be presented as a undirected graphical model. Let $G = (V,E)$ be the graph, and Y_v is the vertex of the graph, X is the condition. Then (X,Y) is a conditional random field, when $P(Y_v|X,Y_w) = P(Y_v|X,Y_w)$, where $w \neq v$ and Y_w and Y_v are the neighbouring vertices in graph G[LMP01].

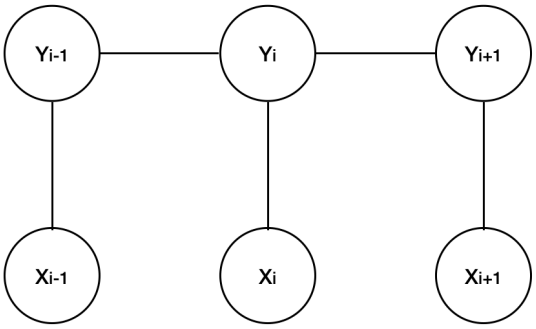
CRF can be considered as an indirected HMM shown in Figure. 3.1.

Based on the work of Lafferty et al.[LMP01], the probability of a tag sequence $y = y_1 y_2 ... y_n$ when given the word sequence $x = x_1 x_2 ... x_n$ in CRF can be calculated as:

$$P(y|x) = \frac{1}{Z(x)} \exp \sum_{i,k} \lambda_k f_k(y_{i-1}, y_i, x) + \sum_{i,k} \lambda_{k'} f_{k'}(y_i, x) \tag{3.4}$$



(a) Hidden Markov Model



(b) Conditional Random Field

Figure 3.1: Conditional Random Field and Hidden Markov Model

In the formula, $Z(x)$ is the normalisation constant, k and k' are the number of features of edges and nodes, respectively. f_i is the feature function and λ_i is the learning weight for each function.

When using CRF for tagging, the tagging sequence will be calculated by $arg\,max_y P(y|x)$.

In CRF, the observation will be considered globally, which means the context will influence the prediction. Similarly, the input is not only the words themselves; some other word features are also included.

Chinese Processing

All these machine learning methods require the annotated corpus with all named entities labelled to train the model. It should be also noticed that in the

methods, words are usually presented by different features. The features could be external features about the context or the internal features about the word itself.

In Chinese processing, different word features are applied. In the work of Chen et al.[CZI06] of recognising person names, places names, and organisation names, the list is generated first for the first characters for person names, the characters in person names, the prefix of person names, the suffix of person names, the characters in place names, the suffix of place names, the characters in organisation names, the suffix of organisation names. By matching the corresponding character in the test text, the feature is generated.

In the other works of Chinese NER using basic machine learning methods, the features used are similar, the n-gram characters before or after the current character are used. Some works like the work of Feng et al.[FSZ05] uses part-of-speech as an extra feature, and proofs that it can improve the performance. Some other works like the work of Han et al.[HWC13] tries to find a suitable window size for context characters.

However, in the methods of using basic machine learning methods, radical features or other graphical features of Chinese characters are ignored. The use of radical features in the basic machine learning method of Chinese Named Entity Recognition will be tested.

3.1.2 Deep Learning

Recently, deep learning methods become the trend of solving NER task. Figure 3.2 shows the traditional format of the deep learning approach. In the deep learning approach, the words are presented as word embeddings, and the look-up layer will convert the words into the corresponding embeddings by looking

at the pretrained word embeddings. Finally, the word embeddings are fed into the neural network model and the result is obtained.

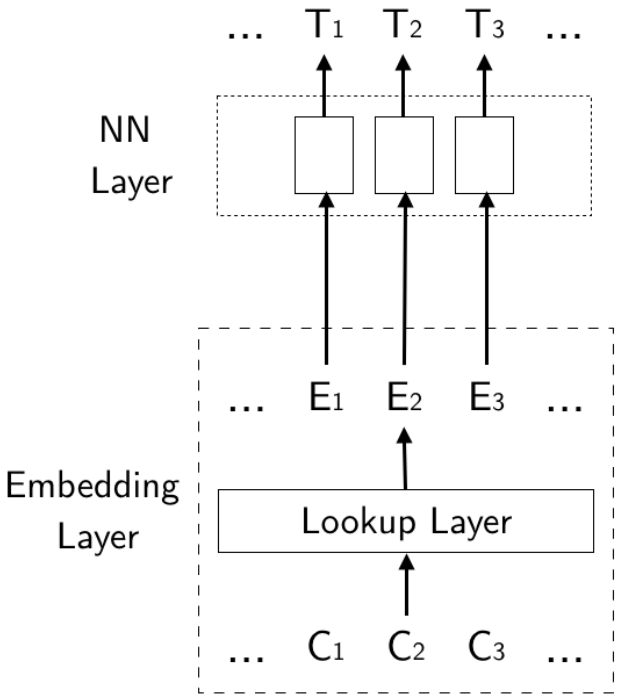


Figure 3.2: Neural Network Structure of Named Entity Recognition

Long Short-Term Memory

Long Short-Term Memory (LSTM) is one of the possible neural network models. It is one kind of Recurrent Neural Network (RNN), which aims to use the previous data on current processing.

The structure of a cell in LSTM is shown in Fig. 3.3, where the information about the previous data is presented as C_{t-1} and h_{t-i}, and X_t is the current input. First, part of the information in C_{t-1} is reset to 0 after the production labelled with F. Then new information about the current input is added to C_{t-1} in the addition labelled with I to update it as C_t. The production labelled with O takes the useful information in C_{t-i} and the concatenation of h_{t-i} with X_t to decide

the output Y_t and the information passed to next cell h_t. Both C_i and h_i include the information about the previous data, while h_i changes rapidly, which can be regarded as the short term memory, and C_i changes slowly, which can be regarded as the long term memory[GSC99].

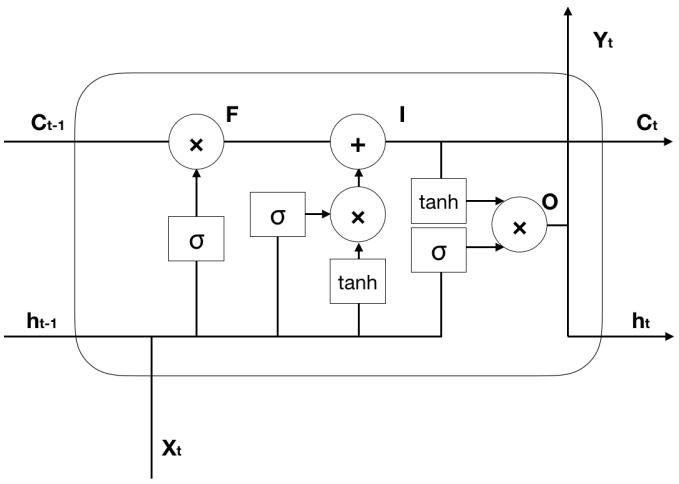


Figure 3.3: Structure of LSTM cell

Bidirectional Long Short-Term Memory (BiLSTM) is the advanced version of LSTM, which does not only use the previous information but also the future information. The structure of BiLSTM is shown in Fig. 3.4, where the data has been processed in both forward and backward directions, and the output is the concatenation of the results of two processes.

Another improvement is to add a CRF layer before outputting the result, like shown in Fig.3.5, which is called the BiLSTM-CRF model.

The BiLSTM-CRF model works very well on the NER task, because the long term memory used can deal with the long-dependency problem in natural language, short term memory with bidirectional processing can be used to deal with the context nearby, and the CRF layer is very effective at converting the result of BiLSTM, which is usually a vector, into different classes. There are

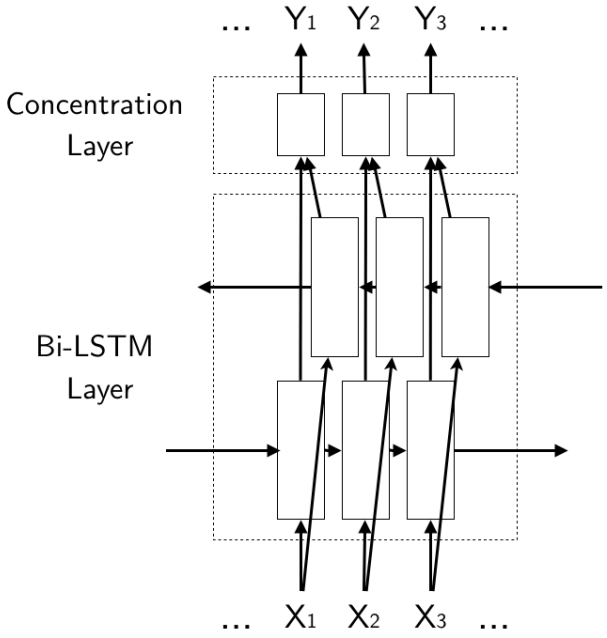


Figure 3.4: Structure of BiLSTM model

many studies about using BiLSTM-CRF in an NER task: Huang et al. first used BiLSTM-CRF on NER tasks on CONLL dataset[HXY15], Ma et al. used an advanced version to capture character information in English by adding another char embedding[MH16], and Reimers et al. discussed how to get the optimal hyperparameters in such task[RG17].

Word Embedding

Words in the deep learning approaches are usually in embedding forms. Word embedding is the task to convert words into vectors for further processes in neural network. The easiest approach is one-hot representation. One-hot representation is to use one-dimensional vector filled with only 0 and 1 to represent the word. For example, when applying one-hot representation to three words A, B and C, A will be $[1,0,0]$, B will be $[0,1,0]$, and C will be $[0,0,1]$[TRB10]. This method is

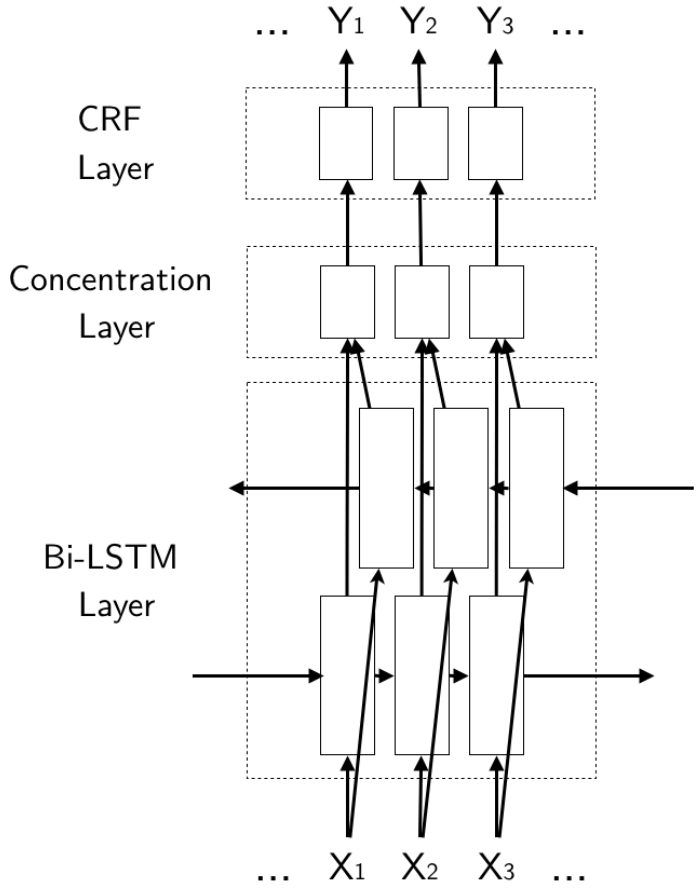
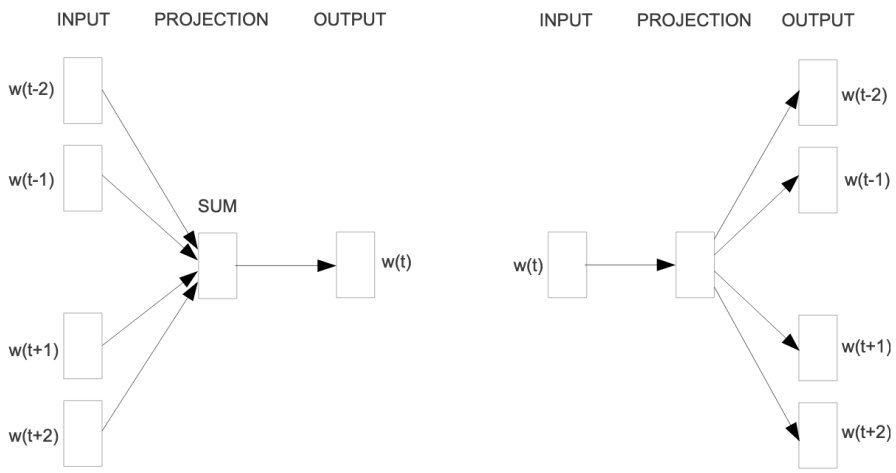


Figure 3.5: Structure of BiLSTM-CRF model

very easy to implement but the performance is not good, because this embedding method will not show the relationship between words, and when size of vocabulary is large, the size of the vectors will be too large to process.

To avoid the problems of one-hot representation, distributed representation is introduced, where the embedding is trained by some given text and fit into a smaller vector compared to one-hot representation. Continuous Bag-of-Words (CBOW) and Skip-gram are two basic models for distributed representation. During the training process of CBOW, the words before and after the current words will be used as input to predict the current word, while in the Skip-gram model, the current word is used for predicting the words before and after. The



(a) CBOW (b) Skip-gram

Figure 3.6: Structure of basic word embedding model

structures of the models are shown in Fig. 3.6 [MCCD13].

These two basic models will only use the word for training. To use the subword features (i.e., internal features mentioned before) in the training of word embedding, several methods have been attempted. In the research of Luong et al., the words were split into several subwords, which are usually prefixes, suffixes and word roots, and the embedding of each subword will be composed to get the embedding of the word[LSM13]. The work of Bojanowski et al. uses an n-gram as the subword and trains the embedding for subwords[BGJM16].

Besides using subword features, another attempt to using an internal feature is to train the embedding on character-level, and to get the character embedding. There are only 26 letters in English, which means that the amount of trained embedding is small and there is no need to deal with the out-of-vocabulary words, as all English words are made of the 26 letters with a limited amount of other symbols. Character embedding is used by Zhang et al. and tested on text classification problems[ZZL15].

Chinese Embedding

In word embedding in Chinese, some models try to use the graphical features including radicals.

JWE

Joint learning word embedding model (JWE) is an embedding model proposed by Yu et al.[YJXS17] that aims to build the embedding from the information of the context words, characters and primary radicals.

The structure of JWE is shown in Figure 3.7. In the figure, w_i is the target word, w_{i-1} and w_{i+1} represents word before and word after, respectively. Similarly, c_{i-1} and c_{i+1} represents the character before and character after, respectively, and s_{i-1} and s_{i+1} represents the subcharacters before and after, respctively. s_i is the subcharacter of the current word w_i.

It is actually a modified version of the CBOW model, which uses the average of context word embedding, that of context character embedding and that of context radical embedding, to predict the word.

The subcharacters used in this model are the radicals. For example, if w_i is 氵汉氵工 (Han River), then s_i will be 氵又氵工, where 氵 and 又 are the radicals of 氵汉, and 氵 and 工 are the radicals of 氵工.

The subcharacters in this model are obtained from Xinhua Dictionary online[1].

cw2vec

Chinese Word to Vector (cw2vec) is an embedding model proposed by Cao el al.[CLZL18], which uses another graphical feature of Chinese character named

[1] `http://tool.httpcn.com/zi/`

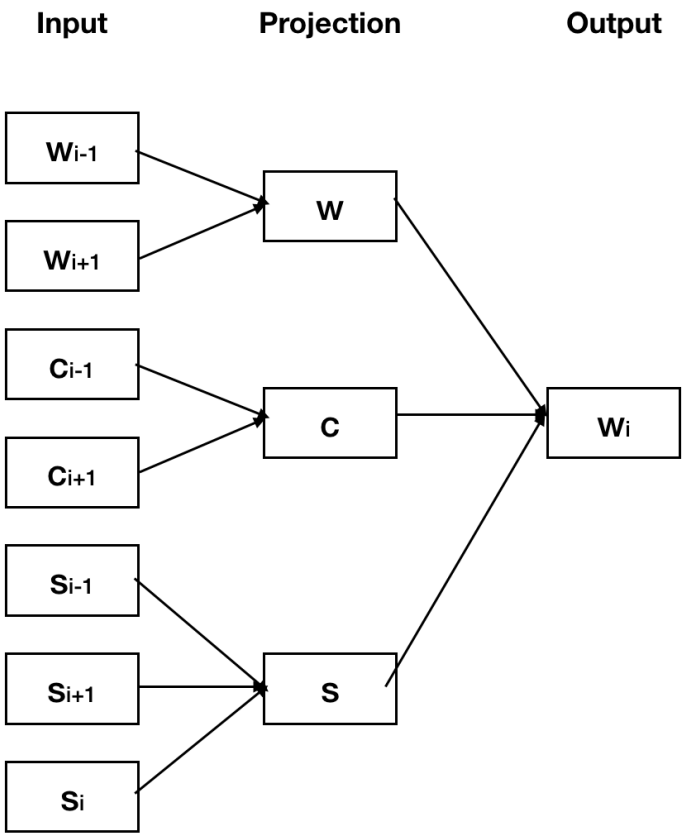


Figure 3.7: Structure of Joint Learning Word Embedding Model

stroke. Stroke is the smallest graphical component of Chinese characters, which can be defined as the mark written without lifting the pen from the paper.

In the cw2vec model, all strokes are classified into five types shown in Table 3.2, only few examples are shown in each type.

ID	Stroke Name	Example
1	横 (horizontal)	一ノ
2	竖 (vertical)	丨乚
3	撇 (left-failing)	丿
4	捺 (right-failing)	乀丶
5	折 (turning)	𠃌𠃍𠄌𠃋

Table 3.2: Five types of strokes in Chinese characters

The structure of the cw2vec model is shown in Figure 3.8. In this example, the input word is 雾霾 (haze), and based on the IDs in Table 3.2, the word can

be transferred to the sequence of IDs of their strokes:

1452444435435145244434432332511211

Based on this sequence, n-grams of strokes can be formed including 3-grams, 4-grams and 5-grams. By using the similar idea with Bojanowski et al.[BGJM16], the word embedding is finally trained for predicting the context words like in Skip-gram model.

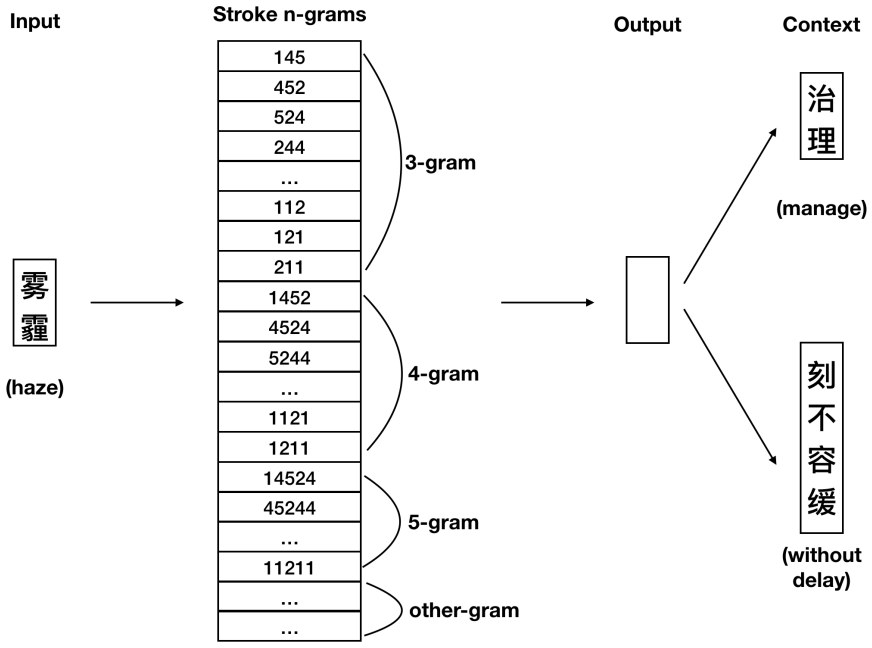


Figure 3.8: Structure of cw2vec model

Glyce

Glyph-vectors for Chinese Character Embedding (Glyce) is an embedding model proposed by Wu et al.[WMH+19]. Unlike other models using the features of characters, Glyce will use the images of the characters for building embeddings.

Tianzige (田字格, tian-shape squares) is the square that primary school students in China used to write characters in. It is split into 4 different squares just like the character 田 (field, tián), and that is how its name comes from.

Glyce uses the idea of Tianzige, and build the word embedding like shown in Figure 3.9.

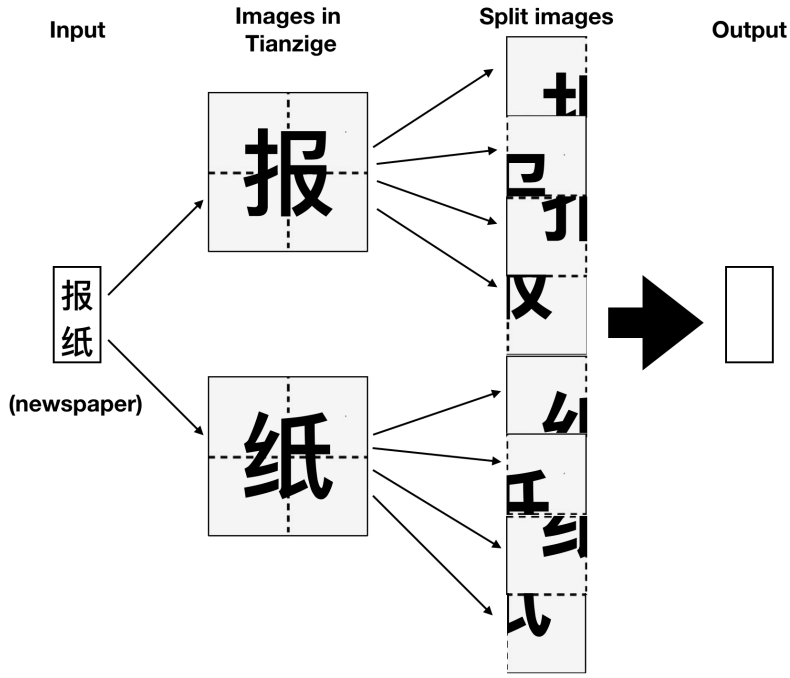


Figure 3.9: Structure of Glyce model

In Figure 3.9, 报纸 (newspaper) is the input word, and then it will be transferred to images of character 报 and 纸. Then both image of characters can be split into 4 images, and the total 8 images are used to train the word embedding of the word 报纸.

Recent works

The three models mentioned above use different graphical features in embedding: all radicals for JWE, the strokes sequence for cw2vec, and the images of the characters. These three features are also the focuses in the recent works on graphical features.

The work of Xiong et al.[XQYL20] tries to improve the idea using strokes. The representation of the Chinese characters using strokes sequence may cause

some ambiguities, for example, the strokes sequences of the character 冷 (cold) and 这 (this) are completely same although their meanings are very different. In order to solve the problem, Xiong et al. use the character with its stroke sequences for character embedding.

Glyph to Vector (Glyph2Vec) proposed by Chen et al [CYL20] is a model using the images of the characters. Unlike Glyce, which splits a character into four parts, Glyph2Vec uses the character and its whole image to form the embedding.

Radical and Stroke-enhanced Word Embeddings (RSWE) is the model proposed by Wang et al.[WZZ20], this model combines the idea of using all radicals and the stroke sequences. In RWSE, the word embeddings are formed by the radicals and the stroke sequences of the characters.

Limitation

These methods have a common issue. JWE uses all radicals, cw2vec uses all strokes, Glyce uses the images of characters, and other recent research uses the same features. They rely on the hypothesis that all parts of all the Chinese characters are useful and suggest the meanings of the characters. However, based on the review and experiment in the last chapter, the hypothesis is not true, because most of the Chinese characters are phono-semantic characters, and in phono-semantic characters, only the primary radicals contain the semantic information. So the combination of the use of graphical features with their pronunciation should perfectly represent the form of phono-semantic character, where the primary radical is the semantic part and pronunciation represents the phonetic radical.

Another problem of these model is about the standard.

In the methods using all radicals, like JWE, it gains all the radicals from Xinhua Dictionary Online, however, there is only an official mobile app of *Xinhua*

Dictionary, and in the mobile app, there is no such feature to get all the radicals of a given character. Even if there is an official resource of getting all radicals, the same problem of using IDS in the last chapter is raised. Characters may have different representations of radicals because some of the radicals are formed by other radicals as well. So using all radicals should deal with the problem about choosing the suitable radical representation of the characters.

In the methods using strokes, like cw2vec, strokes are used and classified in 5 classes. However, some strokes are actually difficult to be classified. For example, 丿 can be considered as both horizontal and right-failing. The standard of the classification should be carefully established.

In the methods using images of the characters , like Glyce, another problem is faced. There are different fonts in Chinese, the characters will look different in different fonts, and the images of these characters in Tianzige are different as well. When using the historic scripts of characters, the thing will become more complicated. In different files or books, the same character will look differently because all the characters are hand-writing. It is more difficult to choose a standard for the images of the characters.

Until now, there is no work on using the combination of the radical features and the phonetic features, which represent the structure of the phono-semantic characters, which is the majority of the Chinese characters.

The following part of this chapter will focus on using the radical features and the phonetic features in the basic machine learning method and in the deep learning method because the rule-based method is a little outdated and writing rules usually requires the involvement of the experts.

3.2 Basic Machine Learning

3.2.1 Methodology

To capture the feature of phono-semantic characters, both primary radical and phonetic radical should be used. However, as said before, not all Chinese characters are phono-semantic characters, so not all Chinese characters have phonetic radicals. Besides, it is also a difficult task to find the phonetic radical of a given phono-semantic character.

Pinyin will be a good alternative feature for a phonetic radical, because both pinyin and phonetic radicals are used to help people understand the pronunciation of the character, and the pinyins of most phono-semantic characters are similar to that of their phonetic radicals.

In the method of identifying phono-semantic character used in the last chapter, the property that most phono-semantic characters share the same final with their phonetic radical is used. So based on the same property, in the alternative use of pinyin, only the final of the pinyin should be used. However, whether only the final is useful or all of the pinyin is useful should be determined by testing.

The use of radical features on basic machine learning methods is easy. As stated in the last section, it is always necessary to present a word in the form of a mixture of word, internal features and external features, and the radical features can be used as internal features of a word.

CRF will be the model used in the test here, and primary radical, pinyin and final will be the additional internal features. Four different tests will be run. The first will only use the character as the internal feature, the second will use both word and primary radical, the third will use word, primary radical and pinyin, and the fourth will use word, primary radical and final. When an

additional internal feature is used, the context feature about that feature will be used as additional external feature as well. For example, the primary radical of the character before and after will be used as the external feature. The detail of the feature used is shown in Table 3.3. In the table, c_i is the target word. c_{i-1} and c_{i-2} is the character before and the character before the character before, respectively. c_{i+1} and c_{i+2} is the character after and the character after the character after, respectively. r_i, p_i and f_i is the primary radical, the pinyin and the final of the target character, respectively.

In each group of the features, the uni-gram, bi-gram and trig-ram within a window with size at 5 will be considered.

The method using CRF usually includes the using of part-of-speech feature; however, different tools will give different part-of-speech tagging results. To avoid that problem and eliminate the difference when using different tools, the part-of-speech will not be included in the experiment.

Four different groups of features are obtained, the experiment will be made on the different combinations of the features.

Six different set of experiment will be made. The first will use character features only for comparison with the other set. The second will use both character and primary radical. The third will use both character and pinyin. The fourth will use character and final. The fifth will use character, primary radical and pinyin. The sixth will use character, primary radical and final. The pair of the third set and the fourth set and the pair of the fifth set and the sixth set are used to verify whether only the final of the pinyin is useful or all the parts of pinyin are necessary.

Similarly to the last chapter, all pinyin and primary radical information of characters has been taken from Unihan database.

Group	Features
Character	1. c_i
	2. c_{i-2}
	3. c_{i-1}
	4. c_{i+1}
	5. c_{i+2}
	6. $c_{i-1}c_i$
	7. c_ic_{i+1}
	8. $c_{i-2}c_{i-1}$
	9. $c_{i+1}c_{i+2}$
	10. $c_{i-2}c_{i-1}c_i$
	11. $c_ic_{i+1}c_{i+2}$
Primary Radical	1. r_i
	2. r_{i-2}
	3. r_{i-1}
	4. r_{i+1}
	5. r_{i+2}
	6. $r_{i-1}r_i$
	7. r_ir_{i+1}
	8. $r_{i-2}r_{i-1}$
	9. $r_{i+1}r_{i+2}$
	10. $r_{i-2}r_{i-1}r_i$
	11. $r_ir_{i+1}r_{i+2}$
Pinyin	1. p_i
	2. p_{i-2}
	3. p_{i-1}
	4. p_{i+1}
	5. p_{i+2}
	6. $p_{i-1}p_i$
	7. p_ip_{i+1}
	8. $p_{i-2}p_{i-1}$
	9. $p_{i+1}p_{i+2}$
	10. $p_{i-2}p_{i-1}p_i$
	11. $p_ip_{i+1}p_{i+2}$
Final	1. f_i
	2. f_{i-2}
	3. f_{i-1}
	4. f_{i+1}
	5. f_{i+2}
	6. $f_{i-1}f_i$
	7. f_if_{i+1}
	8. $f_{i-2}f_{i-1}$
	9. $f_{i+1}f_{i+2}$
	10. $f_{i-2}f_{i-1}f_i$
	11. $f_if_{i+1}f_{i+2}$

Table 3.3: Features used in Conditional Random Field

The CRF++ toolkit[2] will be used in the experiment.

The data used in the experiment were provided in the China Conference on Knowledge Graph and Semantic Computing (CCKS) in 2017, which collects different clinical texts and contains 280,913 characters. The corpus uses BIO format to label five different named entity types: body part (BOD), symptom (SYM), disease (DIS), experiment (EXP) and treatment (TRE). A 5-cross-validation is performed to make the experiment.

3.2.2 Result

The results of the six sets of experiments on CCKS data are shown in Table 3.4, where the letters in the 1st row show the features used. C represents character, R represents primary radical, P represents pinyin, and F represents final.

Features	C	CP	CF	CR	CRP	CRF
BOD	0.567	0.568	0.565	0.564	0.563	0.564
SYM	0.609	0.608	0.606	0.605	0.605	0.610
DIS	0.583	0.567	0.550	0.559	0.568	0.555
EXP	0.569	0.571	0.571	0.563	0.566	0.566
TRE	0.625	0.634	0.654	0.657	0.614	0.613
ALL	**0.589**	0.576	0.586	0.584	0.585	0.586

Table 3.4: F-measures of CRF experiment on Simplified Chinese

It can be seen that the use of the primary radical and the pinyin does not improve the performance of CRF in NER task. Using only the character feature has the highest F-measure, and the second best is the set using character and final and the set using character, primary radical and final, which has a F-measure only 0.002 less than the best. It may because the other features including primary radical, pinyin and final cannot contribute to the final result.

[2] `https://taku910.github.io/crfpp/`

However, in the result tagged by different group of features, primary radical, pinyin and final work better than characters only in some cases, shown in Table 3.5, which is the tags of the sentence 门诊以哮喘性-支气管炎收入院 (*Clinic treated the patient as Acute bronchitis*) by different group of features.

Character	True Tag	C	CP	CF	CR	CRP	CRF
门	O	O	O	O	O	O	O
诊	O	O	O	O	O	O	O
以	O	O	O	O	O	O	O
哮	B-DIS	O	B-DIS	B-DIS	O	B-DIS	B-DIS
喘	I-DIS	O	I-DIS	I-DIS	O	I-DIS	I-DIS
性	I-DIS	O	I-DIS	I-DIS	O	I-DIS	I-DIS
-	I-DIS	O	I-DIS	I-DIS	O	I-DIS	I-DIS
支	I-DIS	B-BOD	I-DIS	I-DIS	B-BOD	I-DIS	I-DIS
气	I-DIS	I-BOD	I-DIS	I-DIS	I-BOD	I-DIS	I-DIS
管	I-DIS	I-BOD	I-DIS	I-DIS	I-BOD	I-DIS	I-DIS
炎	I-DIS	O	I-DIS	I-DIS	O	I-DIS	I-DIS
收	O	O	O	O	O	O	O
入	O	O	O	O	O	O	O
院	O	O	O	O	O	O	O

Table 3.5: Different tags for 门诊以哮喘性-支气管炎收入院 (*Clinic treated the patient as asthmatic bronchitis*)

In this sentence, only the model that pinyin or final is involved (i.e. CP, CF, CRP and CRF) correctly tag the named entity 哮喘性-支气管炎 (asthmatic bronchitis), the other models treated 支气管 (bronchus) as a named entity and not realise that it is also part of another named entity.

Another example of tagging is shown in Table 3.6, which is the tags of the sentence 主因发热 4 天，抽搐 1 次，咳嗽 2 天 (*The main reason is fever for 4 days, twitch once and cough for 2 days*).

In this case, the models having primary radicals involved (i.e. CR, CRP and CRF) correctly find the named entity 发热 (fever) while other models fail to do so.

The examples shown above suggest that in some cases, features including

Character	True Tag	C	CP	CF	CR	CRP	CRF
主	O	O	O	O	O	O	O
因	O	O	O	O	O	O	O
发	B-SYM	O	O	O	B-SYM	B-SYM	B-SYM
热	I-SYM	O	O	O	I-SYM	I-SYM	I-SYM
4	O	O	O	O	O	O	O
天	O	O	O	O	O	O	O
，	O	O	O	O	O	O	O
抽	B-SYM	B-SYM	B-SYM	B-SYM	B-SYM	B-SYM	B-SYM
搐	I-SYM	I-SYM	I-SYM	I-SYM	I-SYM	I-SYM	I-SYM
1	O	O	O	O	O	O	O
次	O	O	O	O	O	O	O
，	O	O	O	O	O	O	O
咳	B-SYM	B-SYM	B-SYM	B-SYM	B-SYM	B-SYM	B-SYM
嗽	I-SYM	I-SYM	I-SYM	I-SYM	I-SYM	I-SYM	I-SYM
2	O	O	O	O	O	O	O
天	O	O	O	O	O	O	O

Table 3.6: Different tags for 主因发热 4 天，抽搐 1 次，咳嗽 2 天 (*The main reason is fever for 4 days, twitch once and cough for 2 days*)

primary radical, pinyin and final are useful, although the overall performance of using these features are not the best.

It should be also noted that the models using final have the higher F-measure than the one using pinyin. For example, the F-measure of model CF is 0.01 higher than that of model CP. It proves that final is a better alternative feature of the phonetic radical than the pinyin.

It can also proof that final is better than pinyin in the example shown in Table 3.7, which is the tags of the sentence 主因鼻塞 3 天，头晕 2 天，伴呕吐 5 次 (*The main reason is nasal congestion for 3 days, dizziness for 2 days with vomit 5 times*).

In this example, models having final involved (i.e. CF and CRF) show better performance by correctly tagging the named entity 呕吐 (vomit), while all other models fail to tag it.

As said before, the simplification of Chinese characters changes the primary

Character	True Tag	C	CP	CF	CR	CRP	CRF
主	O	O	O	O	O	O	O
因	O	O	O	O	O	O	O
鼻	B-SYM	O	O	O	O	O	O
塞	I-SYM	O	O	O	O	O	O
3	O	O	O	O	O	O	O
天	O	O	O	O	O	O	O
，	O	O	O	O	O	O	O
头	B-SYM	B-SYM	B-SYM	B-SYM	B-SYM	B-SYM	B-SYM
晕	I-SYM	I-SYM	I-SYM	I-SYM	I-SYM	I-SYM	I-SYM
2	O	O	O	O	O	O	O
天	O	O	O	O	O	O	O
，	O	O	O	O	O	O	O
伴	O	O	O	O	O	O	O
呕	B-SYM	O	O	B-SYM	O	O	B-SYM
吐	I-SYM	O	O	I-SYM	O	O	B-SYM
5	O	O	O	O	O	O	O
次	O	O	O	O	O	O	O

Table 3.7: Different tags for 主因鼻塞 3 天，头晕 2 天，伴呕吐 5 次 (*The main reason is nasal congestion for 3 days, dizziness for 2 days with vomit 5 times*)

radicals and phonetic radicals of many characters. CCKS data is provided by an organisation in mainland China, which means all the characters are in Simplified Chinese. The experiments are repeated on Traditional Chinese data, using OpenCC to convert the original data into Traditional Chinese, the result is shown in Table 3.8.

Features	C	CP	CF	CR	CRP	CRF
BOD	0.570	0.568	0.564	0.570	0.563	0.564
SYM	0.608	0.634	0.632	0.608	0.605	0.632
DIS	0.544	0.531	0.502	0.544	0.568	0.502
EXP	0.571	0.571	0.577	0.571	0.566	0.577
TRE	0.601	0.648	0.632	0.601	0.614	0.632
ALL	**0.596**	0.594	0.595	0.596	0.585	0.595

Table 3.8: F-measures of CRF experiment on Traditional Chinese

Similar to the result of Simplified Chinese, the performance does not improve with the use of radical features, and models using the final have a better

performance than those using the piyin. Compared with the results on Simplified Chinese text, the F-measures of all named entities are higher than that in Simplified Chinese, and the F-measures of the second best models have less difference with that of the best one, which is only 0.001. All these results suggest that it is better to use radical features in Traditional Chinese than Simplified Chinese.

The result of CRF shows that the use of radical features of Chinese characters could not help in the task of NER using the basic machine learning method, and the final is a better alternative feature for phonetic radical.

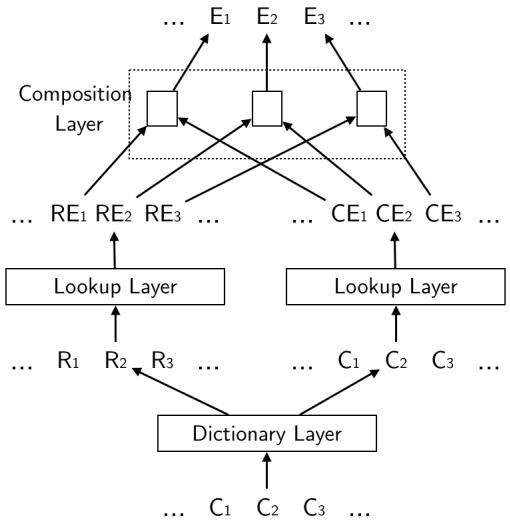
3.3 Deep Learning

3.3.1 Methodology

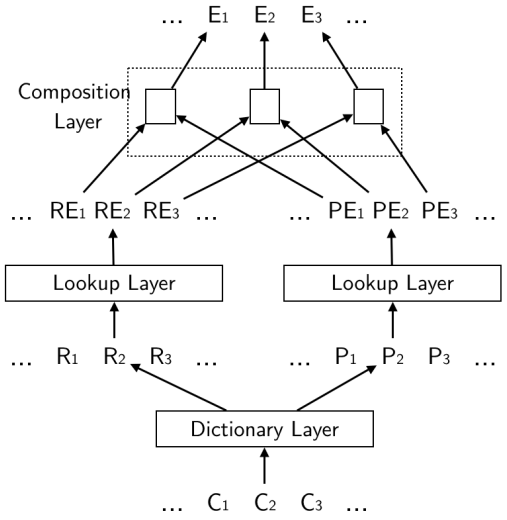
In the approach of deep learning, the radical feature should be included in the embedding of the character. The previous work of using radical features like radical embedding[DZZ+16] tries to train the word/character embedding using the similar process proposed by Bojanowski et al.[BGJM16], using the subword to form n-gram and training the embedding of the subword. Some other approaches like JWE[YJXS17] will train the subword embedding with character embedding together.

Pinyin will be used in the embedding, for the reasons demonstrated in the previous section. However, when using pinyin, it is not suitable to train them together. Pinyin is written in Latin letters and the primary radical is written in Chinese characters; the training of two completely different features together may cause some problems. So in the models proposed, the models are using the pretrained embeddings of characters, primary radicals and pinyins.

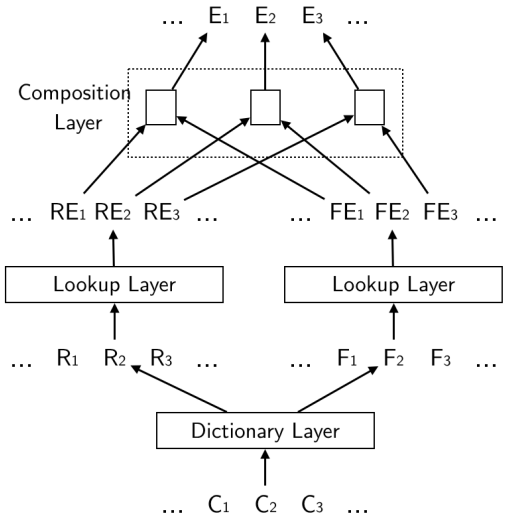
Three proposed models are shown in Fig. 3.10.



(a) Model-CR



(b) Model-RP



(c) Model-RF

Figure 3.10: Models using the radical feature

In the Model-CR shown in Fig 3.10a, the primary radical and the character itself are used to form the character embedding. The primary radical embedding RE_i and character embedding CE_i are obtained from the pretrained embeddings and form the final embedding E_i for character C_i. This method is used to test how phonetic radicals affect the result.

In the Model-RP shown in Fig 3.10b, the character embedding is formed by the primary radical and the pinyin. In Fig 3.10b, P_i, PE_i means the pinyin and pinyin embedding of character C_i, respectively. Because phonetic radicals usually do not provide tone information, only the initial and final are used for the pinyin. This method considers both semantic radicals and phonetic radicals.

In the Model-RF shown in Fig 3.10c, the character embedding comes from the primary radical and the final of pinyin. F_i and FE_i represents the final of pinyin and the embedding of the final of pinyin, respectively. This model is a modified version of Model-RP, as the initial information sometimes is not provided by the phonetic radical.

All three models have the composition layer for compositing the two different embeddings together. One simple way of composition is to get the sum of two embeddings, shown in Formula 3.5.

$$E_i = LE_i + RE_i \tag{3.5}$$

In Formula 3.5, E_i is the final embedding, and LE_i and RE_i represent the embeddings to be composed.

Besides simple addition, another way of composition include learning process is proposed. The final embedding is composed by the following formula:

$$E_i = LW_i * LE_i + RW_i * RE_i + b_i \tag{3.6}$$

In Formula 3.6, E_i, LE_i and RE_i still represent the final embedding and the embeddings to be composed, LW_i and RW_i are weight matrices and b_i is the bias matrix. During the training, both weight matrices and bias matrices would be learnt and updated.

The models are only used for embedding layers; BiLSTM-CRF will be used for further processing.

Again, all primary radical and pinyin features of all characters are taken from Unihan database.

All three models will be tested on the CCKS data used in the last section. Besides that, JWE and cw2vec, two other approaches of using graphical features of Chinese characters are tested as well.

The pretrained embedding are made on Chinese Wikipedia data, and the word2vec toolkit[3] will be used for training embeddings of features. In training the embedding of a certain feature, all characters in the text will be converted to that feature (e.g., primary radical) and then the embedding is trained based on the converted data.

To make the pretrained embedding focus on the biomedical domain, another set of embedding is trained on the all pages in the category or nested subcategory of biology or medicine. An experiment will be made on the embedding of all Wikipedia data and the embedding of biomedical data.

For a fair comparison, all the training of embedding including JWE and cw2vec are using the same set of parameter settings.

[3] `https://code.google.com/archive/p/word2vec/`

3.3.2 Result

The result of applying three novel embedding models, JWE and cw2vec are shown in Table 3.9 and 3.10, where Table 3.9 shows the result using pretrained embedding on all Wikipedia data, and Table 3.10 shows the result using pretrained embedding on biomedical domain data.

Model	JWE	cw2vec	sum			linear		
			cr	rp	rf	cr	rp	rf
BOD	0.627	0.600	0.648	0.650	0.632	0.630	0.666	0.638
SYM	0.712	0.713	0.704	0.715	0.707	0.745	0.732	0.724
DIS	0.603	0.495	0.721	0.656	0.554	0.636	0.713	0.565
EXP	0.699	0.679	0.680	0.711	0.701	0.693	0.693	0.690
TRE	0.471	0.488	0.606	0.657	0.509	0.725	0.685	0.575
ALL	0.666	0.649	0.672	0.686	0.667	0.679	**0.694**	0.672

Table 3.9: Results of using pretrained Wikipedia embedding on Simplified Chinese

Model	JWE	cw2vec	sum			linear		
			cr	rp	rf	cr	rp	rf
BOD	0.587	0.597	0.648	0.663	0.603	0.651	0.667	0.621
SYM	0.701	0.680	0.732	0.737	0.692	0.737	0.743	0.717
DIS	0.675	0.600	0.697	0.703	0.530	0.741	0.750	0.564
EXP	0.660	0.673	0.716	0.723	0.707	0.715	0.730	0.716
TRE	0.494	0.502	0.660	0.671	0.543	0.728	0.703	0.774
ALL	0.639	0.640	0.693	0.703	0.657	0.699	**0.709**	0.674

Table 3.10: Results of using pretrained biomedical Wikipedia embedding on Simplified Chinese

It can be seen that all the proposed models have better performance than the two existing models. For three models, model rp has the best performance, which suggests the model using features of phono-semantic characters is successful. The JWE model, which uses the combination of characters and primary radicals, has the similar result with cw2vec, which uses strokes.

It should be noticed that the F-measure of model cr is between that of model

rp and rf. Pinyin and final both represent the phonetic information of a Chinese character, which means the performance of model rp and rf should be similar.

Table 3.11 shows the tags of the sentence 咽部无红肿，扁桃体无肿大，口唇无发绀，口腔粘膜光滑，咽部充血，咽峡部可见疱疹 (*No redness at the pharynx, no swelling at the tonsil, no purpleness at the lip, there are congestion at the pharynx and herpes can be seen at the isthmus of pharynx*) by different models using linear composition.

There are some confusing tags in this sentence. For example, 口唇 (lip) is tagged as SYM in the golden standard, and 发绀 (purpleness) is tagged as BOD in the golden standard. But most tags tagged by the models are following the golden standard. For example, 发绀 is tagged as BOD in all three models.

In this sentence, model RF tags many entity incorrectly, such as 红肿 (redness) as BOD, 扁桃体 (tonsil) as SYM, 口唇 (lip) as EXP. While model CR can correctly tag these entities. The performance of model RP is the best in tagging this sentence as it can correctly tag 疱疹 (herpes) as BOD, while the other models tag it as SYM.

However, in common sense, 疱疹 (herpes) should be considered as a kind of symptom. The test file has been reviewed, and 疱疹 (herpes) appears 19 times, where 7 of them are tagged as SYM, and the other 12 are tagged as BOD. It means that there is a kind of rule on how 疱疹 (herpes) should be tagged.

The accuracies of tagging 疱疹 (herpes) by three different models are shown in Table 3.12, showing that model RP can tag most of 疱疹 (herpes) correctly.

To explore why model CP is the best among the three in general, the details of the embedding of the characters shared with the same primary radical are shown in Table 3.13.

It can be seen that the amount of different pinyins is very different from the amount of different finals in the characters sharing the same primary radical.

Character	True Tag	CR	RF	RP
咽	B-BOD	B-BOD	B-BOD	B-BOD
部	I-BOD	I-BOD	I-BOD	I-BOD
无	O	O	O	O
红	B-SYM	B-SYM	B-BOD	B-SYM
肿	I-SYM	I-SYM	I-BOD	I-SYM
，	O	O	O	O
扁	B-BOD	B-BOD	B-SYM	B-BOD
桃	I-BOD	I-BOD	I-SYM	I-BOD
体	I-BOD	I-BOD	I-SYM	I-BOD
无	O	O	O	O
肿	B-SYM	B-SYM	B-SYM	B-SYM
大	I-SYM	I-SYM	I-SYM	I-SYM
，	O	O	O	O
口	B-SYM	B-SYM	B-EXP	B-SYM
唇	I-SYM	I-SYM	I-EXP	I-SYM
无	O	O	O	O
发	B-BOD	B-BOD	B-BOD	B-BOD
绀	I-BOD	B-BOD	I-BOD	I-BOD
，	O	O	O	O
口	B-SYM	B-SYM	B-SYM	B-SYM
腔	I-SYM	I-SYM	I-SYM	I-SYM
粘	I-SYM	I-SYM	I-SYM	I-SYM
膜	I-SYM	I-SYM	I-SYM	I-SYM
光	O	O	O	O
滑	O	O	O	O
，	O	O	O	O
咽	B-EXP	B-EXP	B-EXP	B-EXP
部	I-EXP	I-EXP	I-EXP	I-EXP
充	B-BOD	B-BOD	B-BOD	B-BOD
血	I-BOD	I-BOD	I-BOD	I-BOD
，	O	O	O	O
咽	B-SYM	B-BOD	B-BOD	B-BOD
峡	I-SYM	I-BOD	I-BOD	I-BOD
部	I-SYM	I-BOD	I-BOD	I-BOD
可	O	O	O	O
见	O	O	O	O
疱	B-BOD	B-SYM	B-SYM	B-BOD
疹	I-BOD	I-SYM	I-SYM	I-BOD

Table 3.11: Different tags for 咽部无红肿，扁桃体无肿大，口唇无发绀，口腔粘膜光滑，咽部充血，咽峡部可见疱疹 (*No redness at the pharynx, no swelling at the tonsil, no purpleness at the lip, there are congestion at the pharynx and herpes can be seen at the isthmus of pharynx*)

Model	CR	RF	RP
Accuracy	57.9%	31.6%	78.9%

Table 3.12: The Accuracy of Tagging 疱疹 (herpes)

Primary radical	Amount of characters	Amount of pinyins	Amount of finals
疒 (illness)	160	119	31
肉 (meat)	98	75	26
水 (water)	441	219	33
人 (human)	281	174	30

Table 3.13: Detailed information about characters with the same primary radical

Take 水 (water) as the example. All 441 different characters will have 219 different embeddings when using model rp and 33 different embeddings when using model rf. The use of final in the embedding model has significantly decreased the difference among different characters in the embedding form. From the results in Table 3.9 and 3.10, although the use of pinyin also decreases the uniqueness of the characters, it still in the acceptable level.

Compared with the model using the embedding trained with all Wikipedia text and the embedding trained with biomedical Wikipedia text, the latter one improves the result when using model cr and rp. In the sum composition of model rf, the performance becomes worse when using specific domain pretrained embedding. It could be another result of the decrease of uniqueness of different characters when using the final feature.

In the comparison of different composition methods, the linear one has the better results regardless of the model used or the pretrained embedding used.

The same experiments have been done on Traditional Chinese. Both the Wikipedia corpus used for training embeddings and the CCKS datasets have been converted to Traditional Chinese. The results are shown in Table 3.14 and Table 3.15.

Model	JWE	cw2vec	sum			linear		
			cr	rp	rf	cr	rp	rf
BOD	0.608	0.618	0.622	0.633	0.616	0.624	0.639	0.625
SYM	0.685	0.701	0.698	0.710	0.719	0.701	0.717	0.722
DIS	0.599	0.569	0.644	0.738	0.585	0.645	0.741	0.612
EXP	0.697	0.671	0.722	0.718	0.690	0.720	0.722	0.704
TRE	0.549	0.468	0.580	0.693	0.474	0.634	0.705	0.456
ALL	0.655	0.651	0.675	0.685	0.662	0.678	**0.691**	0.671

Table 3.14: Results of using pretrained Wikipedia embedding on Traditional Chinese

Model	JWE	cw2vec	sum			linear		
			cr	rp	rf	cr	rp	rf
BOD	0.662	0.583	0.648	0.672	0.627	0.660	0.674	0.637
SYM	0.705	0.666	0.703	0.731	0.716	0.714	0.735	0.724
DIS	0.592	0.471	0.676	0.649	0.628	0.673	0.603	0.650
EXP	0.709	0.682	0.696	0.727	0.708	0.700	0.730	0.710
TRE	0.527	0.457	0.685	0.705	0.525	0.698	0.722	0.542
ALL	0.666	0.630	0.679	0.706	0.672	0.688	**0.708**	0.680

Table 3.15: Results of using pretrained biomedical Wikipedia embedding on Traditional Chinese

Similar to the result in Simplified Chinese, model rp using a linear composition method gets the best result, and the use of biomedical pretrained embedding has a better performance.

Comparing the results in Simplified Chinese and Traditional Chinese, the performance does not change so much, which means the simplification of the Chinese characters do not affect the use of them in a neural network so much.

As the model proposed here is based on the feature of phono-semantic characters, it is necessary to know how many phono-semantic characters are in the test data.

With the algorithm identifying the phono-semantic characters, it it easy to know how many phono-semantic characters are in a given text. The algorithm

described in Algorithm 1 is applied to CCKS data to get the proportion of phono-semantic characters, and there are only 30.87% phono-semantic characters among the all. It is quite small compared to the 90% of Boltz et al.[Bol94] or the 80% of Hoosain et al.[Hoo13]. Although the current algorithm for identifying phono-semantic characters is not perfect, the result should not differ so widely from excepted.

One possible reason is that some of the high frequent characters are non-phono-semantic characters. The frequency of characters in CCKS data is listed in Table 3.16.

Rank	Character	Frequency	Phono-semantic	Algorithm
1	无	5116	F	F
2	及	2845	F	F
3	痛	2453	T	T
4	未	2446	F	F
5	音	2328	F	F
6	双	2322	F	F
7	常	2245	T	F
8	部	2110	T	F
9	性	2090	T	T
10	查	1896	F	F
11	体	1887	F	F
12	腹	1720	T	T
13	院	1638	T	T
14	病	1534	T	T
15	患	1524	T	T
16	血	1515	F	F
17	肿	1481	T	T
18	正	1474	F	F
19	侧	1445	T	T
20	心	1389	F	F

Table 3.16: Top 20 Characters in CCKS data

In Table 3.16, the *Phono-semantic* column shows whether the character is a phono-semantic character, and the *Algorithm* column shows whether the character will be classified as phono-semantic character by the algorithm proposed. In

both columns, *T* means *True* (i.e., it is phono-semantic character) and *F* means *False* (i.e., it is not a phono-semantic character). It is shown that in the top 20 characters in CCKS data, half of them are phono-semantic characters, however, two of them cannot be correctly classified as phono-semantic characters by the algorithm due to the change of pronunciation of the phonetic radicals. Among them, the most frequent character 尤 is a non-phono-semantic character, which appears about twice as often as the second one. These factors cause the proportion of the phono-semantic characters to be only 25.75%.

Another review on all unique characters in CCKS data is made and the result is shown in Table 3.17. In this result, all different characters in the data are counted, and the phono-semantic character identifying algorithm is applied.

Unique Character	Unique Phono-semantic Character	Phono-semantic Character(%)
1557	873	56.07

Table 3.17: Phono-semantic characeters in CCKS data

It shows that more than half of the unique characters are phono-semantic characters and provides a reason why the models, which are proposed based on the feature of phono-semantic characters, have better performance than the existing models JWE and cw2vec.

3.3.3 Conclusion

This section proposed a novel embedding model based on the fact that most of the Chinese characters are phono-semantic characters. The model uses the pinyin, one of the phonetic features, and the primary radical, one of the graphical features. Compared to other approaches using radical features, this model not only captures the semantic information of the characters but also the phonetic information of the characters. And without using all radicals, the proposed model

is simpler than the existing models. The embedding model is tested with the existing models in the same document in both Simplified Chinese and Traditional Chinese. The proposed model has the best performance, and the results are similar in both Simplified Chinese and Traditional Chinese, suggesting that the simplification of the Chinese characters does not affect the use of the radical features of the Chinese characters.

Because the proposed model is based on the structure of the phono-semantic characters, the model will work better in the domains where phono-semantic characters occur frequently. Besides the biomedical domain, the chemical domain where most named entities are formed by phono-semantic characters is also suitable for the proposed domain.

3.4 Summary

This chapter aims to use radical features in both the basic machine learning method and the deep learning method in the NER task. The previous works on NER and different word embedding models have been reviewed. In the basic machine learning approach, CRF is used for the experiment, and the result shows that the use of radical features does not improve the performance. In the deep learning approach, a novel embedding model is proposed that uses the primary radicals and the pinyins of the characters. The proposed model is tested with other existing models that use the graphical feature of the characters in the BiLSTM-CRF model. The result shows that the proposed model has the best performance. In the basic machine learning experiment, the final is a good alternative feature for the phonetic feature, while all the pinyin are useful in the deep learning experiment. The reason for this may be that the use of the neural network is able to discover the relationship between different pinyins with the

same final. The experiments between Simplified Chinese and Traditional Chinese show that Traditional Chinese is better in basic machine learning approach, while the deep learning approach works similarly in both Simplified Chinese and Traditional Chinese.

Chapter 4

Terminology Extraction

4.1 Previous Work

According to Pavel's handbook[PNL01], terminology is defined as "*the language discipline dedicated to the scientific study of the concepts and terms used in specialised languages*". Specialised language is different from the general language, which is that used in daily life. Specialised language is used to facilitate unambiguous communication in a particular area of knowledge, based on a vocabulary and language usage specific to that area. In another word, terminology is the study of terms.

Term usually refers to a specific concept in a certain subject field. The subject field knowledge is usually presented in its own sublanguage, showing its uniqueness on the lexicons, syntax and semantics [Ana94].

Terminology Extraction (TE) is the task of finding the terms in the given text. The traditional method of terminology extraction relies on a terminologist who will look through the special concepts and the linguistic relationship among them in the specified domain to find the terms.

In the approaches of TE task, similar as NER task, there are traditional

methods and machine learning methods. Due to the lack of a labelled gold standard in Chinese, this thesis will focus on the traditional method.

Based on the definition of term, the words that are terms should appear more often in the texts in the specified domain, so it should be easy to find the terms by some statistical information.

C-value

C-value is a value to measure the termhood of a multi-word. In the approach of using C-value, a corpus will be the input and a list of the candidates of multi-word term with their C-values will be output[FAM00].

The detailed steps of using C-value are shown below[FAM00]:

1. Use natural language processing tool to tag the corpus so that we can get the tags of each words.

2. Write a linguistic rule to filter the candidates, such as a regular expression like $(Adj/Noun)^{+}Noun$.

3. If a candidate is in the stop list, then remove it from the candidate list. A stop list is a list of some common words that can be created by applying terminology extraction on common texts.

4. Determine the frequency of each candidate c_i as $f(c_i)$.

5. Then the C-value of candidate c_i will be calculated as

$$C-value(c_i) = \begin{cases} \log_2|c_i| * f(c_i) & c_i\text{is not nested} \\ \log_2|c_i| * (f(c_i) - \frac{1}{P(T_{c_i})}\Sigma_{a \in T_{c_i}} f(a)) & \text{otherwise} \end{cases} \quad (4.1)$$

where $|c_i|$ is the length of c_i, T_{c_i} is the the set of extracted candidates

containing c_i, and $P(T_{c_i})$ is the number of such candidates. Thus a term candidate is nested if it appears in other candidates.

As shown, the approach of using C-value uses both linguistic and statistical methods to decide if a multi-word phrase is a term or not. The threshold of the C-value is usually calculated by $\log_2|\bar{t}| * 2$, where $|\bar{t}|$ is the average length of a term. This values assumes a candidate whose length is as same as the average term length could be considered as a term if it appears more than twice.

Term Frequency–Inverse Document Frequency

Term Frequency–Inverse Document Frequency (TF-IDF) is another value to measure the termhood of a word, it is proposed by Sparck Jones [SJ72][SJ04], the formula of TF-IDF is [ZYT11]:

$$IDF_{i,j} = tf_{i,j} \times \log(\frac{N}{df_i}) \tag{4.2}$$

In the formula, $w_{i,j}$ is the weight of word i in document j, $tf_{i,j}$ is the frequency of word i in document j, N is the amount of documents in the whole collection, and df_i is the frequency of word i in the whole collections.

The first item in the formula is term frequency and the second one is the inverse document frequency. Term frequency shows how often the word appear in the document, and is used for measuring termhood base on the feature that terms will occur more than some common words in the specified domain text. And Inverse Document Frequency (IDF) shows how often the word appear in all the documents, IDF will be smaller when the word appear more in al the document, because it is inversed. IDF is used for measuring termhood based on the feature that terms in specified domain will less frequently appear in the common text[Rob04].

When using TF-IDF, similar to the method in C-value, it is also necessary to get the candidate lists of terms by some linguistic rules. One method of calculating the threshold is to use the average value of all TF-IDF after the calculation of all IDF of the words in the candidate list.

Kullback–Leibler divergence

The Kullback-Leibler (KL) divergence is a measurement from information theory, defined as *a measure of the inefficiency of assuming that the distribution is q when the true distribution is p* by the following formula.[Cov99]

$$KLD(p,q) = \Sigma_x p(x) \log \frac{p(x)}{q(x)} \tag{4.3}$$

In terminology extraction, KL-divergence can be used to measure the information lost when using the probability of a term w in one collection of documents to describe the probability of w in another collection. So if a term candidate has high KL-div between the documents in certain domain and the general documents, this could be a term.

In the work of Tomokiyo et al.[TH03], KL-divergence is used to measure both the informativeness and the phraseness by the following formula, respectively.

$$KLD_i(c) = P(c|D) \log \frac{P(c|D)}{P(c|G)} \tag{4.4}$$

$$KLD_p(c) = P(c|D) \log \frac{P(c|D)}{\prod_{i=1}^{n} P(u_i|D)} \tag{4.5}$$

In the formulas, $P(c|D)$ is the probability of candidate c occurs in the collection of the certain domain documents, while $P(c|G)$ is that in the collection of the general documents. u_i is the ith element in candidate c, thus $P(u_i|D)$ is the probability of that element in the collection.

The DL-divergence of informativeness is used for the measurement of the information lost as mentioned before. The DL-divergence of phraseness is used for the measurement of the likelihood that the elements are forming the candidate c.

Then the termhood of a candidate c is simply calculated by sum two KL-divergences together shown by the following formula.

$$KLD(c) = KLD_i(c) + KLD_p(c) \tag{4.6}$$

Domain Relevance Domain Consensus

In the approach of Domain Relevance Domain Consensus (DRDC), two factors are calculated, Domain Relevance (DR) and Domain Consensus(DC).[NV04]

The DR is used to measure the termhood of a candidate in a certain domain comparing to the general documents, and the DC is used to measure the termhood of a candidate within the documents in the certain domain. DR, DC and DRDC can be calculated by the following formulas.

$$DR(c) = \frac{P(c|D)}{\max P(t|G_i)} \tag{4.7}$$

$$DC(c) = \sum_{d \in D} (P(c|d) \log \frac{1}{P(c|d)}) \tag{4.8}$$

$$DRDC(c) = \alpha DR(c) + \beta DC(c) \tag{4.9}$$

In the formulas, $P(c|D)$ is the probability of candidate c occurs in the collection of the certain domain documents, $\max P(t|G_i)$ get the maximum probability of candidate c occurs in the document among general documents, $P(c|d)$ get the probability of candidate c occurs in the document d among the certain domain documents, and α and β used for DRDC calculation are the normalisation factors,

where $\alpha, \beta \in (0,1)$.

In the calculation of DC, not only one certain domain document is required, which filters many non-term candidate that only appear frequently in one document. So that the use of DRDC requires several general documents, as well as certain domain documents, which makes this method very different from others.

Chinese Processing

In Chinese processing, the methods used in English could be also applied. However, based on the experiment of Wang et al.[WTTA12], the use of c-value in Chinese cannot deal the nested problems well. There are still many candidates that are nested, so mutual information is used to remove the candidates with unnecessary prefixes and suffixes.

Mutual information is widely used to measure the dependency of two variables in probability theory. One of the variants, pointwise mutual information, can be defined as[MMS99]:

$$PMI(x,y) = \log \frac{P_{X,Y}(x,y)}{P_X(x)P_Y(y)} \tag{4.10}$$

where x and y are events, $P_{X,Y}(x,y)$ is the joint probability of the occurrence of x and y, and $P_X(x)$ and $P_Y(y)$ are the probabilities of the occurrence of x and y, respectively.

The mutual information used in the approach of Wang et al.[Wan12] is another variant of Formula 4.10 proposed by Magerman et al.[MM90]:

$$MI(x,y) \approx \log_2 \frac{A \times (A+B+C+D)}{(A+C) \times (A+B)} \tag{4.11}$$

where A, B, C and D are the frequencies shown in Table 4.1. In Table 4.1, X and Y means x and y occur, respectively, while $\bar{X}$ and $\bar{Y}$ means X and Y do not

occur, respectively.

	Y	$\bar{Y}$
X	A	B
$\bar{X}$	C	D

Table 4.1: Frequencies in Formula 4.11

Based on the average mutual information between two neighbouring character in a common text, the term candidates with the affix having higher mutual information than average will be removed because they are mostly common words[Wan12].

Another try on eliminating the nested terms is made by Hu et al.[HZJ13], where C-value has been improved. In the work of Hu et al., IC-value is proposed, which can be calculated by the following formula.

$$IC-value(c_i) = \begin{cases} |c_i| * f(c_i) * \log(\frac{N}{g(c_i)}) & c_i\text{is not nested} \\ |c_i| * (f(c_i) - \Sigma_{a \in T_{c_i}} f(a)) * \log(\frac{N}{g(c_i)}) & \text{otherwise} \end{cases} \tag{4.12}$$

In the formula, $|c_i|$ is the length of c_i, T_{c_i} is the the set of extracted candidates containing c_i, $g(c_i)$ is the document frequency of word c_i, and N is the amount of total documents.

It is a combination of C-value and TF-IDF. TF-IDF is included in the formula for better measure the termhood of a given term candidate.

However, in the processing of Classical Chinese, it is not easy to select a large amount of documents, as the writing styles changes a lot and the amount of Chinese characters are increasing. To make the writing styles and the possible characters constant, the documents should be selected in a certain short period. So TF-IDF is not a suitable method in processing Classical Chinese.

Although there are several different research about the statistical methods in Chinese terminology extraction, however, none of the research uses the graphical features of the characters. For example, in the research of Du et al.[DLY+16], mutual information is used for term extraction, Liang et al.[LZZ10] use the combination of the c-value and the mutual information, and Han et al.[HA12] use the combination of the c-value and unithood. One of the reasons is that characters or words themselves and the relationship between characters or words are focused in the statistical methods, while the structures of the characters are always ignored. However, in the medical domain, as shown in Table 2.1, most of the terms of the name of the diseases contain the characters whose primary radical is 疒 (sickness). The primary radical should be useful in the Chinese terminology extraction in such domain.

4.2 Methodology

This chapter focus on the experiment of terminology extraction on Traditional Chinese Medicine texts in Classical Chinese.

4.2.1 Primary radicals in terminologies

Gold standard variability is one of the difficulties of evaluating terminology extraction stated by Nazarenko et al.[NZ09]; different terminologists may produce different lists of terms when given the same text. In this experiment, the list produced by the World Health Organisation (WHO), *WHO International Standard Terminologies On Traditional Medicine In the Western Pacific Region* [O+07] will be used.

The list shows the terminologies on Traditional Medicine In the Western Pacific Region, which includes the traditional medicine in China, Japan, Korea

and Vietnam. All of these different traditional medicine are highly influenced by Traditional Chinese Medicine, so it is suitable to use it as a golden standard in this task.

The terminologies are categorised into eight parts:

1. General
2. Basic Theories
3. Diagnostics
4. Disease
5. Therapeutics
6. Acupuncture and Moxibustion
7. Medicinal Treatment
8. Classics of Traditional Medicine

The last category only contains the name of some books and can be ignored.

The amount of terminologies of different categories are shown in Table 4.2.

Category	Term Amount
General	42
Basic Theories	887
Diagnostics	962
Disease	703
Therapeutics	430
Acupuncture and Moxibustion	340
Medicinal Treatment	232
Total	3596

Table 4.2: Categories in *WHO International Standard Terminologies On Traditional Medicine In the Western Pacific Region*

When exploring the list, a lot of one-character terms are found. Most current TE methods can only detect multi-character words. For example, assuming that

c_i is a one-character candidate, when applying Formula 4.1, $\log_2|c_i|$ will always be 0, and thus the c-value of c_i is 0. So all one-character word are removed, and Table 4.3 shows the situation after removing them.

Category	Term Amount
General	42
Basic Theories	817
Diagnostics	950
Disease	677
Therapeutics	430
Acupuncture and Moxibustion	337
Medicinal Treatment	228
Total	3481

Table 4.3: Categories in *WHO International Standard Terminologies On Traditional Medicine In the Western Pacific Region* after removing one-character word

The test text selected in the experiment is one of the classics of Traditional Chinese Medicine, *Bei Ji Qian Jin Yao Fang* (Essential Prescriptions Worth a Thousand Gold for Emergencies) written in the 7th century.

This book contains 30 chapters and 548,167 characters, summarising thousands of prescriptions, what diseases they are used for, and the causes of the diseases. The text of this book in this experiment is collected from *Digital Library of Traditional Chinese Medicine Classics*[1] established by China Academy of Chinese Medical Sciences.

To identify the terms in the test text, another common text is necessary as well. Because *Bei Ji Qian Jin Yao Fang* is written in Classical Chinese. Based on the previous discussion, the common text should be also selected from books written in Classical Chinese in the similar period. Three official historical books written in the same era are selected: *Jin Shu (Book of Jin)*, *Nan Shi (History of the Southern Dynasties)* and *Bei Shi (History of the Northern Dynasties).* All

[1] `http://www.zywx.org.cn/static/reader/index.html`

three books were written in the 7th century as well and recorded the history from the 3rd century to the 6th century. The books are collected from Wikisource. All the three books contain 11,171,847 characters in total.

To use the radical feature in the terminology extraction, all characters in the common text have been examined and the result is shown in Table 4.4.

Rank	Primary Radical	Frequency	Percentage(%)
1	人	188590	6.27
2	口	116825	3.88
3	一	107801	3.58
4	丿	89693	2.98
5	辶	71878	2.39
6	言	71436	2.37
7	水	71432	2.37
8	宀	69622	2.31
9	日	66070	2.20
10	八	62986	2.09
11	心	60567	2.01
12	大	60208	2.00
13	木	54572	1.81
14	二	49173	1.63
15	丶	47774	1.59
16	月	46029	1.53
17	手	41767	1.39
18	子	40971	1.36
19	刀	40835	1.36
20	十	40763	1.35

Table 4.4: Top 20 Primary Radicals in Common Text

The medical text, *Bei Ji Qian Jin Yao Fang* has been processed similarly, and the result is shown in Table 4.5.

It is clear that the distribution of primary radicals are quite different in the two texts. For example, the primary radicals 肉 (meat), 艸 (grass), 疒 (illness) and 火 (fire) occur much more in the medical text than that in the common text, and these primary radicals also occur a lot in the terms of the medical text. Table 4.6 shows the primary radicals in the *WHO International Standard*

Rank	Primary Radical	Frequency	Percentage(%)
1	一	27441	6.33
2	口	22940	5.29
3	水	21809	5.03
4	人	18643	4.30
5	肉	15791	3.64
6	艸	14834	3.42
7	木	11720	2.7
8	二	10749	2.48
9	十	10678	2.46
10	火	9528	2.20
11	疒	9216	2.13
12	丿	8346	1.93
13	心	8290	1.91
14	又	7514	1.73
15	月	7435	1.72
16	刀	6966	1.61
17	日	6963	1.61
18	大	6427	1.48
19	丶	6376	1.47
20	方	5915	1.36

Table 4.5: Top 20 Primary Radicals in Medical Text

Terminologies On Traditional Medicine In the Western Pacific Region and the occurrence in both the common text and medical text, where the *Difference* column shows the absolute value of the difference between the percentage frequency in the common text and that in the medical text, which is used to find the primary radicals whose frequency changes a lot in two different kinds of text.

It is clear that the characters whose occurrence changes a lot in common text and medical text including 肉 (meat), 艸 (grass), 疒 (illness) and 火 (fire) play an important role in the terminologies of Traditional Chinese Medicine. Table 4.7 shows the top 20 radicals with the highest difference. Similar with Table 4.6, the *Difference* column shows the absolute value of the difference between the percentage frequency in the common text and that in the medical text. *Term Rank* column shows the rank of the occurrence in *WHO International Standard*

Rank	Primary Radical	Frequency	Percentage (%)	Common Text Frequency (%)	Medical Text Frequency (%)	Difference
1	水	914	8.37	2.37	5.03	2.66
2	肉	909	8.33	0.45	3.64	3.19
3	疒	608	5.57	0.14	2.13	1.99
4	言	569	5.21	2.37	0.62	1.75
5	火	530	4.86	0.52	2.20	1.68
6	阜	448	4.10	1.28	1.14	0.14
7	气	292	2.67	0.05	0.65	0.60
8	宀	287	2.63	2.31	1.03	1.28
9	口	282	2.58	3.88	5.29	1.41
10	一	276	2.53	3.58	6.33	2.75
11	糸	243	2.23	1.07	0.92	0.15
12	艸	243	2.23	1.31	3.42	2.11
13	心	238	2.18	2.01	1.91	0.10
14	人	229	2.10	6.27	4.30	1.97
15	虍	220	2.02	0.11	0.27	0.16
16	刀	205	1.88	1.36	1.61	0.25
17	手	185	1.69	1.39	1.01	0.38
18	血	169	1.55	0.02	0.25	0.23
19	金	150	1.37	0.44	0.49	0.05
20	凧	144	1.32	0.10	0.40	0.30

Table 4.6: Top 20 Primary Radicals in the *WHO International Standard Terminologies On Traditional Medicine In the Western Pacific Region*

Terminologies On Traditional Medicine In the Western Pacific Region.

It can be found that most primary radicals that having a high difference also occur a lot in the terms. It suggests that the difference between the proportions in the common text and medical text could be used for filtering term candidates.

4.2.2 RC-value

Based on the finding in the last section, an advanced version of C-value, RC-value (Radical C-value) is proposed. Compared to C-value, RC-value will use the frequency difference of the primary radical in common text and specified domain

Rank	Primary Radical	Common Text Frequency (%)	Medical Text Frequency (%)	Difference	Term Rank
1	肉	0.45	3.64	3.19	2
2	一	3.58	6.33	2.75	10
3	水	2.37	5.03	2.66	1
4	艸	1.31	3.42	2.11	12
5	疒	0.14	2.13	1.99	3
6	人	6.27	4.30	1.97	14
7	言	2.37	0.62	1.75	4
8	火	0.52	2.20	1.68	5
9	辶	2.39	0.86	1.53	25
10	口	3.88	5.29	1.41	9
11	宀	2.31	1.03	1.28	8
12	十	1.35	2.46	1.11	55
13	方	0.25	1.36	1.11	77
14	丿	2.98	1.93	1.05	64
15	玉	1.06	0.11	0.95	89
16	木	1.81	2.70	0.89	23
17	巾	1.12	0.23	0.89	104
18	邑	1.03	0.17	0.86	34
19	二	1.63	2.48	0.85	30
20	八	2.09	1.25	0.84	38

Table 4.7: Top 20 Primary Radicals with the highest difference between the proportions in common text and medical text

text to determine the termhood of a candidate. The steps of the calculation of RC-value are listed below:

1. For each primary radical R_i, get the percentage frequency in common text $f_c(R_i)$ and the percentage frequency in specified domain text $f_s(R_i)$.

2. Use linguistic rules to filter the candidates from the specified domain text.

3. Get the frequency of each candidate c_i as $f(c_i)$.

4. The primary radical weight w_i of candidate c_i is calculated as:

$$w_i = \frac{\Sigma_{r \in R_{c_i}} \left| f_c(r) - f_s(r) \right|}{|c_i|} \tag{4.13}$$

where R_{c_i} is the set of all primary radicals in candidate c_i, $|c_i|$ is the length of c_i.

5. Then the RC-value of candidate c_i will be calculated as

$$RC - value(c_i) = \begin{cases} w_i * \log_2 |c_i| * f(c_i) & c_i \text{is not nested} \\ w_i * \log_2 |c_i| * (f(c_i) - \frac{1}{P(T_{c_i})} \Sigma_{a \in T_{c_i}} f(a)) & \text{otherwise} \end{cases} \tag{4.14}$$

where T_{c_i} is the the set of extracted candidates containing c_i, $P(T_{c_i})$ is the number of such candidates.

$f_c(R_i)$ and $f_s(R_i)$ are in percentage form to ensure that the calculated RC-value is not too small because the total amount of primary radicals are about 200.

Similar as the method using C-value, there should be a threshold for RC-value as well. Following the idea of the calculation of C-value, the threshold of RC-value can be calculated by $\frac{100}{N(r)} * \log_2 |\bar{t}| * 2$, where $N(r)$ is the amount of all different primary radicals in the specified domain text, so that $\frac{100}{N(r)}$ is the average percentage frequency of a primary radical, and $|\bar{t}|$ is the average length of a term. This value assumes a candidate should appear more than twice to consider it as a term.

4.3 Experiment

The overall flowchart of the experiment is shown at Figure 4.1.

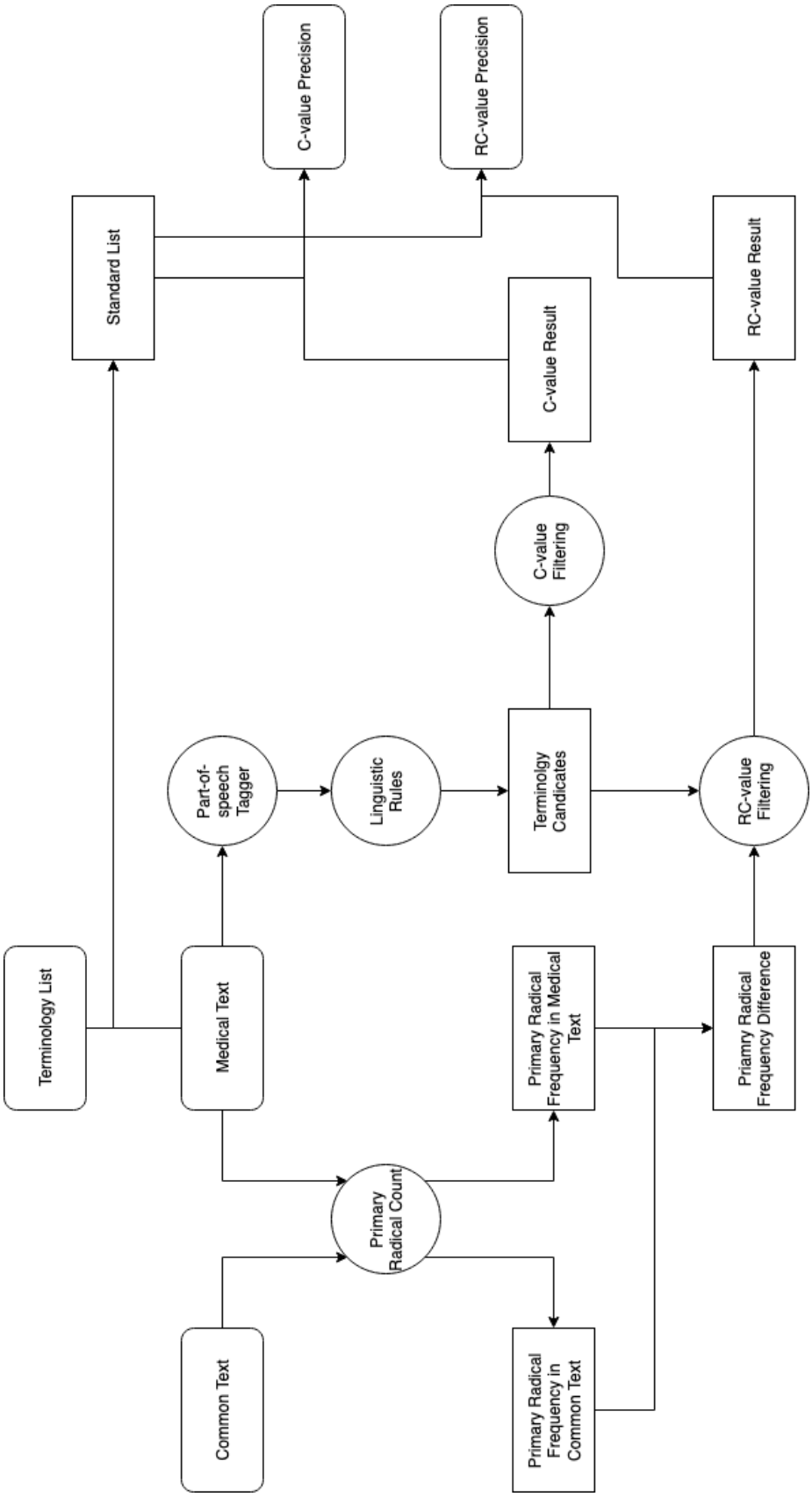


Figure 4.1: Flowchart of Terminology Extraction

In this experiment, *WHO International Standard Terminologies On Traditional Medicine In the Western Pacific Region* will be used as golden standard (i.e. *Terminology List* in Figure. 4.1). As the method is based on the exploration of *Bei Ji Qian Jin Yao Fang*, another test text is necessary to proof that the method is adaptable.

In the experiment on *Bei Ji Qian Jin Yao Fang*, three historical books (*Jin Shu*, *Nan Shi* and *Bei Shi*) will be considered as common text, and *Bei Ji Qian Jin Yao Fang* will be the medical domain text, which is also the test data.

Another experiment will be made on *Zhou Hou Bei Ji Fang (Handbook of Prescriptions for Emergency)*. Similar as *Bei Ji Qian Jin Yao Fang*, *Zhou Hou Bei Ji Fang* summarised thousands of prescriptions. Youyou Tu got idea of using Artemisinin from this book and was awarded Nobel Prize in Physiology or Medicine[Tu11]. The book was written in the 4th century, so different common texts should be selected to ensure the writing styles are similar. In the experiment of *Zhou Hou Bei Ji Fang*, two historical books *San Guo Zhi (Records of the Three Kingdoms)* and *Hou Han Shu (Book of the Later Han)* will be used. Those two books were written in the 4th or 5th century recording the history of China from the 1st century to 3rd century. The data of *Zhou Hou Bei Ji Fang* is collected from *Digital Library of Traditional Chinese Medicine Classics* as well, and the data of the common text (two historic books) comes from Wikisource.

The size of the common data and the medical domain data in both experiment is shown in Table 4.8.

Experiment	Characters in test text	Characters in common text
Bei Ji Qian Jin Yao Fang	548,167	11,171,847
Zhou Hou Bei Ji Fang	102,575	6,317,091

Table 4.8: Data size of the experiments for terminology extraction

Because both *Bei Ji Qian Jin Yao Fang* and *Zhou Hou Bei Ji Fang* are in

Classical Chinese, the word segmentation is not so necessary. A naive search has been performed on both books to find the terms in *WHO International Standard Terminologies On Traditional Medicine In the Western Pacific Region*. As a result, 822 terms are found in *Bei Ji Qian Jin Yao Fang* and 321 terms are found in *Zhou Hou Bei Ji Fang*, which will be used for testing (i.e., *Standard List* in Figure. 4.1).

Currently there is not a tool designed for processing Classical Chinese. However, the difference between Classical Chinese and Modern Chinese is not so significant, and the tools for Modern Chinese could be used for Classical Chinese. In this experiment, TsingHua Univeristy Lexical Analyser for Chinese (THULAC)[2] will be used for part-of-speech tagging. THULAC is developed by Tsinghua University, and the models for words segmentation and tagging were trained by using The People's Daily Corpus[MS16].

As mentioned, the candidate list of terms is produced by linguistic rules. From the exportation of the terms in *WHO International Standard Terminologies On Traditional Medicine In the Western Pacific Region*, there are not only the noun terms (e.g., organ names, concept), but also the verb terms (e.g., the method of curing). So the linguistic rule is shown in the form of Deterministic Finite Automaton (DFA) in Figure 4.2. In Figure 4.2, n represents for noun, v represents verb, adj represents adjective, adv represents adverb, and m and q represent a special part-of-speech in Chinese numeral and quantifier, respectively.

The average length of the terms in *WHO International Standard Terminologies On Traditional Medicine In the Western Pacific Region* is 3.1, so the threshold of C-value in the experiments is calculated as 3.26. The total amount of different primary radicals in *Bei Ji Qian Jin Yao Fang* is 208, so the threshold of RC-value in this experiment is calculated as 1.57. The total amount of different

[2] `http://thulac.thunlp.org/`

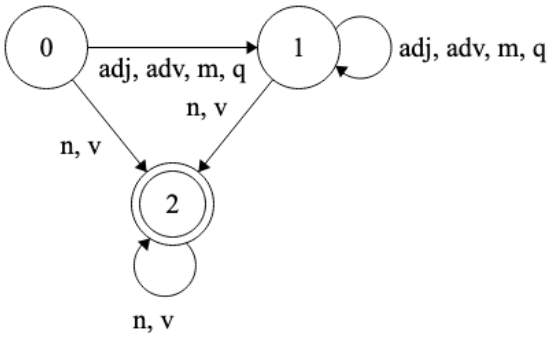


Figure 4.2: Linguistic rule for candidates filtering

primary radicals in *Zhou Hou Bei Ji Fang* is 204, so the threshold of RC-value in this experiment is calculated as 1.60. In the experiment, all candidate words whose value are higher than the threshold will be considered as terms.

In order to remove some common words in the candidates list, mutual information(MI) should be applied. Mutual information is used to measure the dependency of two characters, which means there should be a special method used for the candidates which have more than two characters. One possible solution is to check all the MIs between each pair of the adjacent characters in the word. It is because for example a common word could appear in any positions in a candidate word, shown in Table 4.9.

Each candidate shown in Table 4.9 has a high frequency in the document, and has 妇人 (woman) inside. Obviously none of them is a terminology, they

Candidates	Meaning
妇人产难死	the woman died in childbirth
始妇人	begin with the women
治妇人患癖	cure the illness of the women

Table 4.9: Examples of common word in terminology candidates

are some frequently used phrases. When checking the mutual information of each pair of adjacent characters in the candidate, all of them will be filtered out because 妇人 (woman) is a common word and has a high MI value in the general documents.

The algorithm of filtering by MI is shown in Algorithm 2.

Algorithm 2 Pseudo code of the algorithm for filtering candidates by mutual information

```
cText is the common text
cList is the list of terminology candidates
for all bigram in cText do
  m ← MI(bigram)
end for
m̄ ← average(m)
for all candidate in cList do
  filter ← False
  for c_i c_{i+1}, i ∈ [0,len(candiate) do
    if MI(c_i c_{i+1}) > m̄ then
      filtered ← True
    end if
  end for
  if filtered = False then
    result ← candidate
  end if
end for
return {result}
```

4.4 Result

Besides C-value and RC-value, the Termolator[MHG+18] is also applied for testing. The Termolator uses a combination of TF-IDF, KLD and DRDC to

measure the termhood of a candidate.

The results of the two experiments are shown in Table 4.10, where *R* represents recall, *P* represents precision, and *F-1* represents F-1 score (F-measure). *BJQJYF* represents *Bei Ji Qian Jin Yao Fang*, and *ZHBJF* represents *Zhou Hou Bei Ji Fang*.

Experiment	C-value			RC-value			Termolator		
	P	R	F-1	P	R	F-1	P	R	F-1
BJQJYF	28.6%	9.61%	14.4%	31.2%	14.6%	**19.9%**	35.1%	6.33%	10.7%
ZHBJF	8.65%	2.18%	3.48%	18.7%	6.85%	**10.0%**	12.2%%	1.56%	2.76%

Table 4.10: Result of Terminolofy Extraction of *Bei Ji Qian Jin Yao Fang* and *Zhou Hou Bei Ji Fang*

It can be found that RC-value filtering on both experiments has a better result than C-value filtering. All of the precision, recall and F-measure results are improved. The performance of both C-value and RC-value is much better than that of the Termolator. The reason could be that the Termolator uses DRDC as one of the measurement, which requires more than one document in the certain domain, however, in each test, there is only one document, and it is split into two parts. Another reason is that the Termolator rely more on the results of the POS tagging, but currently there is no tool designed for Classical Chinese.

In both experiments of C-value and RC-value, recalls are also small, which is probably the problem of the linguistic rule. Table 4.11 shows the amount of terms in *WHO International Standard Terminologies On Traditional Medicine In the Western Pacific Region* in all the candidates that produced by linguistic rule.

Medical Text	Term Amount	Candidate Amount
Bei Ji Qian Jin Yao Fang	158	17292
Zhou Hou Bei Ji Fang	36	4798

Table 4.11: Amount of terms among the candidates produced by linguistic rule

It is mentioned that the total amount of terms in *Bei Ji Qian Jin Yao Fang*

and *Zhou Hou Bei Ji Fang* after the naive search is 822 and 321, respectively. The term amounts among the candidate words are much smaller than these amounts. One possible reason is that natural language processing tool of Modern Chinese is used for tagging, while these two books are in Classical Chinese, so that some words may be tagged incorrectly. For example, yinyang is an important concept in Traditional Chinese Medicine, and it should be tagged as a noun. However, from the result of tagging, yinyang in the medical text is tagged as an adjective, which resulted in yinyang failing to be a candidate by the linguistic rule.

The term amount shown in Table 4.11 is the maximum amount of terms that can be extracted using C-value or RC-value. So the recall should be calculated using these values. Recalculated recall and F-measure are shown in Table 4.12. The recall in Table 4.12 is more reasonable now.

Experiment	C-value			RC-value			Termolator		
	P	R	F-1	P	R	F-1	P	R	F-1
BJQJYF	28.6%	53.4%	37.3%	31.2%	81.1%	**45.1%**	35.1%	35.1%	35.1%
ZHBJF	8.65%	20.6%	12.2%	28.7%	64.7%	**29.0%**	12.2%	14.7%	13.3%

Table 4.12: Recalculated result of Terminology Extraction of *Bei Ji Qian Jin Yao Fang* and *Zhou Hou Bei Ji Fang*

The F-measure of each method is not so satisfactory, especially that the precisions are still too low. The reason for that might be the task of terminology extraction of Traditional Chinese Medicine in Classical Chinese is a difficult task at the moment. First, there is no natural language processing tool designed for Classical Chinese, which make the candidates generated unsatisfactory. The result of the Termolator, which rely much on the result of POS tagging, is an evidence for that. Another reason is that the use of statistical methods in Chinese would have the problem of out-of-vocabulary. Although all Chinese characters is known for the machine, but because of the large amount of Chinese characters, there is a larger amount of different combinations for bigram, trigram or other

multigram. The machine is not possible to know every bigram in the general document. This problem not only occurs in Chinese, it is also a problem in some languages with the similar writing system such as Japanese[Bre17]. It is another big area, and it is not suitable to go deeper here.

All the results above are made on Simplified Chinese, but only the medical text obtained from a Mainland Chinese institution is in Simplified Chinese, so all other materials in Traditional Chinese were converted to Simplified Chinese in the above experiments. Another set of experiment on Traditional Chinese is required to see how the proposed method performs in a different writing format, in this one all other materials except medical text are original one, only medical text (i.e., *Bei Ji Qian Jin Yao Fang* and *Zhou Hou Bei Ji Fang*) were converted to Traditional Chinese.

Table 4.13 shows the result in Traditional Chinese, when all of the common texts, the medical texts, term list were converted to Traditional Chinese.

Experiment	C-value			RC-value		
	P	R	F-1	P	R	F-1
Bei Ji Qian Jin Yao Fang	27.5%	52.4%	36.1%	29.7%	76.6%	**42.8%**
Zhou Hou Bei Ji Fang	8.53%	21.9%	12.2%	17.0%	62.5%	**26.7%**

Table 4.13: Result of C-value and RC-value filtering of *Bei Ji Qian Jin Yao Fang* and *Zhou Hou Bei Ji Fang* in Traditional Chinese

The c-value is simply related to the occurrence of the candidate words, so theoretically the c-value perform the same in both Simplified Chinese and Traditional Chinese. However, in the experiment on *Bei Ji Qian Jin Yao Fang*, the performance on the text in Traditional Chinese is worse than that in Simplified Chinese. From checking the results in Simplified Chinese and Traditional Chinese, the following three words are missing in the result in Traditional Chinese:

1. 五脏, five viscera (organs including heart, liver, spleen, lung and kidney)

2. 口干, dry mouth

3. 针灸, acupuncture and moxibustion

All three missing words are related to the problem in converting from Simplified Chinese to Traditional Chinese. The official Traditional Chinese version of these three words (i.e., the version in the *WHO International Standard Terminologies On Traditional Medicine In the Western Pacific Region*) are:

1. 五臟

2. 口乾

3. 鍼灸

However, in the converted text, these three words are:

1. 五藏

2. 口幹

3. 針灸

In these results, 五臟 and 五藏 are both possible writings of 五脏 in Classical Chinese. 乾 (dry) and 幹 (do or the main part) are both the corresponding character of 干 (dry, do or the main part) in Traditional Chinese, but with the different meanings. 乾 means dry, and 幹 means do or the main part. The converter fails to get the correct meaning in this case. 針灸 is the usually word for 针灸 in Traditional Chinese. However, *WHO International Standard Terminologies On Traditional Medicine In the Western Pacific Region* uses 鍼灸 in its recording. It is the word writing in Japanese Kanji and is really rare in any text recording Traditional Chinese Medicine. 鍼 (needle) in Chinese is one of the variant characters

of 针 (needle), so the converter can convert it into 针 (needle) in the conversion to Simplified Chinese, which makes 针灸 a term candidate correctly.

So in the calculation of true positive, it failed to match these three words as terms, which makes the result in Traditional Chinese slightly worse than that in Simplified Chinese.

In the RC-value approach, the results are highly related to primary radicals, so that the performance in Traditional Chinese and Simplified Chinese should be different. In Table 4.12 and Table 4.13, it can be found that the RC-value has a better performance in Simplified Chinese.

Some of the examples of missing terms in Traditional Chinese are listed below:

1. 五脏, five viscera (organs including heart, liver, spleen, lung and kidney)
2. 火针, fire needling
3. 头重, heavy-headedness
4. 心烦, vexation

Case 1 and Case 2 are similar to the problems of C-value mentioned above. They are caused by the conversion between Simplified Chinese and Traditional Chinese, where 脏 and 针 are converted to 藏 and 鍼, respectively, while they are written as 臟 and 鍼 in the terms.

Case 3 is caused by the change of primary radical during simplification. 头 (head) is in Simplified Chinese with the primary radical 大 (big), and 頭 (head), the character in Traditional Chinese has 頁 (page) as its primary radical. Compared to 頁 (page), 大 (big) is more related to the concept of head.

In Case 4, the primary radicals of both characters are not changed. 心烦 will be converted to 心煩 in Traditional Chinese, in both words, 心 (heart) and

火 (fire) are the primary radicals. However, the difference of primary radical frequency between the common text and medical text changes. Table 4.14 shows the frequency of these two primary radicals in Simplified Chinese and Traditional Chinese.

Primary Radical	Simplified Chinese			Traditional Chinese		
	Common Text Frequency (%)	Medical Text Frequency (%)	Difference	Common Text Frequency (%)	Medical Text Frequency (%)	Difference
火	2.20	0.52	1.68	2.20	2.07	0.13
心	1.91	2.01	0.10	1.91	2.17	0.26

Table 4.14: Primary Radical Frequency of 火 (fire) and 心 (heart) in Simplified Chinese and Traditional Chinese

It can be found that the difference of the frequency of primary radical 火 (fire) between the common text and the medical text changes from 1.68 in Simplified Chinese to 0.13 in Traditional Chinese, which makes the weight of 火 (fire) higher in Simplified Chinese and thus a higher RC-value of 心烦 in Simplified Chinese. In Simplified Chinese, many characters whose primary radical is 火 (fire) and have little relationship with fire change their primary radical, such as 乌 (dark, raven) and 烏 (dark, raven), 无 (none) and 無 (none), 为 (for) and 爲 (for). In all three pairs of examples, the first one is in Simplified Chinese and the second one is in Traditional Chinese. Among the examples, 無 (none) and 爲 (for) are both high-frequency characters in common text, which makes the difference of frequency of 火 (fire) much greater in Simplified Chinese than that in Traditional Chinese.

4.5 Summary

This chapter aims to use the primary radical feature in the terminology extraction task on Traditional Chinese Medicine text. The previous work on Chinese Terminology Extraction has been reviewed. Due to the lack of an annotated corpus, the traditional method using statistical information is focused. With the exploration of common text and medical text in Classical Chinese, an advanced version of C-value is proposed, RC-value.

RC-value uses the difference of the frequency of primary radicals in common texts and specified domain texts as an extra weight to filter the term candidates. With the simple experiment on two different test data, RC-value shows the better performance on both sets of test data. Compared to other statistical methods, RC-value not only use the characters themselves but also the semantic information of the characters. So RC-value should work well in the domain that contains more characters whose primary radicals represent their meanings. Based on the discussion in Chapter 2, the primary radicals of both phono-semantic characters and compound ideographic characters contain much semantic information of the characters. Compared to the amount of the phono-semantic characters, the amount of the compound ideographic characters could be ignored. It suggests that RC-value will work well in the domain where phono-semantic characters occur frequently.

The experiment for comprising the performance on Simplified Chinese and Traditional Chinese is also applied. The result of the RC-value in Simplified Chinese and Traditional Chinese should be the same as most of the primary radicals within the terms are not changed in the simplification of the characters. However, due to the limitation of the simplification tool, some radicals are wrongly changed. The actual result between Simplified Chinese and Traditional Chinese

shows a slight difference.

Chapter 5

Conclusion

5.1 Objectives Achieved

The six objectives listed before research have been achieved.

1. Radical features and pronunciation features of Chinese characters are reviewed and discussed, and six categories of Chinese characters are concluded from the ancient dictionary.

2. A method is proposed to identify the phono-semantic character and the method is tested on the ancient dictionary.

3. Radical features are applied to Conditional Random Fields, and the result on clinical data shows that they cannot improve the performance. Three embedding models using radical features are then proposed for deep learning. The result shows that radical features can improve the performance of named entity recognition.

4. An advanced version of C-value, RC-value, is proposed for Terminology Extraction, using the primary radical frequency difference between the common text and the specified domain text. The experiments on two different

sets of common text and medical text are made, and RC-value gets a better result.

5.2 Research Review and Contribution

The research focuses on the use of graphical features in the clinical text mining tasks in Chinese.

In Chapter 2, the graphical and phonetic features of Chinese characters are discussed. Based on the existing researches and literature, most of the Chinese characters are phono-semantic characters, which have primary radicals for their meanings and phonetic radicals for their pronunciations. However, none of them has given an exact percentage of phono-semantic characters among all characters or a method to identify phono-semantic characters by machines.

Based on the concept of phono-semantic characters, a simple algorithm using Ideographic Description Sequence (IDS) is proposed. The proposed algorithm is used to check a given character is a phono-semantic character or not. Thus the algorithm can be used in other algorithms for other purposes, such as calculation of the percentage of the phono-semantic characters among the documents. It is useful in not only the text mining or natural language processing domains but also some other domains related to phono-semantic characters like linguistics.

In Chapter 3, the use of the radical features in the machine learning method in text mining is discussed. Although the use of the graphical features is not the main focus of the current research on text mining in Chinese, there are still some attempts to use them. However, the existing methods are all based on the hypothesis that all parts of all the Chinese characters are useful and suggest the meanings of the characters, no matter which graphical feature is used. The hypothesis is not true based on the research in Chapter 2.

So based on the fact that most of the Chinese characters are phono-semantic characters, a novel embedding model that using the primary radical and the pinyin is proposed. Compared to the embedding model using all radicals of the character, the novel model has a simpler structure that only uses the primary radical of a character, and the other radicals will be represented by the pinyin of it. In a domain like biomedicine where more phono-semantic characters exist than the common domain, the model works better because it can well represent the structure of the phono-semantic characters. As the model proposed is an embedding model, so it could be applied to many text mining tasks in deep learning such as named entity recognition and sentiment analysis.

In Chapter 4, the use of the radical features in the statistical method in text mining is discussed. In the statistical methods, the radicals have not been focused because these methods focus more on the words or characters themselves but not their features. However, in the terms of medicine, phono-semantic characters occur more, among them, some of the primary radicals such as 疒 (sickness) appear frequently. In the terminology extraction of the medical domain, the radical features should be used.

So based on the different distributions of the primary radicals in common text and medical text, RC-value is proposed. RC-value is an improved version of the C-value which uses the primary radical weights of the candidate words. By using the primary radical weight, if one of the characters among the candidate word has a primary radical that occur more frequently in the medical domain than that in the common domain, the candidate word will have a higher RC-value. So compared to other methods that not using radical features, RC-value captures the semantic information of the word by using the primary radical. As a statistical method, RC-value can catch the semantic information easily, so it works better than other statistical methods.

Because both the embedding model in Chapter 3 and RC-value in Chapter 4 are based on the structure of phono-semantic characters, the proposed methods can not only be used in the biomedical or traditional medical domain. In the other domain where phono-semantic characters occur more frequently, such as the chemical or toponymy domain, these proposed methods should also work better than the existing methods.

In conclusion, the research shows the usefulness of using the radical features in medical text mining tasks in Chinese by proposing different methods using the radical features in machine learning and statistical methods based on the structure of the phono-semantic characters. It suggests that the future research on Chinese text mining should focus on the radicals features of Chinese characters.

5.3 Limitations and Future Work

The method proposed for identifying the phono-semantic characters gets a good result on *Shuowen Jiezi*. However, when applying the method on CCKS data, the percentage of the phono-semantic characters is still far from the expected. More exploration on the phono-semantic characters percentage in both Classical Chinese and Modern Chinese should be made to make sure that phono-semantic characters are the majority in both Classical Chinese and Modern Chinese.

The models proposed for the deep learning approach of Named Entity Recognition use two features only. It is because that the composition of more features using special composition methods such as sum or linear method will extremely increase the training time of the models. Some other possible method should be proposed for the composition of more features.

The IDS used in the method of phono-semantic characters shows the structures of all Chinese characters, and this should be definitely useful in the representation of a Chinese character. There should be some experiments to find out if the IDS could be used for character embedding.

The RC-value uses only primary radical features of Chinese characters; some exploration should be also made on other radicals, such as phonetic radicals (or its alternative feature, pinyin) to determine whether they can be used for terminology extraction as well. The experiment on Modern Chinese is also necessary.

The research only focuses on Named Entity Recognition and Terminology Extraction tasks. But radical features should be useful in some other tasks, such as Sentiment Analysis and Relationship Extraction. More research on other tasks should be done in the future.

Bibliography

[AAB+12] Julie D Allen, Deborah Anderson, Joe Becker, Richard Cook, Mark Davis, Peter Edberg, Michael Everson, Asmus Freytag, Laurentiu Iancu, Richard Ishida, et al. *The Unicode Standard*, volume 8. Citeseer, 2012.

[Ana94] Sophia Ananiadou. A methodology for automatic term recognition. In *COLING 1994 Volume 2: The 15th International Conference on Computational Linguistics*, 1994.

[BGJM16] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*, 2016.

[BJM83] Lalit R Bahl, Frederick Jelinek, and Robert L Mercer. A maximum likelihood approach to continuous speech recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (2):179–190, 1983.

[Bol94] William G Boltz. *The origin and early development of the Chinese writing system*, volume 78. Eisenbrauns, 1994.

[Bre17] James Breen. *Extraction of neologisms from Japanese corpora.* PhD thesis, 2017.

[CL00] Zheng Chen and Kai-Fu Lee. A new statistical approach to chinese pinyin input. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 241–247, 2000.

[CLZL18] Shaosheng Cao, Wei Lu, Jun Zhou, and Xiaolong Li. cw2vec: Learning chinese word embeddings with stroke n-gram information. 2018.

[Cov99] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

[CSZ11] Yidong Chen, Xiaodong Shi, and Changle Zhou. A simplified-traditional chinese character conversion model based on log-linear models. In *2011 International Conference on Asian Language Processing*, pages 3–6. IEEE, 2011.

[CV95] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[CWB+11] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(Aug):2493–2537, 2011.

[CYL20] Hong-You Chen, Sz-Han Yu, and Shou-De Lin. Glyph2vec: Learning chinese out-of-vocabulary word embedding from glyphs. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2865–2871, 2020.

[CZI06] Wenliang Chen, Yujie Zhang, and Hitoshi Isahara. Chinese named entity recognition with conditional random fields. In *Proceedings*

of the Fifth SIGHAN Workshop on Chinese Language Processing, pages 118–121. Association for Computational Linguistics, 2006.

[DLY+16] Liping Du, Xiaoge Li, Gen Yu, Chunli Liu, and Yu Liu. 基于互信息改进算法的新词发现对中文分词系统改进. 北京大学学报 *(*自然科学版*)*, 52(1):35–40, 2016.

[DZZ+16] Chuanhai Dong, Jiajun Zhang, Chengqing Zong, Masanori Hattori, and Hui Di. Character-based lstm-crf with radical-level features for chinese named entity recognition. In *International Conference on Computer Processing of Oriental Languages*, pages 239–250. Springer, 2016.

[Eme05] Thomas Emerson. The second international chinese word segmentation bakeoff. In *Proceedings of the fourth SIGHAN workshop on Chinese language Processing*, 2005.

[FAM00] Katerina Frantzi, Sophia Ananiadou, and Hideki Mima. Automatic recognition of multi-word terms:. the c-value/nc-value method. *International journal on digital libraries*, 3(2):115–130, 2000.

[FSZ05] Yuanyong Feng, Le Sun, and Julin Zhang. Early results for chinese named entity recognition using conditional random fields model, hmm and maximum entropy. In *2005 International Conference on Natural Language Processing and Knowledge Engineering*, pages 549–552. IEEE, 2005.

[GAoQSQ92] Inspection General Administration of Quality Supervision and Quarantine. State standard of peoples' republic of china gb13715,

contemporary chinese language word segmentation specification for information processing, 1992.

[GSC99] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. 1999.

[HA12] Hongqi Han and Xiaomi An. C-value 值和 unithood 指标结合的中文科技术语抽取. 图书情报工作, (19):85–89, 2012.

[HGA+98] Kevin Humphreys, Robert Gaizauskas, Saliha Azzam, Charles Huyck, Brian Mitchell, Hamish Cunningham, and Yorick Wilks. University of sheffield: Description of the lasie-ii system as used for muc-7. In *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998*, 1998.

[HGL03] Changning Huang, Jianfeng Gao, and Mu Li. 对自动分词的反思 (in chinese). 语言计算与基于内容的文本处理, 2003.

[Hoo13] Rumjahn Hoosain. *Psycholinguistic implications for linguistic relativity: A case study of Chinese*. Psychology Press, 2013.

[HPWW02] Liang Huang, Yinan Peng, Huan Wang, and Zhenyu Wu. Statistical part-of-speech tagging for classical chinese. In *International Conference on Text, Speech and Dialogue*, pages 115–122. Springer, 2002.

[HWC13] Aaron L-F Han, Derek F Wong, and Lidia S Chao. Chinese named entity recognition with conditional random fields in the light of chinese characteristics. In *Intelligent Information Systems Symposium*, pages 57–68. Springer, 2013.

[HXY15] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[HZJ13] Apei Hu, Jing Zhang, and Liu Junli. 基于改进 c-value 方法的中文术语抽取 (in chinese). 数据分析与知识发现, 29(2):24–29, 2013.

[JCGL14] Peng Jin, Xingyuan Chen, Zhaoyi Guo, and Pengyuan Liu. Integrating pinyin to improve spelling errors detection for chinese language. In *Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 01*, pages 455–458. IEEE Computer Society, 2014.

[Jia03] Hongjie Jia. 试论汉字简化对部首体系的冲击 (in chinese). 汉字书同文研究, 2003.

[JWZ11] Zhenfei Ju, Jian Wang, and Fei Zhu. Named entity recognition from biomedical text using svm. In *2011 5th international conference on bioinformatics and biomedical engineering*, pages 1–4. IEEE, 2011.

[KH98] George R Krupka and Kevin Hausman. Isoquest inc.: Description of the netowl™ extractor system as used for muc-7. In *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998*, 1998.

[KMOT02] Jun'ichi Kazama, Takaki Makino, Yoshihiro Ohta, and Jun'ichi Tsujii. Tuning support vector machines for biomedical named entity recognition. In *Proceedings of the ACL-02 workshop on*

Natural language processing in the biomedical domain-Volume 3, pages 1–8. Association for Computational Linguistics, 2002.

[lan] Summary by language size | ethnologue. `https://www.ethnologue.com/statistics/size`. Accessed on 01-06-2019.

[LCJ+07] Yumei Li, Xiao Chen, Zixia Jiang, Jiangyan Yi, Guangjin Jin, and Changning Huang. 分词规范亟需补充的三方面内容 (in chinese). 中文信息学报, 21(5):1–7, 2007.

[Li05] Shuai Li. 部分肉部字演变考察 (in chinese). 广西民族学院学报*:* 哲学社会科学版, 2005.

[Liu05] Jianyu Liu. 记号字, 半记号字及其在现代汉字中基本情况探讨 *(in Chinese)*. PhD thesis, 2005.

[LLSL15] Yanran Li, Wenjie Li, Fei Sun, and Sujian Li. Component-enhanced chinese character embeddings. *arXiv preprint arXiv:1508.06669*, 2015.

[LMP01] John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

[LRS83] Stephen E Levinson, Lawrence R Rabiner, and Man Mohan Sondhi. An introduction to the application of the theory of probabilistic functions of a markov process to automatic speech recognition. *Bell System Technical Journal*, 62(4):1035–1074, 1983.

[LSM13] Thang Luong, Richard Socher, and Christopher D Manning. Better word representations with recursive neural networks for morphology. In *CoNLL*, pages 104–113, 2013.

[LW94] Yulong Leng and Yixin Wei. *Zhonghua Zihai*. 中华书局, 1994.

[LZ08] Bo Lin and Jun Zhang. A novel statistical chinese language model and its application in pinyin-to-character conversion. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 1433–1434. ACM, 2008.

[LZZ10] Yinhong Liang, Wenjing Zhang, and Youcheng Zhang. *C* 值和互信息相结合的术语抽取. PhD thesis, 2010.

[MCCD13] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[MCF+98] Scott Miller, Michael Crystal, Heidi Fox, Lance Ramshaw, Richard Schwartz, Rebecca Stone, Ralph Weischedel, et al. Bbn: Description of the sift system as used for muc-7. In *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998*, 1998.

[MH16] Xuezhe Ma and Eduard Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*, 2016.

[MHG+18] Adam L Meyers, Yifan He, Zachary Glass, John Ortega, Shasha Liao, Angus Grieve-Smith, Ralph Grishman, and Olga Babko-Malaya. The termolator: terminology recognition based on chunking, statistical and search-based scores. *Frontiers in Research Metrics and Analytics*, 3:19, 2018.

[MM90] David M Magerman and Mitchell P Marcus. Parsing a natural language using mutual information statistics. In *AAAI*, volume 90, pages 984–989, 1990.

[MMS99] Christopher D Manning, Christopher D Manning, and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT press, 1999.

[MS16] Kaixu Zhang Zhipeng Guo Zhiyuan Liu Maosong Sun, Xinxiong Chen. Thulac: An efficient lexical analyzer for chinese. 2016.

[NV04] Roberto Navigli and Paola Velardi. Learning domain ontologies from document warehouses and dedicated web sites. *Computational Linguistics*, 30(2):151–179, 2004.

[NZ09] Adeline Nazarenko and Haifa Zargayouna. Evaluating term extraction. In *International Conference Recent Advances in Natural Language Processing (RANLP'09)*, pages 299–304, 2009.

[O+07] World Health Organization et al. Who international standard terminologies on traditional medicine in the western pacific region. 2007.

[oEotPRoC15] Ministry of Education of the People's Republic of China. Scheme for the chinese phonetic alphabet, 2015.

[PNL01] Silvia Pavel, Diane Nolet, and Christine Leonhardt. *Précis de terminologie = Handbook of terminology*. Ministre des Travaux publics et Services, 2001.

[RG17] Nils Reimers and Iryna Gurevych. Optimal hyperparameters for

deep lstm-networks for sequence labeling tasks. *arXiv preprint arXiv:1707.06799*, 2017.

[Rob04] Stephen Robertson. Understanding inverse document frequency: on theoretical arguments for idf. *Journal of documentation*, 60(5):503–520, 2004.

[SCH11] Xiaodong Shi, Yidong Chen, and Xiuping Huang. Key problems in conversion from simplified to traditional chinese characters. In *International Conference on Asian Language Processing*, 2011.

[SE03] Richard Sproat and Thomas Emerson. The first international chinese word segmentation bakeoff. In *Proceedings of the second SIGHAN workshop on Chinese language processing-Volume 17*, pages 133–143. Association for Computational Linguistics, 2003.

[SGSC96] Richard Sproat, William Gale, Chilin Shih, and Nancy Chang. A stochastic finite-state word-segmentation algorithm for chinese. *Computational linguistics*, 22(3):377–404, 1996.

[SHTN17] Yan Shao, Christian Hardmeier, Jörg Tiedemann, and Joakim Nivre. Character-based joint segmentation and pos tagging for chinese using bidirectional rnn-crf. *arXiv preprint arXiv:1704.01314*, 2017.

[SJ72] Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972.

[SJ04] Karen Spärck Jones. Idf term weighting and ir research lessons. *Journal of documentation*, 60(5):521–523, 2004.

[SLY+14] Yaming Sun, Lei Lin, Nan Yang, Zhenzhou Ji, and Xiaolong Wang. Radical-enhanced chinese character embedding. In *International Conference on Neural Information Processing*, pages 279–286. Springer, 2014.

[Son97] Rou Song. 关于分词规范的探讨 (in chinese). 语言文字应用, (3):113–114, 1997.

[SZY+15] Xinlei Shi, Junjie Zhai, Xudong Yang, Zehua Xie, and Chao Liu. Radical embedding: Delving deeper to chinese radicals. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 594–598, 2015.

[TH03] Takashi Tomokiyo and Matthew Hurst. A language model approach to keyphrase extraction. In *Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment*, pages 33–40, 2003.

[TJM05] Huihsin Tseng, Daniel Jurafsky, and Christopher Manning. Morphological features help pos tagging of unknown words across language varieties. In *Proceedings of the fourth SIGHAN workshop on Chinese language processing*, pages 32–39, 2005.

[TRB10] Joseph Turian, Lev Ratinov, and Yoshua Bengio. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics, 2010.

[TSL87] Ho-Ping Tseng, M Sabin, and E Lee. Fuzzy vector quantazation applied to hidden markov modeling. In *ICASSP'87. IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 12, pages 641–644. IEEE, 1987.

[Tu11] Youyou Tu. The discovery of artemisinin (qinghaosu) and gifts from chinese medicine. *Nature medicine*, 17(10):1217, 2011.

[Wan11] Qun Wang. 论许慎"转注"概念的真实用意 (in chinese). 汉字文化, (3):42–44, 2011.

[Wan12] Xinkai Wang. *Chinese-English Cross-Lingual Information Retrieval in Biomedicine Using Ontology-Based Query Expansion.* PhD thesis, 2012.

[WCL09] Lijie Wang, Wanxiang Che, and Ting Liu. An svmtool-based chinese pos tagger. *Journal of Chinese Information Processing*, 23(4):16–22, 2009.

[WMH+19] Wei Wu, Yuxian Meng, Qinghong Han, Muyu Li, Xiaoya Li, Jie Mei, Ping Nie, Xiaofei Sun, and Jiwei Li. Glyce: Glyph-vectors for chinese character representations. *arXiv preprint arXiv:1901.10125*, 2019.

[WTTA12] Xinkai Wang, Paul Thompson, Jun'ichi Tsujii, and Sophia Ananiadou. Biomedical chinese-english clir using an extended cmesh resource to expand queries. *Dementia*, 10(228.140):380, 2012.

[WW08] Xiaoming Wang and Linmei Wei. Key problems of conversion between simplified chinese and traditional chinese. *5TH CDF* 研討會數位社群雙效 *(CD2E)*, 2008.

[WZZ20] Shirui Wang, Wenan Zhou, and Qiang Zhou. Radical and stroke-enhanced chinese word embeddings based on neural networks. *Neural Processing Letters*, pages 1–13, 2020.

[XLL+13] Dan Xiong, Qin Lu, Fengju Lo, Dingxu Shi, Tin-shing Chiu, and Wanyin Li. *Chinese Lexical Semantics: 13th Workshop, CLSW 2012, Wuhan, China, July 6-8, 2012, Revised Selected Papers*, chapter Specification for Segmentation and Named Entity Annotation of Chinese Classics in the Ming and Qing Dynasties, pages 280–293. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[XQYL20] Zongyang Xiong, Ke Qin, Haobo Yang, and Guangchun Luo. Learning chinese word representation better by cascade morphological n-gram. *Neural Computing and Applications*, pages 1–12, 2020.

[XS03] Nianwen Xue and Libin Shen. Chinese word segmentation as lmr tagging. In *Proceedings of the second SIGHAN workshop on Chinese language processing-Volume 17*, pages 176–179. Association for Computational Linguistics, 2003.

[Xury] Shen Xu. *Shouwen Jiezi*. 2nd century.

[Yip00] Po-Ching Yip. *The Chinese lexicon: A comprehensive survey*. Psychology Press, 2000.

[YJXS17] Jinxing Yu, Xun Jian, Hao Xin, and Yangqiu Song. Joint embeddings of chinese words, characters, and fine-grained subcharacter components. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 286–291, 2017.

[Zho03] Youguang Zhou. *The historical evolution of Chinese languages and scripts*, volume 8. Ohio State Univ Foreign Language, 2003.

[ZM17] Jinyi Zhang and Tadahiro Matsumoto. Improving character-level japanese-chinese neural machine translation with radicals as an additional input feature. In *2017 International Conference on Asian Language Processing (IALP)*, pages 172–175. IEEE, 2017.

[ZS02] GuoDong Zhou and Jian Su. Named entity recognition using an hmm-based chunk tagger. In *proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 473–480. Association for Computational Linguistics, 2002.

[ZSFH09] Hai-Jun Zhang, Shu-Min Shi, Chong Feng, and He-Yan Huang. A method of part-of-speech guessing of chinese unknown words based on combined features. In *2009 International Conference on Machine Learning and Cybernetics*, volume 1, pages 328–332. IEEE, 2009.

[ZSZ+04] Jie Zhang, Dan Shen, Guodong Zhou, Jian Su, and Chew-Lim Tan. Enhancing hmm-based biomedical named entity recognition by studying special phenomena. *Journal of biomedical informatics*, 37(6):411–422, 2004.

[ZYT11] Wen Zhang, Taketoshi Yoshida, and Xijin Tang. A comparative study of tf* idf, lsi and multi-words for text classification. *Expert Systems with Applications*, 38(3):2758–2765, 2011.

[ZZL15] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657, 2015.