FLINDERS
UNIVERSITY
ADELAIDE
AUSTRALIA

Word Segmentation and Ambiguity in English and Chinese NLP & IR

by

黄金虎
Jin Hu Huang, *B.Eng.(Comp.Sc)Grad.Dip.(Soft.Eng.)*
School of Computer, Engineering and Mathematics,
Faculty of Science and Engineering

August 10, 2011

A thesis presented to the
Flinders University of South Australia
in total fulfillment of the requirements for the degree of
Doctor of Philosophy
in
Computer Science

Adelaide, South Australia, 2012

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Statistical language learning is the problem of applying machine learning technique to extracting useful information from large corpus. It is important in both statistical natural language processing and information retrieval. In this thesis, we attempt to build some statistical language learning and modeling algorithms to solve some problems in both English and Chinese natural language processing. These problems include context sensitive spelling correction in English, adaptive language modeling for Chinese Pinyin input, Chinese word segmentation and classification, and Chinese information retrieval.

Context sensitive spelling correction is a word disambiguation process to identify the word-choice errors in text. It corrects real-word spelling errors made by users when another word was intended. We build large scale confused word sets based on keyboard adjacency. Then we collect the statistics based on the surrounding words using affix information and the most frequent functional words. We store the contexts significant enough to make a choice among the confused words and apply this contextual knowledge to detect and correct the real-word errors. In our experiments we explore the performance of auto-correction under conditions where significance and probability are set by the user. The technique we developed in this thesis can be used to resolve lexical ambiguity in the syntactic sense.

Chinese Pinyin-to-character conversion is another task of word disambiguation. Chinese character can't be entered by keyboard directly. Pinyin is the phonetic transcription of Chinese characters using the Roman alphabet. The process of Pinyin-to-character conversion, similar to speech recognition, is to decode the sequence of Pinyin syllables into corresponding characters based on statistical

n-gram language models. The performance of Chinese Pinyin-to-Character conversion is severely affected when the characteristics of the training and conversion data differs. As natural language is highly variable and uncertain, it is impossible to build a complete and general language model to suit all the tasks. The traditional adaptive maximum a posteriori (MAP) models mix the task independent model with task dependent model using a mixture coefficient but we never can predict what style of language users have and what new domain will appear. We present a statistical error-driven adaptive n-gram language model to Pinyin-to-character conversion. This n-gram model can be incrementally adapted during Pinyin-to-Character converting time. We use a conversion error function to select what kind of data to adapt the model. The adaptive model significantly improves Pinyin-to-Character conversion rate.

Most Asian languages such as Chinese and Japanese are written without natural delimiters, so word segmentation is an essential first step in Asian language processing. Processing at higher levels will be impossible if there is no effective word segmentation. Chinese word segmentation is a basic research issue on Chinese language processing tasks such as information extraction, information retrieval, machine translation, text classification, automatic text summarization, speech recognition, text-to-speech, natural language understanding, and so on. This thesis presents a purely statistical approach to segment Chinese sequences into words based on contextual entropy on both sides of a bi-gram. It is used to capture the dependency with the left and right contexts in which a bi-gram occurs. Our approach tries to segment text by finding the word boundaries instead of the words. Although developed for Chinese it is language independent and easy to adapt to other languages, and it is particularly robust and effective for Chinese word segmentation.

Traditionally Chinese words are not regarded being inflected with respect to tense, case, person and number, this information is captured by separate words that attach as clitics rather than affixes. Telling the part-of-speech of a word is not straightforward. In this thesis we classify Chinese words according to the substitutability of linguistic entities from the same class. We merge words/classes

together based on contextual information and class overlapping.

Traditional information retrieval systems for European languages such as English use words as indexing units and thus cannot apply directly to Asian languages such as Chinese and Japanese due to lack of word delimiters. A preprocessing stage called segmentation has to be performed to determine the boundaries of words before traditional IR approaches based on words can be adapted to Chinese language. Different segmentation approaches, N-grams based or word based, have their own advantages and disadvantages. No conclusion has been reached among different researchers as to which segmentation approach is better or more appropriate for the purpose of IR even on standard Chinese TREC corpus. In this thesis we investigate the impact of these two segmentation approaches on Chinese information retrieval using standard Chinese TREC 5 & 6 corpus. We analyze why some approaches may work effectively in some queries but work poorly in other queries. This analysis is of theoretical and practical importance to Chinese information retrieval.

CERTIFICATION

I certify that this thesis does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any university; and that to the best of my knowledge and belief it does not contain any material previously published or written by another person except where due reference is made in the text.

As requested under Clause 14 of Appendix D of the *Flinders University Research Higher Degree Student Information Manual* I hereby agree to waive the conditions referred to in Clause 13(b) and (c), and thus

- Flinders University may lend this thesis to other institutions or individuals for the purpose of scholarly research;
- Flinders University may reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Signed Dated

Jin Hu Huang

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude to my supervisor Prof. David Powers for bringing me into natural language processing research field. I thank him for his guidance, tolerance, patience, support and encouragement over the past 10 years.

I am grateful to the members of AI group and KDM group for their discussion, friendship and support. I would like to thank the school for the financial support to attend conference in Beijing and Singapore.

I thank my family for their eternal understanding, patience and most of all, support. An enormous thanks go to my wife, Li Ping, for her love and support. Going overseas, becoming a mother and being a student were tough, but she has done so well and even found a job in government. So many times I promised to her that I would take time to finish this year. Finally I will make it in 2011. I would like to thank my parents-in-law and sister-in-law, who came over from China for the tremendous help. Feel sorry to my lovely daughter and son, Lydia and Jurgen. Sometime they are my excuses not finishing on time.

Specially thanks to Mr Eric Robins who supported me to come to study at Flinders at beginning. I would also like to thank Mr Jurgen Kracht who provided accommodation for me. I really enjoyed staying with him and his two cats, Heihei and Mushi in early my study.

I would like to thank the members of Somerton Baptist Church and Flinders Overseas Christian Fellowship for their sharing, prayers and friendship. Specially Kenneth Lim, Barry Wee, Merrilyn Teague, Dahlis and Robert Willcock, John and Marrilyn.

Due to some personnel reasons, I studied part time since 2003 and intermitted my study several times during 2005-1010.

This work was carried out during the tenure of International Postgraduate Research Scholarship and Flinders University Research Scholarship, and also received support under an Australia Research Council Discovery grant in the early stages.

To my parents and wife!

Jin Hu Huang

July 2011

Adelaide.

PREFACE

Almost all the results presented in this thesis have already been published in national and international conferences.

Chapter 2 is based on (Huang & Powers 2001)

Huang, J. H. & Powers, D. (2001), Large scale experiments on correction of confused words, in 'Australiasian Computer Science Conference', IEEE Computer Society Press, Gold Coast, Queensland.

Chapter 4 is based on (Huang & Powers 2004)

Huang, J. H.& Powers, D. (2004), Adaptive compression-based approach for Chinese Pinyin input, in 'Third ACL SIGHAN Workshop on Chinese Processing'.

Chapter 5 is based on (Huang & Powers 2011)

Huang, J. H. & Powers, D. (2011), Error-driven adaptive language modeling for Pinyin-to-character conversion, in 'International Conference on Asian Language Processing (IALP2011)', Nov 15-17, 2011, Penang, Malaysia.

Chapter 7 is based on (Huang & Powers 2003)

Huang, J. H. & Powers, D. (2003), Chinese word segmentation based on contextual entropy, in '17th Pacific Asia Conference on Language, Information and Computation', Singapore.

Chapter 8 is based on (Huang & Powers 2002)

Huang, J. H. & Powers, D. (2002), Unsupervised Chinese word segmentation and classification, in 'First Student Workshop in Computational Linguistics', Beijing, China.

Chapter 10 is based on (Huang & Powers 2008)

Huang, J. H. & Powers, D. (2008), Suffix-tree-based approach for Chinese

information retrieval, in 'International Conference on Intelligent Systems Design and Applications (ISDA)'.

Rather than including the published paper as is permitted under the PhD rules, I have sought to integrate the material into the thesis in a cohesive way whilst achieving a balance between chapter that stand alone and avoidance of redundancy in relation to literature review.

1. INTRODUCTION

This chapter introduces the issues in spelling checking and Chinese language processing that motivate me to do this research. It provides some basic knowledge about Chinese language and shows a number of characteristics that make Chinese language processing particularly challenging.

1.1 Context Sensitive Spelling Correction

Words are a basic unit to all natural language processing. But the physical signals that embody word, whether realized in electronic, acoustic, optical, or other forms frequently arrive at their destinations in imperfect and ambiguous condition. Before any text understanding, speech recognition, machine translation, computer-aided learning, optical character, or handwriting recognition system can achieve a marketable performance level, it must tackle the pervasive problem of dealing with noisy, ill-fitting, novel and otherwise unknown words. Assuming phonetic or orthographic representation based on speech or optical recognition, Pinyin input, or English typing, there are very similar stages we must go through to end up with a clean word based text we can use for further processing or applications. For English this can be characterized as spelling correction, and we set the scene using this as our initial application.

When we are writing documents, we often use spelling checkers to find the errors in the texts. Some spelling checkers even mark the errors with different colors to alert users. Many spelling checkers concentrate on non-word errors. This kind of errors can very easily be discovered by looking up a dictionary. For example, when I type "teh" in the editor, it is marked with red color and

advise me that there is an error. Then spelling checkers will provide a list of suggested corrections according to the string edit distance (Gusfield 1997). A possible correction would be "the". Even when a text is free of non-word errors, there is no guarantee that the text is error-free. There are several types of errors in which correct words are used in the incorrect contexts. For example, "from" may easily by mistyped as "form". This kind of errors are much harder to detect than non-word errors.

Fixing these kinds of errors requires analyzing the contextual information and is not handled by conventional spell-check programs although Microsoft Office 2007 started to incorporate with context-sensitive spell checking. The task of fixing these spelling errors that happen to result in valid words is called context-sensitive spelling correction. Context sensitive spelling correction is a word disambiguation process to identify the word-choice errors in text. It corrects real-word spelling errors made by users when another word was intended. Most NLP systems resolve such ambiguity with the help of a large corpus of text, even from world wide web (Bergsma, Lin & Goebel 2009, Islam & Inkpen 2009). Statistical machine learning techniques(Yarowsky 1994, Golding 1995, Golding & Roth 1996, Powers 1997*a*, Mangu & Brill 1997, Wilcox-O'hearn, Hirst & Budanitsky 2008, Stehouwer & van Zaanen 2009*a*), with the capability of automatically acquiring knowledge from corpora, are especially appropriate to solve these problems of ambiguity and ill-formedness.

1.2 Chinese Pinyin Input

Since the Chinese language uses a logographic script. There are more characters, or glyphs, than there are keys on a standard computer keyboard. Inputting and processing Chinese language text on a computer can be very difficult. A variety of keyboard input methods have been designed to allow the input of Chinese characters using standard keyboards.

Keyboard input methods can be classified in three main types:

- by encoding
- by pronunciation (Pinyin)
- by structure of the characters

One common method available today for inputting Chinese language text into a computer system is one using phonetic input, e.g. Pinyin. Pinyin uses Roman characters and has a vocabulary listed in the form of multiple syllable words. However, the Pinyin input method results a homonym problem in Chinese language processing.

In linguistics, homonyms are words that share the same spelling and the same pronunciation but have different meanings. For example, in English the word (homograph) "rose" can mean at least two different things, i.e., the flower "rose" or the past tense for the verb "rise". The words "see" and "sea" have identical sounds but they are spelled differently and have different meanings. They are homophones. In Chinese, the syllables written in Pinyin such as "yu" are homonyms but the characters such as 雨(rain) and 鱼(fish) that pronounce "yu" are homophones. There are two different concepts. One the other hand, homographs, which mean words that have same spelling but have more than one sound and meaning, also exist in Chinese. For example, the character 地 can pronounce either as di or de, meaning ground and the adverbial marker respectively.

Compared with English, Chinese character has a much smaller number of possible syllables due to its simpler syllable structure, which does not allow consonant clusters such as "pl" in "play", and only two final nasal consonants "n" and "ng". The total number of possible syllables is much smaller than that in English. There are only some 405 syllables in Mandarin without the tones. When you add the tones, the number is about 1100 as not all syllables have four tones. But this is still a far less than English, which has over 80,000 possible syllables. This causes many Chinese characters share the same syllable. Extensive homophony means most sound/syllables map to multiple Chinese characters. For the native speaker,

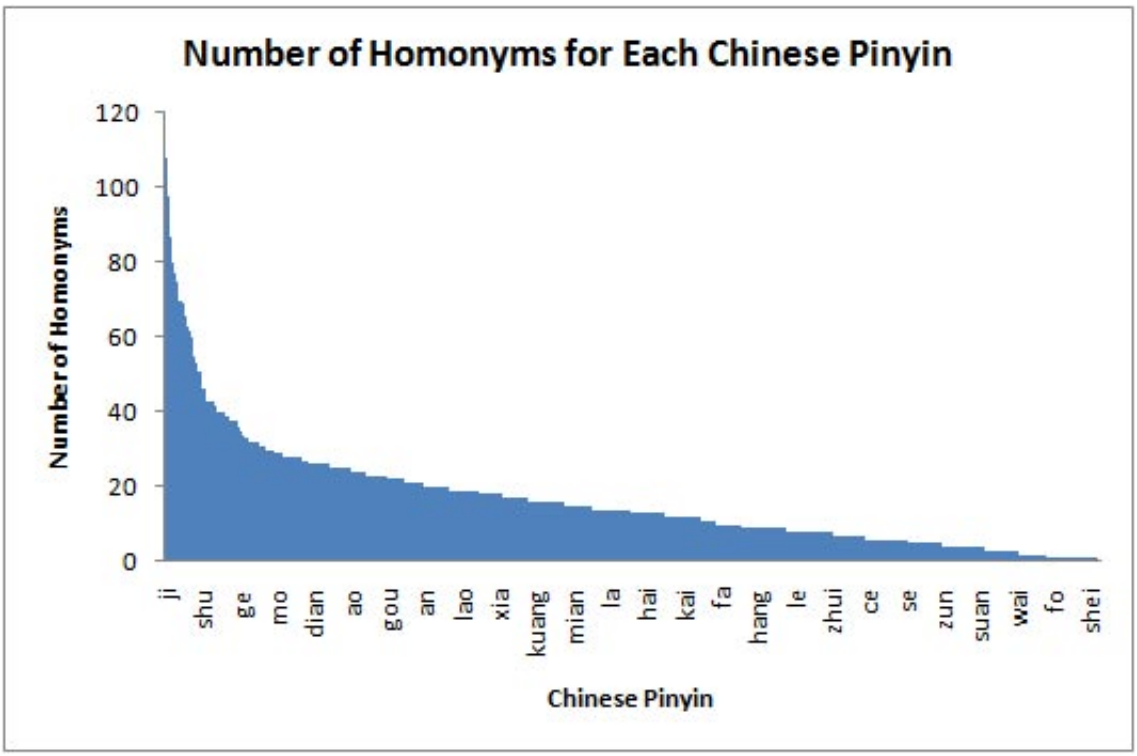

Fig. 1.1: Number of Homonyms for Each Pinyin

we are not aware of the tone when we type Pinyin. If we consider the tone it actually slows down typing greatly. There are about average 17 homonyms in each Pinyin without tones in GB2312-80 coding, with even more homonyms in current GB18030 as more characters are included. This creates ambiguities when translating the phonetic syllables into characters. Table 1.1 show the distribution of the number of homonyms for Pinyins. Pinyin "ji" has most homonyms of 108 and "shei" has only one.

The famous linguist Chao (Chao 1968) once wrote a mock classical style tale using one single-syllable "shi" to illustrate this phenomenon in early last century. Most people think he wrote it to show his opposition to the romanization of Mandarin Chinese (the use of the Latin alphabet to write Chinese) but he was one of the designers of early form of Pinyin. It is difficult to understand this tale in Pinyin "shi shi shi shi shi shi, shi shi, shi shi shi shi" (first sentence of the 96-character tale) as there are more than 30 homonyms for "shi". This shows automatic Pinyin-to-character conversion very challenge in Chinese natural language processing. The first sentence is written in Chinese as following.

"石(shí stone)室(shì house)诗(shī poem)士(shì scholar)
施(shī Shi,surname)氏(shì surname), 嗜(shí like)狮(shí lion),
誓(shì promise)食(shī eat)十(shí ten)狮(shī lion)"

It translates as "Mr. Shi, a poet who lived in a stone house, liked to eat lions and promised to eat ten lions."

The main problem in Chinese Pinyin-to-character conversion is how to select the right character among its related homophones. The process of Chinese homophone disambiguation can be defined as how to convert sequence of Pinyin into the corresponding sequence of characters correctly. Statistical language model is widely used to solve this problem due to its power and robustness. The Pinyin-to-Chinese-Character conversion is the fundamental and core technique in Chinese Input system, Chinese speech recognition and Chinese information processing.

1.3 Chinese Segmentation

We introduce Chinese word segmentation through the following story.

Once an upon time, there was a rich man who was very mean. He employed a scholar as a private tutor for his son. In order to save money, he signed a contract with the scholar.

wú jī yā yě kě wú yú ruò yě kě wéi qīng cài dòu fǔ bù kě shǎo bù dě xué fèi

“无鸡鸭也可无鱼肉也可唯青菜豆腐不可少不得学费”

We now rewrite the sentence vertically with English gloss.

Character	Meaning	Word Gloss	Meaning
无	Without		
鸡	Chicken		
鸭	Duck		
也	Either, Also		
可	All Right, OK		
无	Without		
鱼	Fish		
肉	Meat		
也	Either, Also		
可	All Right, OK		
唯	Only, alone		
青	Green		
菜	Vegetable	青菜	Green Vegetable
豆	Bean	菜豆	Phaseolus vulgaris (green bean)
腐	Rotten, curd	豆腐	Tofu, Bean curd
不	Not		
可	All Right, Can		
少	Little, Few, Lack		
不	No, Not		
得	Get		
学	Study, Learn		
费	Fee	学费	Tuition Fee

After starting work, the rich man always served the scholar with vegetable and Tofu. The scholar was not happy. They went to court. The judge asked them to explain. The rich man said I did follow the contract.

无鸡鸭也可 无鱼肉也可 唯青菜豆腐不可少 不得学费

Without chicken and duck is also all right. Without fish and meat is also all right. Only vegetable and Tofu could not be lacked. Not getting tuition fee.

The scholar argued it.

无鸡 鸭也可 无鱼 肉也可 唯青菜豆腐不可 少不得学费

Without chicken, duck is also all right. Without fish, meat is also all right. Only vegetable and Tofu are not all right. Could not get less tuition fee.

Readers unfamiliar with Chinese can gain an appreciation of the problem of multiple interpretations from story, which shows two alternative interpretations of the same Chinese character sequence. The text is a joke that relies on the ambiguity of phrasing, and represents a general problem that can arise with word segmentation. The following sentence "唯青菜豆腐不可" can be further segmented into the following three types. Which one is the correct interpretation really depends on the context of the sentence.

((唯(青 菜豆)) (腐(不 可)))

Only green been is not all right to be rotten

((唯((青 菜豆) 腐)) (不 可))

Only green bean curd is not all right

((唯(青菜 豆腐)) (不 可))

Only green vegetable and bean curd are not all right

Similarly the following two sentences in English

- Time flies like an arrow.
- Fruit flies like a banana.

can be considered as the problems of segmentation or chunking (Abney 1991) such as "(Time flies) (like an arrow)" or "(time) (flies like an arrow)" and "(Fruit flies) (like a banana)" or "(Fruit) (flies like a banana)". They cannot be segmented without the semantic information. But this is at the syntactic level. An example at the morphological level is "menswear" vs " men swear". Another phonetic example is given in 3.1. In Chinese these phenomenons are more common as discussed in 6.3.1.

Chinese orthography fails to represent word boundaries. The absence of word boundaries makes it very difficult to segment the text. For example, in a sentence consisting of 10 characters, there will be 9 character boundaries. Since each character boundary could also be a potential location for a word boundary, the 10-character sentence would generate $(2^9) = 512$ different word strings, among which most of the time there is only one correct word string. Word segmentation uncertainty grows exponentially as the length of the sentence grows. The exponential nature of word segmentation ambiguity seems extremely challenging.

Chinese word segmentation is an essential first step for Chinese information processing tasks such as information extraction, information retrieval, machine translation, text classification, automatic text summarization, speech recognition, text-to-speech, natural language understanding, and so on. Although it has been investigated for more than twenty years (Sproat & Shih 2002)(Sproat & Shih 1990)(Sproat, Shih, Gale & Chang 1996)(Xue 2003)(Gao, Li, Wu & Huang 2005)(Huang & Zhao 2007)(Zhao, Huang, Li & Lu 2010)(Dong, Dong & Hao 2010)(Zhao & Kit 2011), it is still a bottleneck for Chinese language processing.

1.4 Chinese Information Retrieval (IR)

In the era of information explosion, finding useful information we really want from the Internet becomes a necessity. The aim of an IR system is to organize and store information, and retrieve the useful information when the user poses a query to the system. Given a larger collection of documents, keywords are matched between a query and the documents to predict the potential relevance of the documents to satisfy the user's information need. Based on this prediction, all documents are ranked in the decreasing order of their predicted relevance. Using such a ranking, documents that are potentially more relevant to query are presented at top ranks to the user (Salton & McGill 1986).

The modern IR systems developed for retrieving Romanized documents are based on words. In written Chinese, no explicit separators are inserted between

written words to indicate boundaries. A Chinese sentence usually can be segmented into many different possible word combinations due to segmentation ambiguities, and it is difficult to decide on the correct combination. Appropriate word segmentation of Chinese text should rely on sophisticated syntactic and semantic analysis on the text. Word segmentation and unknown word identification is a difficult task in natural language processing(Gao et al. 2005)(Huang & Zhao 2007) . This results in two serious problems for Chinese IR. Firstly, there will be not guarantee for a consistent unique tokenization in documents and queries. Secondly, it is difficult to identify unknown words, such as names and locations, which are usually the keywords in queries.

The other approach is to abandon word-based approach and to do all indexing via overlapped character n-grams, regardless of whether particular sequences cross word boundaries or not. A bigram approach is effective for Chinese IR (Nie & Ren 1999, Kwok 1999) as most Chinese words are two characters. This approach is exhaustive and avoids difficulty of segmentation. Unknown words are treated as two-character chunks instead of single characters in word-based approach.

A lot of studies have been done to compare the effectiveness of these approaches (Voorhees & Harman 1996, Wilkinson 1998, Nie & Ren 1999, Nie, Gao, Zhang & Zhou 2000), but the results have not been conclusive. Further researches (Foo & Li 2004, Peng, Huang, Schuurmans & Cercone 2002) on the relationship between word segmentation and Chinese information retrieval show that the segmentation accuracy does not monotonically influence subsequent retrieval performance. It is still not very clear the impact of retrieval model and word segmentation on Chinese IR.

1.5 Thesis Contribution

We use statistical n-gram models and machine learning as the main solutions to address above concerns. N-gram models are effective and robust to tackle problems related to natural language processing.

1. **Context sensitive spelling Correction**

   Context sensitive spelling correction detects and corrects real-word errors in text using contextual information. In our experiments real-word errors are modeled by large confused sets based on keyboard adjacency. To avoid the machine learning problems of high-dimensional feature space, we use the affix information and the most frequent functional words to capture the characterization of linguistic context. We collect n-gram contextual statistics surrounding the confused words from WSJ corpus and apply this knowledge to detect and correct the real-word errors using significance and probability among the confused words. We even found two real errors in the WSJ corpus. We built an interface on Word 97 to explore the performance of auto-detection and auto-correction using different levels of significance and probabilities.

2. **Error-Driven Adaptive Language Modeling for Chinese Pinyin-to-character Conversion**

   Most researches do not consider the distribution of homonyms in Pinyins for the task of Chinese Pinyin-to-character conversion. Some Pinyins have more than 100 homonyms, and others have only one. The Pinyin with more than 100 homonyms maybe need more data to disambiguate the homonyms. The error-driven adaptive language model will detect the conversion errors and adapts to the new data. In other words, it adapts to new data only if it could not make correct conversion. An conversion error function is defined to measure the number of conversion errors produced by the model. It leads the model adapt to new data more effectively. We use the Witten-Bell discounting method to smooth probability estimates to cope with sparse data. The mechanism of the Witten-Bell discounting method, which was developed for the task of text compression, is efficient for prediction and data updating in the running time. Experiments show that the adaptive model greatly reduces perplexity and Pinyin-to-Character conversion error rate in the test data.

3. **Unsupervised Chinese Word Segmentation and Classification**

   We present a unsupervised word segmentation method to break Chinese text into tokens without prior lexical knowledge. This purely statistical approach is based on contextual entropy (branching or boundary entropy) on both sides of a n-gram. Contextual entropy measures the branching factor at the boundaries following n-grams. The branching uncertainty is higher at a boundary between two language units such as words. As most Chinese words are two characters, a bigram model is effective for word segmentation. Various thresholds can be obtained from training on a small segmented data. We segment the text by trying to identify word boundaries instead of words. Although our approach is developed for Chinese it is language independent and easy to adapt to other languages and domains, and it is particularly robust for Chinese word segmentation.

4. **N-grams for Chinese Information Retrieval**

   We investigate the impacts of the n-gram based approach and word based approach on Chinese information retrieval. We compare the performance of different length of n-gram based approaches, traditional dictionary based maximum match approach and statistical segmentation approach, adding longer unknown words and removal of stop words. Experiments on standard Chinese TREC 5 and TREC 6 data sets show there is inconsistent performance on the 54 queries for different approaches. Some approaches work effectively in some queries but work poorly in others queries. Correct segmentation dose not necessarily lead to improve the precision of Chinese IR. This is mainly caused by the ambiguous nature of language and weakness of retrieval model. Characters, words, phrases (compound or collocation), segmentation, synonyms, homonyms, term weighting and vector space model all have great impact on performance of Chinese IR. Some have impact on precision but others have impact on recall. A bigram approach with unigrams and longer extracted words achieves best results by combining the advantage of shorter and longer words. It strikes a balance between the

precision and recall.

1.6 Thesis Organization

We intend to make each chapter self-contained. The thesis is structured as largely a series of papers, with one paper per chapter and interspersed with theoretical background chapters. We review the literature in situ on a chapter-by-chapter basis. There is some minor repetition in some chapters. The thesis is organized as follows.

Introduction

- In chapter 1 we introduce the issues in English spelling checking and Chinese language processing.

Part 1 Word Disambiguation for English Spelling Checking and Chinese Pinyin Input

- In chapter 2 we apply machine learning approaches for context sensitive spelling checking in English.
- In chapter 3 we present background on statistical n-gram language modeling.
- In chapter 4 we adopt a compression-based approach for Chinese Pinyin input.
- In chapter 5 we propose an error-driven adaptive language model for Chinese Pinyin-to-character conversion.

Part 2 Chinese Word Segmentation and Classification

- In chapter 6 we show the difficulties in Chinese word definition and Chinese word segmentation and provide a literature review on Chinese word segmentation.

- In chapter 7 we use contextual entropy for Chinese word segmentation.
- In chapter 8 we explore the possibility to classify Chinese word unsupervised.

Part 3 Chinese Information Retrieval

- In chapter 9 we describe using suffix array to computer term frequency and document frequency for n-grams.
- In chapter 10 we investigate the impact on the performance of Chinese information retrieval using Word-based approaches and N-gram based approaches.

Conclusions

- In chapter 11 we conclude by summarizing the thesis and providing an outlook for future research.

Part I

WORD DISAMBIGUATION FOR ENGLISH SPELLING CHECKING AND CHINESE PINYIN INPUT

2. MACHINE LEARNING FOR CONTEXT SENSITIVE SPELLING CHECKING

This chapter[1] describes a new approach to automatically learn contextual knowledge for spelling and grammar correction - we aim particularly to deal with cases where the words are all in the dictionary and so it is not obvious that there is an error. Traditional approaches are dictionary based, or use elementary tagging or partial parsing of the sentence to obtain context knowledge. Our approach uses affix information and only the most frequent words to reduce the complexity in terms of training time and running time for context-sensitive spelling correction. We build large scale confused word sets based on keyboard adjacency and apply our new approach to learn the contextual knowledge to detect and correct them. We explore the performance of auto-correction under conditions where significance and probability are set by the user.

2.1 Introduction

In many applications it is necessary to correct errors that have been introduced by human typists and operators, including non-native speakers, or by artificial intelligence systems such as speech recognition or (optical or handwritten) character recognition, or even by machine translation.

Errors that simply involve non-words being generated can very easily be discovered by looking up a dictionary, but such simple spell-checkers are inadequate

[1] This chapter is based the paper (Huang & Powers 2001)

Huang, J. H. & Powers, D. (2001), Large scale experiments on correction of confused words, in 'Australiasian Computer Science Conference', IEEE Computer Society Press, Gold Coast, Queensland.

to the extent that they cannot pick up errors which involve substitution of another valid word, or which involve grammatical errors. We (Powers 1997*a*) distinguish six different types of reasons for substituted word errors: typographic error ("form" versus "from"), homophone error ("peace" and "piece"), grammatical error ("among" and "between"), frequency disparity errors, learners' errors and idiosyncratic error. These are often present in combination - in particular frequent words like "are" are often substituted for less frequent but similar sounding words like "our": it seems that our fingers automatically complete the more common confusions of words that are nearby either on the keyboard or phonetically - and they can even complete common endings like "-ing".

These errors account for anywhere from 25% to over 50% of observed spelling errors (Kukich 1992). Fixing these kinds of errors requires analyzing the contextual information and is not handled by conventional spell-check programs. The task of fixing these spelling errors that happen to result in valid words is called context-sensitive spelling correction. Note however, that all spelling correction is context sensitive - the difference with confused words is that the identification of spelling errors is also context sensitive.

2.2 Confused Words

Rather than attempting to detect and correct all possible errors, our context-sensitive correction algorithm attempts to choose between known pairs or sets of ambiguous words for which statistics are present at significant levels. The ambiguity among words is modelled by confusion sets. A confusion set means that each word in the set could mistakenly be typed when another word in the set was intended.

These confusion sets can be discovered based on a number of models and sources of errors, including keyboard proximity (typos), phonological similarity (phonos) and grammatical confusion (grammos).

For keyboard proximity, we model which keys are adjacent and thus often

substituted, we model omissions of letters, shifting of a pattern left or right on the keyboard, clipping an adjacent key causing an insertion. These models can be used to auto-correct words that aren't in the dictionary, or can be used with the methods explained below to pick up and correct problems where they happen to produce a valid word.

For phonological similarity, we use a dictionary to map to a phonological representation and then look in a similar way for exact homophones as well as near homophones resulting from substitution, deletion or insertion of a phoneme.

Frequency information also needs to be taken into account as a bias, and we can potentially tune our models at run time to the kinds of idiosyncratic errors that are frequently made by an individual - taking note of the corrections that they make themselves as they type or on subsequent proof-reading.

There are also databases/corpora of common errors made by second-language learners, e.g. foreign speakers of English. This information can be treated in the same way as the sets of words discovered using the above models, and indeed there are also models explaining the type of errors made by language learners of a specific linguistic/cultural background.

2.3 Context-sensitive Spelling Correction

The general problem considered in context-sensitive spelling correction is the resolution of lexical ambiguity, both syntactic and semantic, based on the features of the surrounding context. Two kinds of features have been shown useful for this: context words and collocations. Context words test for the presence of a particular context word within $\pm$n words of the ambiguous target word. The context words capture the semantic atmosphere (discourse topic, tense, etc.). Collocations test for a pattern of up to m contiguous words and/or part of speech tags around the target word. Collocations capture local syntax.

Previous work has been done based on how to learn these two types of features for the lexical disambiguation. The word trigram methods (Mays, Damerau &

Merser 1991) capture the collocation information and perform well if the confused words have different part-of-speech tags. Bayesian classifiers (Gale, Church & Yarowsky 1994) can be trained very efficiently to learn features for word disambiguation and perform well if the confused words have same part-of-speech tags. A hybrid of trigram and Bayesian methods (Golding & Schabes 1996) capture the both features and improve the performance irrespective of the part-of-speech tags of the confused words.

The decision list methods (Yarowsky 1994) learn set of features during training time and these features are sorted in order of decreasing strength and stored in a decision list. The strength of a feature reflects its reliability for decision-making. An ambiguous target word is then classified by running down the list and matching each feature against the target context. The first feature that matches is used to classify the target word. The decision list methods only take into account the single strongest piece of features.

The Bayesian hybrid (Golding 1995) is similar as the decision list method. It starts with a list of all features, sorted by decreasing strength. It classifies an ambiguous target word by matching each feature in the list in turn against the target context. Instead of stopping at the first matching feature, however, it traverses the entire list, combining evidence from all matching features to classify the target word..

The Winnow-based approach (Golding & Roth 1996) overcomes the limitation of maintaining large set of features and is implemented in layers with many classifiers to each word in the confusion set and different set of features with weights are assigned to each classifier. Activation value is calculated for each word in a confusion set by multiplying the feature and its corresponding weight and summing up the values of every classifier. Word with highest activation value is treated as the correct word.

Transformation based learning (TBL) (Mangu & Brill 1997) is an automatic rule acquisition method. It represents the learned knowledge in an easily understandable form and uses it to enhance the performance.

All these learning methods have gradually improved the accuracy of the context-sensitive spelling correction. But in obtaining the collocations most need to use a dictionary to tag each word in the sentence with its set of possible part-of-speech tags, which increases the complexity of the system in terms of both training time and the running time, whilst those that use words directly are limited to trigram statistics due to the exponential explosion of possibilities.

Entwisle's (Entwisle 1997) parser which uses crude affix information to parse English inspires us to obtain syntactic information only based on sentence form. We use two kinds of word forms to capture the syntax around the target word: the most frequent words and affixes. Entwisle exploits the fact that the most frequent words tend to have a syntactic function and indeed almost all are function words (Kilgaroff 1996). Noting of vowel or a consonant prefix allows us to make the 'a/an' distinction, whilst suffixes capture the most useful syntactic features. Both the most frequent words and affixes give us the syntactic cues to discriminate the confused words. We define them as eigenunits. Tagging each word around the target word using a dictionary is simply replaced by matching the eigenunit. This significantly reduces the complexity from the order of a million possible tokens per position, to a few hundred.

With the availability of large text corpora, it has become possible to automatically learn the grammatical rules directly from the text, instead of manually generated rules, which can be time consuming. Furthermore it is difficult to generate all syntactic and semantic rules, as the rules of language are vast and idiosyncratic. Learning rules from corpora is more realistic and applicable. Traditional 'spell-checking' and 'grammar-checking' tend to use fixed rules of thumb which lead them to flag all occurrences of particular words like 'which' or particular constructs like passives or prepositions at the end of sentences. These are deprecated by style manuals, but are very commonly used and not really wrong.

2.4 Experiment and Result

The Wall Street Journal (1987-1992 - WSJ) and the Lewis Carroll's novel Alice's Adventures in Wonderland (Alice) were used in this experiment. Around 71.6M words (WSJ87-89, 91-92) were used for training and 1990 WSJ was used for testing. Alice was used as an additional validation corpus representing a totally different genre from WSJ. The first phase of the project involved developing the initial sets of confused words - primarily for the modeled typographical errors. Peterson (Peterson 1986) shows that up to 15% of typographical errors yield another valid word in the language. We used a Perl script to extract 7,407 pairs of confusable words based on 6,131 words from the 25,143-word Unix dictionary by systematically performing character insertion, deletion or transposition. These include the following four situations:

1. where adjacent keys are substituted such as "sun" and "sin";
2. where one character is deleted or inserted such as "its" and "it's";
3. two characters are transposed such as "form" and "from";
4. where two characters are adjacent on the keyboard and are substituted with the wrong pair of adjacent characters such as "trap" and "reap".

About 44% of the words in the training corpus belong to these confused word sets.

The second phase concerns selection of the eigenunits. We use the 145 frequent words and function words plus 65 common suffixes, a dummy null-inflection suffix and the 34 individual non-alphanumeric punctuation characters as our eigenunits. In order to distinguish between "a" and "an", we use vowel and consonant prefix versions of the 66 suffixes doubling the number of suffix towards to 132. For example "invention" and "nation" convert to "V-ion" and "C-ion". We also classify week, month, ordinal number and cardinal number as separate classes. Irregular word forms can also be usefully added to the eigenunits to reduce to the

	1	2	3	4	5	Total	%
Occurs<10	5601175	3928959	114481	414	2171	9647200	87
Occurs≥10	1020095	398380	13798	151	653	1433077	13
S≥70,P≥70	283959	23002	0	0	0	306961	2.8

Tab. 2.1: Diameter with occurrences, significance and probability - number of contexts (WSJ 87-89,91-92)

noise in these but we did not choose to use these as the existing eigenunits cover at least 85% of the training and validation texts (85.6% in the 18.7M 1991 WSJ and 88.4% in the 26.5K Alice corpus).

Once we had the sets of confused words and the eigenunits, the third phase was to develop statistics from a large corpus (5 years of WSJ from 1987-89,91-92). For each ambiguous word we learn the rules simply by substituting the surrounding words with eigenunits and counting the occurrences of the rules. We gradually extend the window size from 1 to 5 on both sides until a desired degree of significance is reached. We do this to avoid learning rules that are useless because the context is so large that insufficient examples are present to learn from. Based on each context for each confusion set, we ignore the context occurring less than a minimum occurrence threshold, currently set to 10, as these occurrences are not sufficient to discriminate confused words reliably. Only where there are more than 10 contexts available do we perform the relatively expensive significance calculation according to Fisher's exact test (Winston 1993). We store the statistics (S and P) of the contexts for each confusion set. P is the probability of cases that we correct, viz the probability we can correct this word in the this context. It corresponds to recall. S is significance (in %) and is complementary to traditional α. It reflects whether we have enough example for P to be significant. It corresponds to precision.

From the Tables 2.1 and 2.2 we can see that certain contexts allow reliable correction and what window size of the contexts represents the syntactic information which is most significant and useful. From Table 2.1 the diameter 2 is optimal to catch the syntax around the targeted word given our limited train-

Occurrence≥10	P≥0	P≥70	P≥80	P≥90	P≥95
S≥0	1433077	1413030	1399495	1379896	1350122
S≥70	326420	306961	294005	272192	247368
S≥80	291713	272843	261050	242090	220657
S≥90	247499	231416	222023	206704	189346
S≥95	218889	204697	196581	183469	168604

Tab. 2.2: Relationship between probability and significance - number of contexts (WSJ 87-89,91-92)

ing corpus. Golding (Golding 1995) obtained a similar result indicating that the window size 2 for collocations generally did best to discriminate among words in the confusion set. Table 2.2 shows that most highly significant contexts are high probability but not vice versa, as expected. High probabilities without high significance are probably not trustworthy. It remains to be seen how best to tradeoff between probability and significance in user trials - some users want to be sure to catch all errors even if that means lots of false corrections are proposed. Others would rather see only errors with a high degree of certainty, viz. high significance and probability. Note that precision and recall are inappropriate as there are no positives seeded.

We record the contexts which are reasonably significant and likely to suggest a correction (S≥70% and P≥70%). Both significance and probability can be used in defining a function for correction. These statistics based on the surrounding words will be sufficient to give us a context in which one choice is clearly preferred.

We tested our text corrector on two issues of the withheld 1990 WSJ test corpus as well as on a validation corpus of an entirely different genre, namely Alice's Adventure on Wonderland. Initially we did not seed any error into these corpora. Table 2.3 tells us that our system will introduce around 0.3% false errors on the same genre (WSJ) but introduce 0.6% false errors on the different genre (Alice). With less significance and probability, more false errors will introduce. This shows that our system is a genre oriented as expected, and that our use

Corpus Size Confused Words	S & P	Errors Introduced	% E/W	Confused Sets(CS)	Significant Sets(SS)	Coverage SS/CS %
WSJ0801	S,P≥95	199	0.30		52791	24.3
64718	S,P≥80	532	0.80	216963	65355	30.1
34805	S,P≥70	666	1.02		70794	32.6
WSJ1231	S,P≥95	169(2)	0.30		44587	23.9
56163	S,P≥80	467	0.83	186812	55543	29.7
29594	S,P≥70	572	1.02		60435	32.4
Alice	S,P≥95	156(11)	0.60		17467	15.0
26457	S,P≥80	772	2.92	116640	22464	19.3
20082	S,P≥70	938	3.55		24472	21.0

Tab. 2.3: False errors and coverage testing on test and validation corpora with no errors seeded but two real errors found

of significance and probability even at these moderate levels keeps the number of false corrections under control - this is the major problem with conventional systems.

In order to evaluate our system, we collect the statistics for all the confused words occurring on the test and validation corpora. For each of these confused words we count the significant confused word sets of all its possible confused word sets according to the levels of significance and probability. We can only detect and correct the errors occurring on these significant confused word sets, so the coverage represented by these significant word sets predicts the expected number of errors we can detect (recall). Table 2.3 shows us that our system has about 24% coverage on the same genre but only 15% on the different genre (Alice) - at the levels of 95% significance and probability. One reason why we obtain such a low coverage is that we have a very comprehensive set of confusable words and the training corpus is not large enough to learn significant contexts for all of them. Also, our confusion sets include semantic errors such as “he” and “she” which are difficult to distinguish using local context alone.

Corpus	Errors Seeded	S & P	Errors Detected	Detect Rate(%)	Errors Corrected	Correct Rate(%)
WSJ0801 (90)	3253	S,P≥95	806	24.8	664	20.4
		S,P≥80	980	30.1	804	24.7
		S,P≥70	1058	32.5	853	26.2
WSJ1231 (90)	2792	S,P≥95	663	23.7	541	19.4
		S,P≥80	856	30.7	686	24.6
		S,P≥70	919	32.9	734	26.3
Alice	1588	S,P≥95	279	17.6	226	14.2
		S,P≥80	364	22.9	293	18.5
		S,P≥70	390	24.6	308	19.4

Tab. 2.4: True errors detected (recall) and corrected when errors seeded randomly

Corpus	Errors Seeded	Errors Detected	Detect Rate (%)	Errors Corrected	Correct Rate (%)
WSJ0801(90)	331	267	80.7	227	68.6
WSJ1231(90)	288	245	85.1	214	74.3
Alice	36	18	50.0	16	44.4

Tab. 2.5: Seeded errors of the confusion set of "*from*" and "*form*" (S,P≥95%)

To evaluate our system further, we seed one error in every 20 words randomly across the testing corpus according to the confusion set. Table 2.4 shows that we do get the expected levels of recall, but not all of these errors are successfully corrected so the correction rate is slightly lower. We also see that detection and correction rates drop for the contrastive validation corpus as expected. Note however that Table 2.5 shows that we get much better than average detection and correction rates for syntactic errors like "form" versus "from", but even this is affected by the genre.

Another reason for low recall is that irregular forms that do not take the standard suffixes and are not included amongst our 145 most frequent words distort the contexts around the target word - e.g. less common irregular past tense forms are misinterpreted but this could be remedied by adding these forms in to the eigenset. This distortion also causes many of the false errors introduced by the system. Note that we can decrease the level of significance and probability to increase the recall but it will then introduce more false errors and miscorrections (as reported in Table 2.3 to 2.5).

As seen in Table 2.4 we actually obtain about 24% detection rate at 95% significance/probability level overall. This coincides with the testing results of Table 2.3. But we only obtain about 20% correction rate at the same significance/probability level. From Table 2.4 we know that the system can detect 806 errors at levels of 95% both significance and probability when seeded with 3253 errors on the testing corpus (WSJ0801). Of these 806 errors the system can automatically correct 664 errors (82%). The other 142 errors have two or more proposals to correct them - all of these are marked incorrect here, although around 50% would be expected to be handled correctly by simply choosing the most probably confusion set. Further experiments need to be done to find out how many errors of these 142 errors detected can be automatically corrected based on the value of significance and probability of each proposal. Note that two real errors were discovered in the WSJ test corpus shown in Table 2.7.

Corpus	Errors Seeded	S & P	FPR %	TPR Detected	TPR Corrected	Informedness	
						Detected	Corrected
WSJ0801	3253	S,P≥95	0.30	24.80	20.40	24.50	20.10
		S,P≥80	0.80	30.10	24.70	29.30	23.90
		S,P≥70	1.02	32.50	26.20	31.48	25.18
WSJ1231	2792	S,P≥95	0.30	23.70	19.40	23.40	19.10
		S,P≥80	0.83	30.70	24.60	29.87	23.77
		S,P≥70	1.02	32.90	26.30	31.88	25.28
Alice	1588	S,P≥95	0.60	17.60	14.20	17.00	13.60
		S,P≥80	2.92	22.90	18.50	19.98	15.58
		S,P≥70	3.55	24.60	19.40	21.05	15.85

Tab. 2.6: False positive rate (FPR), true positive rate (TPR) and informedness when errors seeded

Table 2.6 shows false positive rate (FPR), true positive rate (TPR) and informedness. Informedness specifies the probability that a prediction is informed in relation to the condition (versus chance) (Powers 2008, Powers 2011). It measures how consistently our model predicts the errors by combining FPR and TPR about what proportion of errors are correctly predicted. $Informedness = tpr - fpr$. Informedness is more reliable and consistent than TPR to evaluate the model.

We now turn to look at accuracy in terms of the false errors from the original corpus (Table 2.3). No matter how many true errors are seeded in the corpus, we cannot change these false errors. The more seeded errors, and the higher accuracy we require, the more false errors introduced. Note that an error in one word may be identified as an error in an adjacent word - every context it is a member of will be affected if it is a listed eigenword or the affix is corrupted. We also see that the number of false errors rises significantly with the change of corpus and that use of high significance high probability contexts is even more critical.

Note that in these result tables we only illustrate with results for comparable significance and probability levels, except for Table 2.2 where we show the relationship between these. In general, there is little point in accepting a high

The company emphasized that immediate cutbacks in its work force won't be necessary **beacause** the assembly line already is working at reduced speed and can rely on a multibillion-dollar cushion of previously authorized planes.
In fact, as the insurance business becomes more complex, regulators are relying more on computers than increased manpower to monitor **insurers's** health.
Although each vendor's product is different, all do allow searching on most of the **data bases**.
He said the jump in rates reflects **unsually** strong demand for funds at a time when many lenders in the market
UnionFed said its sharply higher addition to reserves resulted from deterioration in **commerical** real estate markets and sounded a back-to-basics theme.
The CFTC is seeking a temporary restraining order from a federal judge in Chicago, where Stotler is based, and is asking for the **appointement** of a receiver "to protect the customer funds in the pool," said CFTC enforcement director Dennis Klejna.
Civil libertarian **conservativism** is a natural in an age of political prosecutions, independent counsel and RICO.
Nevertheless, the White House so far has displayed no sign that Mr. Bush has accepted advice from some congressional Republicans that he **criticize** Democrats for not putting forward a deficit-reduction proposal of their own.
Ever since Tuffier, Ravier, Py & Associes, one of France's largest independent **stockbrokage** firms, went bankrupt last month, rumors have run rife of who will be the next to go under.

Tab. 2.7: Spelling Errors Found in WSJ0801 and WSJ1231

probability that is not supported by high significance. Conversely, when we insist on high significance, and have therefore reduced the coverage considerably, we probably also want high certainty corrections. Significance tells us how confident we are in the probabilities, while the probabilities tell us the likelihood of the proposed correction being correct. In a more complex approach (e.g. involving lattice techniques to deal with probabilities on multiple words in a context) probability will be used independently of significance.

A final issue relating to accuracy is the lack of a psychologically or empirically motivated user-model. At this stage we are using an elementary model that assumes that all errors relate directly to low keyboard or phonological distance, but in fact as discussed above, word frequency, language and idiolectic background play a role, and certain types of errors compassed in our confused words sets are much rarer than our model predicts. We propose to tune this model by obtaining corpora of language learner errors, typographic corrections, and by making use of the statistics for errors which do lead to non-dictionary words to inform our model.

2.5 Interface

In order to compare our text corrector to Microsoft Spelling and Grammar-checking, we integrated our text corrector into Microsoft Word 97 using Macro, Visual Basic and Access. This is useful for the user in evaluating the performance of the system as well. Microsoft Word 97 spelling checker can only correct 90 pairs of confused words but our corrector can detect and correct 7,407 pairs of confused words. Our text corrector outperforms Microsoft Word 97 spelling checker by picking up more errors based on informedness (Microsoft Word 97 is less 2% while ours is around 25$\tilde{3}$0%) but both introduce some new errors. Initially we proposed to use the significance and probability to color the words so that the words that are more likely to be wrong are highlighted more strongly but experience with the color coding in the latest versions of Word indicate that this may confuse or annoy the user and detract from appropriate attention to the

significant corrections in the text. At this stage we only display the significance and probability of the alternative to the user in a dialog box when a highlighted word satisfying the significance and probability thresholds is selected.

Note that, as discussed above, there are two types of errors that a spelling corrector always can make: false negatives (complaining about a correct word) and false positives (failing to notice an error), so in order to give the user the opportunity to trade off these two kinds of errors, we allow the user to change the significance and probability at which notification of potential errors occurs. Thus users can decide the balance between being bothered for some false errors and missing some true errors. Conservatively we set this at a 95% significance level and a presion setting of 95% in the confusion set but in fact higher informedness is achieved at 70% with equal weight to both kinds of errors but we prefer to bias to avoid false positives. The interface allows setting levels of significance and probability for auto-correction to occur (as well as auto-detection).

2.6 Conclusion and Future Work

The technique we developed here can be used to resolve lexical ambiguity in the syntactic sense. It captures the local syntactic patterns but not semantic information as the eigenunits can not represent the semantic association with the target word. For example the word "cake" maybe is useful to disambiguate the confusion set dessert and desert but "cake" does not exist in the eigenunits so this association cannot be learned. Furthermore the window size 2 is too small to capture this association. For semantic information a window size of 20 seems to be required (Powers 1997*a*, Golding 1995, Golding & Roth 1996, Mangu & Brill 1997, Kilgaroff 1996), but this is far larger than we can deal with using the present approach - normally such windows are handled by simply looking for cooccurrences within a certain distance rather than specific sequences of the window size. Further work need to be done to exploit this distant word association to generate more efficient algorithm for resolving this problem and minimizing the features we learned.

In order to improve the performance of the system, we must also handle the noise caused by the irregular words in the eigenunits. As mentioned above this noise does not make the statistic collection much worse but it will distort the context around the target word when the corrections are being made. This is one of the main causes of the false errors. Given the vast confusion sets we have, we need to optimise the confusion sets to build a better model through evaluating each confusion set as discussed in relation to the test and validation results.

We expect to be able to reduce the number of false corrections by modeling the kind of errors people actually make in more detail, as at present we primarily use keyboard adjacency.

2.7 Reflections

Many new machine learning methods (Reynaert 2004, Al-Mubaid & Nagula 2005, Van den Bosch 2006*a*, Van den Bosch 2006*b*, Rodriguez & Diaz 2007, Stehouwer & den Bosch 2007, Stehouwer & van Zaanen 2009*a*, Stehouwer & van Zaanen 2009*b*, Stehouwer & van Zaanen 2010) have been proposed for context sensitive spelling correction since this paper (Huang & Powers 2001) was published. Al-Mubaid and Nagula (Al-Mubaid & Nagula 2005) used two machine learners based on logic learning and support vector machine (SVM) to train the classifiers. Wilcox-O'hearn, Hirst and Budanitsky (Wilcox-O'hearn et al. 2008) combined the tri-gram models and WordNet for real-word spelling correction. Islam and Inkpen (Islam & Inkpen 2009) proposed a normalized and modified version of the longest common subsequence (LCS) string matching algorithm using the Google Web 1T n-gram data set for correcting real-word spelling errors. Bergsma, Lin and Goebel(Bergsma et al. 2009) applied supervised and unsupervised approaches to learn the weight of the Google web counts of different context sizes and positions for lexical disambiguation. Stehouwer and van Zaanen (Stehouwer & van Zaanen 2010) used unlimited size n-gram language models based on suffix array for word correction. Several new applications have emerged, including correction of search queries (Cucerzan & Brill 2004, Li, Zhu, Zhang & Zhou 2006, Gao, Li, Micol,

Quirk & Sun 2010).

The use of significance as integral part of the algorithm remains a unique contribution of our work.

3. STATISTICAL N-GRAM LANGUAGE MODELING

In this chapter, we present some background on statistical language modeling, particularly on ngram language models.

3.1 Statistical Language Modeling

Statistical language modeling was first investigated in the context of speech recognition. Speech recognition consists of generating accurate written transcriptions for spoken utterances. The problem can be formulated as a maximum-likelihood decoding problem, or the so-called noisy channel problem. Given a speech utterance, speech recognition consists of determining its most likely written transcription.

If we let O denote the observation sequence produced by a signal processing system, W a word transcription sequence over an alphabet $\mathcal{A}$, and $P(W|O)$ the probability of the transduction of O into W, the problem consists of finding $\hat{W}$ as defined by:

$$\hat{W} = \arg \max_{W \in \mathcal{A}^*} P(W|O) \tag{3.1}$$

Using Bayes' law,

$$P(W|O) = \frac{P(O|W)P(W)}{P(O)} \tag{3.2}$$

Since $P(O)$ does not depend on W, the problem can be reformulated as:

$$\hat{W} = \arg \max_{W \in \mathcal{A}^*} P(O|W)P(W) \tag{3.3}$$

where $P(W)$ is the a priori probability of the written sequence W in the language considered and $P(O|W)$ the probability of observing O given that the sequence W has been uttered. The probabilistic model used to estimate $P(W)$ is called the language model. The generative model associated to $P(O|W)$ is called the acoustic model. Without a language model, a speech recognition system will typically demonstrate poor performance. For example, the following two word sequences may have similar acoustic signals.

- word sequence 1: How to recognize speech
- word sequence 2: How to wreck a nice beach

An acoustic model alone may struggle to discriminate between these two word sequences. However, with the introduction of language model $P(W)$, these two sentences might be easily discriminated, since "how to recognize speech" is a much more likely word sequence than "how to wreck a nice beach" particularly in the context of language technology.

In general, statistical language modeling is concerned with determining the probability of naturally occurring word sequences in a language. Although the traditional motivation for language modeling has come from speech recognition, statistical language models have recently become more widely used in many other application areas, such as information retrieval, machine translation, optical character recognition, spelling correction, document classification, information extraction, and bio-informatics.

The goal of statical language modeling is to predict the probability of natural word sequences:

$$W = w_1 w_2 \cdots w_N = w_1^N$$

It puts high probability on word sequences that actually occur and low probability on word sequences that never occur. The quality of a language model should be measured by its impact on the actual application at hand. For example, in speech recognition, the quality of a language model should be measured by how much it improves recognition accuracy. However, since speech recognition also involves a

complex acoustic model, evaluating the quality of a language model in terms of recognition accuracy is complicated. In practice, people measure the quality of a language model by its empirical entropy (or perplexity) on test data. Given a word sequence $w_1 w_2 \cdots w_N$ to be used as a test corpus, the quality of a language model can be measured by the empirical perplexity and entropy scores on this corpus (Bahl, Jelinek & Mercer 1983).

$$Perplexity = \sqrt[n]{\prod_{i=}^{n} \frac{1}{\Pr(w_i|w_1 \cdots w_{i-1})}} \tag{3.4}$$

$$Entropy = \log_2 Perplexity \tag{3.5}$$

The goal is to obtain small values of these measures. Although these measures have received criticism. It does not address the issue of lexical 'bursting'. A word may have a much higher frequency in one segment of text than in another. Therefore its probability over an entire text may be much lower than its actual occurrence in some parts of the text and much higher than its actual occurrence in other parts of the text. Some new measures have been proposed (Clarkson 1999), but they are still the standard measures used in the language modeling literature.

The simplest and most successful basis for language modeling is the n-gram model wherein a word is assumed to depends only on the previous $n-1$ words. Many language models have been proposed in the literature to improve a basic n-gram models. These more sophisticated techniques include link grammars (Lafferty, Sleator & Temperley 1992), sentence mixtures (Iyer & Ostendorf 1999) , decision trees (Bahl, Brown, de Souza & Mercer 1989), clustering (Brown, Pietra, deSouza, Lai & Mercer 1992), caching(Kuhn & De Mori 1990), skipping models (Rosenfeld 1994, Siu & Ostendorf 2000), latent semantic analysis (Bellegarda 2000), structured language models (Chelba & Jelinek 1998, Charniak 2001), neural network models (Bengio & Vincent 2001), maximum entropy models (Khudanpur & Wu 2000), probabilistic logic (Bouchaffra 2005), Random forests (Xu & Jelinek 2007) and conditional random field (Lafferty, McCallum & Pereira 2001, Roark, Saraclar, Collins & Johnson 2004). The two references

(Goodman 2001, Rosenfeld 2000) provide a thorough overview and systematic investigation of current techniques. However the improvements obtained by these sophisticated language models sometime do not justify their added complexity. In many situations the n-gram language model is still the best choice in practice and has been successfully applied to many real world problems such as information retrieval, language modeling, part-of-speech tagging, word sense disambiguation and speech recognition. Below we will introduce n-gram language modeling in more details.

3.2 N-Gram Markov Language Models

Note that by the chain rule of probability we can decompose the probability of any word sequence as

$$p(w_1 \dots w_i) = p(w_1^i) = p(w_1) * p(w_2|w_1) * \dots * p(w_i|w_1^{i-1}) \tag{3.6}$$

An n-gram model approximates this probability by assuming that the only words relevant to predicting

$$p(w_i|w_1^{i-1})$$

are the previous $n-1$ words; that is, it assumes

$$p(w_i|w_1^{i-1}) \approx p(w_i|w_{i-n+1}^{i-1}) \tag{3.7}$$

A straightforward maximum likelihood estimate of n-gram probabilities from a corpus is given by the observed frequency

$$p_{ML}(w_i|w_{i-n+1}^{i-1}) = \frac{C(w_{i-n+1}^i)}{\sum_{w_i} C(w_{i-n+1}^i)} \tag{3.8}$$

where $C(w_{i-n+1} \dots w_i)$ is the number of occurrences of a specified n-gram in the training corpus.

For a typical trigram model, the probability of a word sequence $w_1 \dots w_i$ can be formulated as

$$p(w_1^i) = \prod_{n=1}^{i} p(w_n|w_{n-2}w_{n-1}) \tag{3.9}$$

Although one could attempt to use these simple n-gram models to capture long range dependencies in language, attempting to do so directly immediately creates sparse data problems. The maximum likelihood method of training depends on the probabilities of word sequence occurrences as measured from training data. It is not practically possible to ensure that every word in a language model's lexicon appears in a training data set. One is likely to encounter novel n-grams that were never witnessed during training in any test corpus. The probability of these novel n-grams which, by chance, did not appear in the training data would be zero. This phenomenon is referred to as data scarcity. Therefore some mechanism for assigning non-zero probability to novel n-grams is a central and unavoidable issue in statistical language modeling.

3.3 Smoothing Methods

The standard approach to smoothing probability estimates to cope with the sparse data problem (and to cope with potentially missing n-grams) is to use some sort of interpolated or back-off estimator. Different methods can be used for computing the discounted probability. Typical discounting techniques include add one smoothing, absolute smoothing, Good-Turing smoothing, Witten-Bell smoothing, Kneser-Ney smoothing and modified Kneser-Ney smoothing. Chen and Goodman (Chen & Goodman 1999) have made a complete comparison of these smoothing techniques.

3.3.1 Add One Smoothing

In add one discounting (Lidstone 1920), the frequency of a word is added by one. The probability of a word w_i given $w_{i-n+1} \cdots w_{i-1}$ is calculated as:

$$P(w_i|w_{i-n+1}^{i-1}) = \frac{C(w_{i-n+1}^{i}) + 1}{|V| + C(w_{i-n+1}^{i-1})} \tag{3.10}$$

where V is the vocabulary, $|V|$ is the size of vocabulary.

3.3.2 Interpolation Smoothing

A linear interpolated model (Jelinek & Mercer 1980) is a linear combination of several different order n-gram models.

$$p(w_i|w_{i-n+1}^{i-1}) = \lambda p_{ML}(w_i|w_{i-n+1}^{i-1}) + (1-\lambda)p(w_i|w_{i-n+2}^{i-1}) \tag{3.11}$$

where $ML(w_i|w_{i-n+1}^{i-1})$ is the maximum likelihood estimate and λ is its weight which is normally computed by hold-out estimation.

3.3.3 Absolute Discounting

In absolute discounting (Ney, Essen & Kneser 1994), the frequency of a word is reduced by a constant D. The probability of w_i given $w_{i-n+1} \cdots w_{i-1}$ is then calculated as:

$$p(w_i|w_{i-n+1}^{i-1}) = \frac{c(w_{i-n+1}^{i}) - D}{c(w_{i-n+1}^{i-1})} \tag{3.12}$$

where D is often defined as:

$$D = \frac{n_1}{n_1 + 2n_2} \tag{3.13}$$

Here n_r denotes the number of words that occur r times.

3.3.4 Good-Turing Discounting

In Good-Turing discounting (Good 1953), the probability of a word w_i given w_{i-n+1}^{i-1} is calculated as:

$$p(w_i|w_{i-n+1}^{i-1}) = \frac{C_{GT}(w_{i-n+1}^{i})}{C(w_{i-n+1}^{i-1})} \tag{3.14}$$

where the frequency r is discounted as:

$$r_{GT} = (r+1)\frac{n_{r+1}}{n_r} \tag{3.15}$$

3.3.5 Katz Back-off Smoothing

A back-off n-gram model (Katz 1987) is defined as

$$p(w_i|w_{i-n+1}^{w_{i-1}}) = \begin{cases} p_{discount}(w_i|w_{i-n+1}^{w_{i-1}}) & if c(w_{i-n+1}^{w_i}) > 0 \\ \alpha(w_{i-n+1}^{w_{i-1}}) * p(w_i|w_{i-n+2}^{w_{i-1}}) & otherwise \end{cases} \tag{3.16}$$

where

$$p_{discount}(w_i|w_{i-n+1}^{i-1}) = \frac{c_{discount}(w_{i-n+1}^{i})}{c(w_{i-n+1}^{i-1})} \tag{3.17}$$

and $\alpha(w_{i-n+1}^{i-1})$ is normalization constant, calculated to be

$$\alpha(w_{i-n+1}^{i-1}) = \frac{1 - \sum_{x:c(w_{i-n+1}^{i-1}x)>0} p_{discount}(x|w_{i-n+1}^{i-1})}{1 - \sum_{x:c(w_{i-n+1}^{i-1}x)>0} p_{discount}(x|w_{i-n+2}^{i-1})} \tag{3.18}$$

The Katz back-off model extends the Good-Turing discount method by adding the combination of higher order models with lower order models.

3.3.6 Witten-Bell Smoothing

Witten-Bell discounting (Bell, Cleary & Witten 1990, Witten & Bell 1991) is dependent on the number of types of occurrences which followed the particular context rather on the count. The probability of a word w_i given $w_{i-n+1} \cdots w_{i-1}$ is calculated as:

$$p(w_i|w_1^{i-1}) = \alpha \frac{c(w_{i-n+1}^{i})}{c(w_{i-n+1}^{i-1})} \tag{3.19}$$

where α is defined as:

$$\alpha = 1 - \frac{C}{c(w_{i-n+1}^{i-1}) + C} \tag{3.20}$$

C denotes the number of distinct words that can follow w_{i-n+1}^{i-1} in the training data.

3.3.7 Kneser-Ney Smoothing

Kneser and Ney discounting (Kneser & Ney 1995) is an extension of absolute discounting. It combines higher order models with lower order models by back-off model.

$$p_{KN}(w_i|w_{i-n+1}^{i-1}) = \begin{cases} \frac{c(w_{i-n+1}^{i})-D}{c(w_{i-n+1}^{i-1})} & if\ c(w_{i-n+1}^{i}) > 0 \\ \alpha(w_{i-n+1}^{i-1}) p_{KN}(w_i|w_{i-n+2}^{i-1}) & if\ c(w_{i-n+1}^{i}) = 0 \end{cases} \tag{3.21}$$

where

$$p_{KN}(w_i|w_{i-n+2}^{i-1}) = \frac{|w_{i-n+1} : c(w_i^{i-n+1}) > 0|}{|(w_{i-n+1}, wi) : (w_{i-n+1}) > 0|} \tag{3.22}$$

D is optimized on held out data, $\alpha(w_{i-n+1}^{i-1})$ is a normalization constant such that the probabilities sum to 1.

3.3.8 Modified Kneser-Ney Smoothing

Modified Kneser-Ney discounting (Chen & Goodman 1999) is an interpolated variation of Kneser-Ney smoothing with an augmented version of absolute discounting. Instead of using a single discount D for all n-grams, three separate discounts D_1, D_2 and D_{3+} are used for n-grams with one count, two counts, and three or more counts, respectively.

$$\begin{aligned} p_{KN}(w_i|w_{i-n+1}^{i-1}) = & \frac{c(w_{i-n+1}^{i-1}) - D(c(w_{i-n+1}^{i-1}))}{\sum_{w_i} c(w_{i-n+1}^{i-1})} \\ & + \gamma(w_{i-n+1}^{i-1}) p_{KN}(w_i|w_{i-n+2}^{i-1}) \end{aligned} \tag{3.23}$$

where

$$D(c) = \begin{cases} 0 & if\ c = 0 \\ D_1 & if\ c = 1 \\ D_2 & if\ c = 2 \\ D_3+ & if\ c \geq 3 \end{cases}$$

$$\gamma(w_{i-n+1}^{i-1}) = \frac{D_1 N_1(w_{i-n+1}^{i-1}\cdot) + D_2 N_2(w_{i-n+1}^{i-1}\cdot) + D_{3+} N_{3+}(w_{i-n+1}^{i-1}\cdot)}{\sum_{w_i} c(w_{i-n+1}^{i-1})} \tag{3.24}$$

$$N_1(w_{i-n+1}^{i-1}\cdot) = \mid \{w_i : c(w_{i-n+1}^{i-1} w_i) = 1\} \mid \tag{3.25}$$

$$N_2(w_{i-n+1}^{i-1}\cdot) = \mid \{w_i : c(w_{i-n+1}^{i-1} w_i) = 2\} \mid \tag{3.26}$$

$$N_{3+}(w_{i-n+1}^{i-1}\cdot) = \mid \{w_i : c(w_{i-n+1}^{i-1} w_i) \geq 3\} \mid \tag{3.27}$$

$$\begin{aligned} Y &= \frac{n_1}{n_1 + 2n_2} \\ D_1 &= 1 - 2Y\frac{n_2}{n_1} \\ D_2 &= 1 - 3Y\frac{n_3}{n_2} \\ D_{3+} &= 1 - 4Y\frac{n_4}{n_3} \end{aligned} \tag{3.28}$$

3.4 Discussion

The language modeling techniques described above use individual words as the basic unit, although one could instead consider models that use individual characters as the basic unit. The remaining details remain the same in this case. The only difference is that the character vocabulary is always much smaller than the word vocabulary, which means that one can normally use a much higher order n in a character-level n-gram model. The benefit of the character-level model is that it avoids the need for explicit word segmentation in the case of Asian languages.

3.5 Conclusion

In this chapter, we have presented some background on Markov n-gram models, which will serve as a base for next chapters.

4. COMPRESSION-BASED ADAPTIVE APPROACH FOR CHINESE PINYIN INPUT

This chapter[1] presents a compression-based adaptive algorithm for Chinese Pinyin input. There are many different input methods for Chinese character text and the phonetic Pinyin input method is the one most commonly used. Compression by Partial Match (PPM) is an adaptive statistical modelling technique that is widely used in the field of text compression. Compression-based approaches are able to build models very efficiently and incrementally. Experiments show that adaptive compression-based approach for Pinyin input outperforms the most efficient smoothing method - modified Kneser-Ney smoothing method (Chen & Goodman 1999) implemented by SRILM language tools (Stolcke 2002).

4.1 Introduction

Chinese language is a logographic language. There are vastly more characters, or glyphs, than keys on a standard computer keyboard. These characters can't be entered by keyboard directly. To allow the input of Chinese using standard keyboards a variety of keyboard input methods have been designed. Keyboard input methods can be classified in 3 main types: by encoding, by pronunciation such as Pinyin and Zhuyin, and by structure of the characters such as Wubi (Wang 2005) and Changjie. Wubi is more efficient for professional typists but more difficult for ordinary users as users have to remember all the radical parts

[1] This chapter is based on the paper (Huang & Powers 2004)

Huang, J. & Powers, D. (2004), Adaptive compression-based approach for Chinese Pinyin input, in 'Third ACL SIGHAN Workshop on Chinese Processing'.

of each character. Pinyin input method is easy to learn and most widely used. It is not only used in computer devices but also mobile devices (Tseng 2008).

Early products using Pinyin input methods are very slow because of the large number of homonyms in the Chinese language. There are more than 6,700 commonly used Chinese characters and only about 400 phonologically allowed syllables without tones in Mandarin speech. Such homonym problems make the manual selection among all possible homonym characters on the screen necessary after each Pinyin has been entered. This makes Chinese Pinyin input very slow and awkward. It is therefore highly desired to develop some automatic algorithms which can directly decode the sequence of Pinyin syllables into corresponding characters without manual selection. With the progress of statistical language modelling (Goodman 2001) the current phonetic input system such as Microsoft IME for Chinese, Chinese Star and Google Pinyin has achieved great success.

Many research work has been done to improve the automatic conversion of corresponding Chinese characters from input syllables. There are mainly two approaches for automatic conversion task: the linguistic approach(Hsieh, Lo & Lin 1989, Wang 1993, Kuo 1995) and the statistical approach (Gu, Tseng & Lee 1991, Gao, Goodman, Li & Lee 2002, Gao, Suzuki & Yuan 2006, Tseng & Chen 2006). The linguistic approach forms many possible hypotheses from input Pinyin and applies grammars rules to filter out all ungrammatical combinations and obtain the output sentence. Such approaches are very expensive because of too many possible sentence combination and too complicated Chinese grammar rules. The coverage of dictionary and rules is very critical to these approaches. The statistical approach is mainly based on statistical language modeling, especially the Markov n-gram models. It has been shown to be very robust and efficient for Chinese phonetic input without considering any complicated grammar rules. Gu et al. (Gu et al. 1991) only applied Markov models with first order to successfully decode the phonetic sequence into Chinese characters. Gao et al. (Gao et al. 2002) combined pruning and clustering techniques to improve standard statistical language modeling and achieved very good conversion results. But there are some

drawbacks in the n-gram models as it can only capture the local dependency. Many researchers have successfully applied different techniques to address these issues. Trigger techniques (Zhou & Lua 1999), word-pair techniques (Tsai 2006*b*), maximum entropy (Xiao, Liu & Wang 2007), rough set technique (Xiaolong, Qingcai & Yeung 2004) and conditional random fields (Li, Xuan, Wang & Yu 2009) have been proposed to integrate long distance constraints.

4.2 Statistical Language Modelling

The task of statistical language modelling is to determine the probability of a sequence of words.

$$P(w_1 \dots w_n) = P(w_1) * P(w_2|w_1) * \dots * P(w_n|w_1 \dots w_{n-1}) \tag{4.1}$$

An n-gram Markov model approximates this probability by assuming that only words relevant to predict are previous n-1 words. The most commonly used is trigram.

$$P(w_1 \dots w_n) = \prod_{i=1}^{n} P(w_i|w_{i-2}w_{i-1}) \tag{4.2}$$

One major problem with statistical n-gram models is that they must be trained from data. Any particular training data is finite. One never tends to have enough training data to cover all the n-grams. It is bound to have a very larger number of n-grams with zero probability. Therefore some mechanism for assigning non-zero probability to novel n-grams is a critical issue in statistical language modeling. The task of reevaluating some of the zero-probability and low-probability n-grams and assigning them to non-zero values is called smoothing. Smoothing is used to adjust the probabilities and make distributions more uniform. The following smoothing algorithms have been discussed in 3.3.

1. Add one smoothing
2. Absolute discounting
3. Good-Turing discounting

4. Witten-Bell smoothing

5. Kneser-Ney smoothing

6. Modified Kneser-Ney smoothing

Chen and Goodman (Chen & Goodman 1999) made a complete comparison of most smoothing techniques and found that the modified Kneser-Ney smoothing outperformed others.

4.2.1 Pinyin-to-Character Conversion

The process of Pinyin-to-Character conversion task is similar to speech recognition. If we let A be the phonetic Pinyin sequence, W be a word transcription sequence over an alphabet $\mathcal{A}$, and $P(W|A)$ be the probability of the transduction of A into W, the problem consists of finding $\hat{W}$ as defined by:

$$\hat{W} = \arg\max_{W \in \mathcal{A}^*} P(W|A) \tag{4.3}$$

Using Bayes' law,

$$P(W|A) = \frac{P(A|W)P(W)}{P(A)} \tag{4.4}$$

Since $P(A)$ does not depend on W, the problem can be reformulated as:

$$\hat{W} = \arg\max_{W \in \mathcal{A}^*} P(A|W)P(W) \tag{4.5}$$

We assume each Chinese character has only one pronunciation without tones in our experiments . Thus we can use the Viterbi algorithm to find the word sequences to maximize the language model according to Pinyin input.

4.2.2 SLM Evaluation

Perplexity

Given a word sequence $w_1 w_2 \cdots w_n$ to be used as a test corpus, the quality of a language model can be measured by the empirical perplexity and entropy scores

on this corpus.

$$Perplexity = \sqrt[n]{\prod_{i=}^{n} \frac{1}{\Pr(w_i|w_1 \cdots w_{i-1})}} \tag{4.6}$$

$$Entropy = \log_2 Perplexity \tag{4.7}$$

The goal is to obtain small values of these measures. Although these measures have received criticism, they are still the standard measures used in the language modeling literature.

Word Error Rate

Word error rate is also used as a measure for evaluating the language model. It is defined as percentage of the number of misrecognized words out of all tested utterances. In Pinyin-to-Character conversion task, we use character error rate (CER) to evaluate the Pinyin-to-character conversion as one Pinyin is corresponding to one Chinese character.

4.3 Compression Theory

One of the fundamental ideas in information theory is that of entropy, which is measure of the amount of order in a message. It is a measure of quantity of information. Suppose that there is a set of possible events with known probabilities $p_1, p_2, \cdots, p_n$ that sum to 1. Shannon(Shannon 1948) demonstrated:

$$E(p_1, p_2, \cdots, p_n) = -k \sum_{i=1}^{n} p_i \log p_i \tag{4.8}$$

where the positive constant k governs the units in which entropy is measured. Normally, the units are bits, where k=1 and logs are taken with base 2:

$$E(p_1, p_2, \cdots, p_n) = -\sum_{i=1}^{n} p_i \log_2 p_i \ bits \tag{4.9}$$

We are often more interested in finding the information content of a particular choice than in knowing the average over all possible choices. If the probability of

Model	symbol	initial pr.	final pr.	coding(bits)
A	$a \ldots z$	$\frac{1}{26}$	$\frac{1}{26}$	$-\log_2(\frac{1}{26}\frac{1}{26}\frac{1}{26}\frac{1}{26}\frac{1}{26}\frac{1}{26}\frac{1}{26}\frac{1}{26}\frac{1}{26}\frac{1}{26}\frac{1}{26}\frac{1}{26}\frac{1}{26}\frac{1}{26}) = 56.4$
B	a	$\frac{1}{26}$	$\frac{2}{38}$	$-\log_2(\frac{1}{26}\frac{1}{27}\frac{1}{28}\frac{1}{29}\frac{1}{30}\frac{1}{31}\frac{1}{32}\frac{2}{33}\frac{2}{34}\frac{2}{35}\frac{2}{36}\frac{2}{37}) = 54.6$
	b	$\frac{1}{26}$	$\frac{1}{38}$	
	c	$\frac{1}{26}$	$\frac{1}{38}$	
	d	$\frac{1}{26}$	$\frac{2}{38}$	
	e	$\frac{1}{26}$	$\frac{2}{38}$	
	$f \ldots k$	$\frac{1}{26}$	$\frac{1}{38}$	
	l	$\frac{1}{26}$	$\frac{2}{38}$	
	m	$\frac{1}{26}$	$\frac{1}{38}$	
	n	$\frac{1}{26}$	$\frac{1}{38}$	
	o	$\frac{1}{26}$	$\frac{2}{38}$	
	$p \ldots z$	$\frac{1}{26}$	$\frac{1}{38}$	

Tab. 4.1: Compression models for the string "dealornodeal"

a choice is p_i, its entropy is defined to be the negative log of its probability with respect to the model:

$$E_i = -\log_2 p_i \; bits \tag{4.10}$$

It determines the optimum code length required to encode each symbol with respect to the model.

4.4 Adaptive Modelling

To illustrate how adaptive models work, Table 4.1 lists two possible models for the string "dealornodeal". Model A is a static model, always using the same probability estimates and model B is adaptive, adjusting its probability estimates as the coding proceeds. The coding of the string is represented on the right by computing its entropy.

Model A predicts each of the twenty-six symbols in the alphabet, a to z, with equal probability $\frac{1}{26}$. Because this is a static model, the initial probability estimate at the start of the coding is equal to the final probability estimate at

the end. Hence, each symbol requires 4.7 (or $-\log_2(\frac{1}{26})$) bits to encode it, with the resultant code length being 56.4 bits.

Model B adapts to the text coded so far by maintaining frequency counts for each symbol. The probability estimates are obtained by dividing each symbol's frequency count by the total count for all symbols. The frequency counts are initially set to 1, so that at the start the probability estimates are the same as they were for model A. As each symbol is encoded, its frequency count is incremented. The final count for the symbol l is 2, for example, because it appears twice in the string. Its final probability estimate is $\frac{2}{38}$ as the total count is 38 (there are 12 symbols in the string, plus 26 for the initial 1 counts at the start). The resultant code length for the model is 54.6 bits, slightly better than for model A.

An adaptive model adapts itself by changing the probabilities it assigns to symbols. The it applies the new model to encode the new symbols.

4.5 Prediction by Partial Matching

Prediction by Partial Matching (PPM)(Cleary & Witten 1984, Bell et al. 1990) is a symbolwise compression scheme for adaptive text compression. PPM generates a prediction for each input character based on its preceding characters. The prediction is encoded in form of conditional probability, conditioned on previous context. PPM maintains predictions, computed from the training data, for larger context as well as all shorter con-texts. If PPM cannot predict the character from current context, it uses an escape probability to "escape" another context model, usually of length one shorter than the current context. For novel characters that have never seen before in any length model, the algorithm escapes down to a default "order-1" context model where all possible characters are present.

PPM escape method can be considered as an instance of Jelinek-Mercer smoothing. It is defined recursively as a linear interpolation between the nth-order maximum likelihood and the (n-1)th-order smoothed model. Various methods have been proposed for estimating the escape probability. In the following

description of each method, e is the escape probability and $p(\phi)$ is the conditional probability for symbol ϕ , given a context. $c(\phi)$ is the number of times the context was followed by the symbol ϕ . n is the number of tokens that have followed. t is the number of types.

Method A works by allocating a count of one to the escape symbol.

$$e = \frac{1}{n+1} \tag{4.11}$$

$$p(\phi) = \frac{c(\phi)}{n+1} \tag{4.12}$$

Method B makes assumption that the first occurrence of a particular symbol in a particular context may be taken as evidence of a novel symbol appearing in the context, and therefore does not contribute towards the estimate of the probability of the symbol which it occurred.

$$e = \frac{t}{n} \tag{4.13}$$

$$p(\phi) = \frac{c(\phi)-1}{n} \tag{4.14}$$

Method C (Moffat 1990) is similar to Method B, with the distinction that the first observation of a particular symbol in a particular symbol in a particular context also contributes to the probability estimate of the symbol itself. Escape method C is called Witten-Bell smoothing in statistical language modelling. Chen and Goodman (Chen & Goodman 1999) reported it is competitive on very large training data sets comparing with other smoothing techniques.

$$e = \frac{t}{n+t} \tag{4.15}$$

$$p(\phi) = \frac{c(\phi)}{n+t} \tag{4.16}$$

Method D (Howard 1993) is minor modification to method B. Whenever a novel event occurs, rather than adding one to the symbol, half is added instead.

Order 2				
Prediction			c	p
al	→	o	1	1/2
	→	Esc	1	1/2
de	→	a	2	3/4
	→	Esc	1	1/4
ea	→	l	2	3/4
	→	Esc	1	1/2
lo	→	r	1	1/2
	→	Esc	1	1/2
no	→	d	1	1/2
	→	Esc	1	1/2
od	→	e	1	1/2
	→	Esc	1	1/2
or	→	n	1	1/2
	→	Esc	1	1/2
rn	→	o	1	1/2
	→	Esc	1	1/2

Order 1				
Prediction			c	p
a	→	l	2	3/4
	→	Esc	1	1/4
d	→	e	2	3/4
	→	Esc	1	1/4
e	→	a	2	3/4
	→	Esc	1	1/4
l	→	o	1	1/2
	→	Esc	1	1/2
n	→	o	1	1/2
	→	Esc	1	1/2
o	→	d	1	1/4
	→	r	1	1/4
	→	Esc	2	1/2
r	→	n	1	1/2
	→	Esc	1	1/2

Order 0				
Prediction			c	p
	→	a	2	3/24
	→	d	2	3/24
	→	e	2	3/24
	→	l	2	3/24
	→	n	1	1/24
	→	o	2	3/24
	→	r	1	1/24
	→	Esc	7	7/24

Order –1				
Prediction			c	p
	→	A	1	$1/\|A\|$

Tab. 4.2: PPM model after processing the string *dealornodeal*

$$e = \frac{t}{2n} \tag{4.17}$$

$$p(\phi) = \frac{2c(\phi) - 1}{2n} \tag{4.18}$$

To illustrate the PPM compression modelling technique, Table 4.2 shows the model after string *dealornodeal* has been processed. In this illustration the maximum order is 2 and each prediction has a count c and a prediction probability p. The probability is determined from counts associated with the prediction using escape method D(equation 4.18). $|A|$ is the size the alphabet which determines the probability for each unseen character.

Suppose the character following *dealornodeal* is *o*. Since the order-2 context is *al* and the upcoming symbol *o* has already seen in this context, the order-2 model is used to encode the symbol. The encoding probability is 1/2. If the next character were *i* instead of *o*, it has not been seen in the current order-2 context (*al*). Then an order-2 escape event is emitted with a probability of 1/2 and the context truncated to *l*. Checking the order-1 model, the upcoming character *i* has not been seen in this context, so an order-1 escape event is emitted with a probability of 1/2 and the context is truncated to the null context, corresponding to the order-0 model. As *i* has not appeared in the string *dealornodeal*, a final level of escape is emitted with a probability of 7/24 and the *i* will be predicted with a probability of 1/256 in the order- –1, assuming that the alphabet size is 256 for ASCII. Thus *i* is encoded with a total probability of $\frac{1}{2} * \frac{1}{2} * \frac{7}{24} * \frac{1}{256}$.

In reality, the alphabet size in the order- –1 model may be reduced by the number of characters in the order-0 model as these characters will never be predicted in the order- –1 context. Thus it can be reduced to 249 in this case. Similarly a character that occurs in the higher-order model will never be encoded in the lower-order models. So it is not necessary to reserve the probability space for the character in the lower-order models. This is called "exclusion", which can greatly improve compression.

Table 4.3 shows the compression result for the file People's Daily (9101) with

Compression Method	Size	Compression Rate
Escape A(order 2)	434228	54.8%
Escape B(order 2)	332278	41.9%
Escape C(order 2)	333791	42.1%
Escape D(order 2)	332829	42.0%
Escape D(order 1)	345841	43.6%
Escape D(order 3)	332932	42.0%
gzip	434220	54.8%
compress	514045	64.8%

Tab. 4.3: Compression results for different compression methods

792964 Bytes using different compression methods. PPM compression methods are significantly better than practical compression utilities like Unix *gzip* and *compress* except escape method A but they are slower during compression. The compression rates for escape method B and D are both higher than escape method C. Order-2 model (trigram) is slightly better that order-1 and order-3 models for escape method D.

In our experiment we use escape method D to calculate the escape probability as escape method D is slightly better than other escape methods in compressing text although Method B is the best here. Teahan (Teahan, Wen & R. McNab 2000) has successfully applied escape method D to segment Chinese text.

4.6 Experiment and Result

We use 220MB People's Daily (91-95) as the training corpus and 58M People Daily (96) and stories download from Internet (400K) as the test corpus.

We used SRILM language tools (Stolcke 2002) to collect trigram counts and applied modified Kneser-Ney smoothing method to build the language model. Then we used *disambig* to translate Pinyin to Chinese characters. The *disambig* is a procedure in SRILM tools to disambiguate text tokens using an n-gram

model. It converses a stream of tokens from a vocabulary V_1 to a corresponding stream of tokens from a vocabulary V_2 according to a probabilistic 1-to-many mapping. Ambiguities in the mapping are resolved by finding the V_2 sequence with the highest posterior probability given the V_1 sequence. This probability is computed from pairwise conditional probabilities $P(V_1|V_2)$ and a language model for sequences over V_2.

In PPM model we used the same count data collected by SRILM tools. We chose a trie structure to store the symbol and count. Adaptive PPM model updates the counts during Pinyin input. It is similar to a cache model (Kuhn & De Mori 1990). We tested both static and adaptive PPM models on test corpus. PPM models run twice faster than SRILM tool *disambig*. It took 20 hours to translate Pinyin (People's Daily 96) to character on a Sparc with two CPUs(900Mhz) using SRILM tools. The following Table 4.4 shows the results in terms of character error rate excluding the start and end symbols of the sentences in the total number of characters (both included in the paper (Huang & Powers 2004)). People's Daily(96) is the same domain as the training corpus. Results obtained testing on People's Daily are consistently much better than Stories. Static PPM is a little worse than modified Kneser-Ney smoothing method. Adaptive PPM model testing on large corpus is better than small corpus as it takes time to adapt to the new model.

	People's Daily (96)	Stories
modified Kneser-Ney	14.45%	23.17%
Static PPM	14.90%	26.48%
Adaptive PPM	4.98%	14.24%

Tab. 4.4: Character Error Rates for Kneser-Ney, Static and Adaptive PPM

4.7 Conclusion

We have introduced a method for Pinyin input based on an adaptive PPM model. Compression-adaptive model builds statistical language model incrementally. Adaptive PPM model outperforms both static PPM and modified Kneser-Ney smoothing.

In this chapter we only compared the adaptive language model with modified Kneser-Ney smoothing method. In the next chapter we will extend our work to compare adaptive language model with all smoothing methods and propose a new approach based on statistical error-driven adaptive language modeling to Chinese Pinyin input system.

5. ERROR-DRIVEN ADAPTIVE LANGUAGE MODELING FOR PINYIN-TO-CHARACTER CONVERSION

The performance of Chinese Pinyin-to-Character conversion is severely affected when the characteristics of the training and conversion data differ. As natural language is highly variable, it is impossible to build a complete and general language model to suit all the tasks. The traditional adaptive MAP models mix the task independent data with task dependent data using a mixture coefficient but we never can predict what style of language users have and what new domain will appear. This chapter[1] presents a statistical error-driven adaptive language modeling approach to Chinese Pinyin input system. This model can be incrementally adapted when an error occurs during Pinyin-to-Character converting time. It significantly improves Pinyin-to-Character conversion rate.

5.1 Introduction

As described in previous chapter Chinese texts comprise ideographic and pictographic characters. Unlike English, these characters can't be entered by keyboard directly. They have to be transliterated from keyboard input based on different input methods. Pinyin phonetic input method is the most popular and widely used.

Statistical n-gram language models trained on large corpus have been successfully applied to Chinese Pinyin input (Gao et al. 2002). However language

[1] This chapter is based on the paper (Huang & Powers 2011)

Huang, J. & Powers, D. (2011), Error-driven adaptive language modeling for Pinyin-to-character conversion, in 'International Conference on Asian Language Processing (IALP2011)', Penang, Malaysia.

is highly variable and changing. The rapid development of Internet has created huge impact on the Chinese language. Everyone can notice the change of language in our daily life as new words keep emerging. People change their uses of language on different domains and occasions. Different domains tend to involve relatively disjoint concepts. The statical language model (SLM, LM) trained for one domain may not suitable for other domains. The performance of LM always suffers from such mismatch. The various techniques (Gao, Suzuki & Yuan 2006) have been proposed to adapt LM from one domain to a different domain: maximum a posteriori (MAP) (Bellagarda 2004), boosting (Collins 2000), perceptron (Collins 2002) algorithms and the minimum sample risk (MSR) method (J., Yu, Yuan & Xu 2005). MAP combines the general LM with a domain specific LM using a combination factor. The other three methods are called discriminative training methods. These methods try to minimize the conversion errors by optimizing the model parameters during training.

In this chapter we propose a MAP method to incrementally build SLM which can be used for training and adaption. Instead of training/adapting on all the data, we use an error function similar to discriminative training method to select what kind of data to integrate to the model. It makes the model adapt to the new data and reduces the character error rate.

5.2 LM Adaption Methods

In order to adapt a general SLM to a new domain, adaption data of the new domain is required to derive a domain-specific SLM called as dynamic model. The general SLM is also called as background or static model.

5.2.1 MAP Methods

There are two maximum a posteriori (MAP) estimation methods to combine the static and dynamic models depending on parameterization of the prior distribution: linear interpolation and count merging.

Linear Interpolation

Two SLMs are generated on the background data and adaption data respectively. The two models are combined into one as:

$$Pr(w_i|h) = \lambda P_B(w_i|h) + (1 - \lambda)P_A(w_i|h)$$

where λ serves as the interpolation coefficient. This parameter can be estimated on held-out data from the adaption data using the EM algorithm(Bellagarda 2004). P_B is the probability of the background model, P_A is the probability of the adaptation model and the history h corresponds to two preceding words for a trigram model.

Count Merging

As the name mentioned the combination is done at frequency count level rather than the model level. The combination can formulated as:

$$Pr(w_i|h) = \frac{\varepsilon C_A(hw_i) + C_B(hw_i)}{\varepsilon C_A(h) + C_B(h)}$$

where ε is a constant factor which is estimated empirically.

The performance of MAP methods depend on the way the adaption data is accumulated, the specific form of combination and the particular method selected for estimation and/or smoothing.

5.2.2 Discriminative Training Methods

All the three discriminative training methods follow the general framework of linear models (Collins 2002). The following definition is adapted from Gao(Gao, Suzuki & Yuan 2006).

- Training data are represented as $\{Ai, W_i^R\}$, for $i = 1 \ldots M$, where each A_i is an input phonetic string and W_i^R is the reference transcript of A_i.

- A set of candidate word strings is generated denoted by $GEN(A)$ given input A.
- A set of $D+1$ features $f_d(W)$, for $d = 0 \ldots D$, is defined to map W to real values. These features are defined as the counts of n-grams in W.
- A parameter λ_d is associated with each feature function $f_d(W)$. The score of a word string W can be formulated as:

$$Score(W, \lambda) = \lambda f(W) = \sum_{d=0}^{D} \lambda_d f_d(W)$$

$$W^*(A, \lambda) = argmax_{W \in GEN(A)} Score(W, \lambda)$$

  It views Pinyin-to-character as a ranking problem, producing the ranking score, not probabilities.
- An error function $Er(W_R, W)$ is defined to measure the number of conversion errors in W by comparing it with a reference transcript W_R.

Discriminative training methods apply different approaches to minimize the errors by optimizing the model parameters. We will not describe in details here.

5.3 Error-driven Adaption

In MAP estimation methods, adaptation data is used to adjust the parameters of the background model to maximize the likelihood of the adaptation data through linear interpolation or count merging. Instead of considering all the adaption data, we use a error function similar to the discriminative methods to control the merging. This error function can be considered as a merging function.

For each sentence in adaption data, we use the background model to generate candidate character string. Then we compare it with the reference transcript to measure the conversion errors. If the conversion is correct, no changes are done to the combination model. It means the background SLM matches the current adaption sentence for conversion. If an error occurs, it means the background

SLM mismatch the current data. An adaption has to be carried out. The sentence which the error occurs is added to to model. This is similar to discriminative methods which the counts of n-grams are defined as their feature functions. For example, the sentence in the adaption data is "I see the boys swim in the sea." but "I sea the boys swim in the sea." is recognized. The counts of all the correct n-grams such as "I", "I see", "I see the" etc will increase and those of the bad ones such as "I","I sea", "I sea the" etc will decrease. Then these feature functions are used to train the parameters to minimize the errors through multi-pass training.

We use Witten-Bell smoothing method in our experiments. Chen and Goodman (Chen & Goodman 1999) reported it is competitive on very large training data sets comparing with other smoothing techniques. Witten-Bell smoothing method is an adaptive statistical modeling technique that is widely used in the field of text compression. It is the escape method C shown in Section 4.5. It is able to build SLM very efficiently and incrementally. The advantage of Witten-Bell smoothing method is that we can use original counts in stead of log_2 probabilities to update SLMs. From Eq.3.19 and Eq.3.20 we derive the following equations for a trigram model:

$$\begin{aligned} P_{WB}(w_i|w_{i-2}w_{i-1}) &= \frac{C(w_{i-2}w_{i-1}w_i)}{C(w_{i-2}w_{i-1})+C} \\ &if \quad C(w_{i-2}w_{i-1}w_i) > 0 \qquad (5.1) \\ P_{WB}(w_i|w_{i-2}w_{i-1}) &= \frac{C * P_{WB}(w_i|w_{i-1})}{C(w_{i-2}w_{i-1})+C} \\ &if \quad C(w_{i-2}w_{i-1}w_i) = 0 \qquad (5.2) \end{aligned}$$

Where C is the number of distinct words that can follow $w_{i-2}w_{i-1}$ in the training data. $\frac{C}{C(w_{i-2}w_{i-1})+C}$ is called the escape probability to "escape" another context model, usually of length one shorter than the current context. For novel characters that have never seen before in any length model, the algorithm escapes down to a default "order- –1" context model where all possible characters are present.

To illustrate Witten-Bell modeling technique, we use the same example *dealornodeal* described in Section 4.5. Table 5.1 shows the model after string *dealornodeal*

Order 2

Prediction			c	p
al	→	o	1	1/3
	→	Esc	1	1/3
de	→	a	2	2/3
	→	Esc	1	1/3
ea	→	l	2	2/3
	→	Esc	1	1/3
lo	→	r	1	1/2
	→	Esc	1	1/2
no	→	d	1	1/2
	→	Esc	1	1/2
od	→	e	1	1/2
	→	Esc	1	1/2
or	→	n	1	1/2
	→	Esc	1	1/2
rn	→	o	1	1/2
	→	Esc	1	1/2

Order 1

Prediction			c	p
a	→	l	2	2/3
	→	Esc	1	1/3
d	→	e	2	2/3
	→	Esc	1	1/3
e	→	a	2	2/3
	→	Esc	1	1/3
l	→	o	1	1/2
	→	Esc	1	1/2
n	→	o	1	1/2
	→	Esc	1	1/2
o	→	d	1	1/4
	→	r	1	1/4
	→	Esc	2	1/2
r	→	n	1	1/2
	→	Esc	1	1/2

Order 0

Prediction			c	p
	→	a	2	2/19
	→	d	2	2/19
	→	e	2	2/19
	→	l	2	2/19
	→	n	1	1/19
	→	o	2	2/19
	→	r	1	1/19
	→	Esc	7	7/19

Order –1

Prediction			c	p
	→	A	1	$1/\|A\|$

Tab. 5.1: Witten-Bell smoothing model after processing the string *dealornodeal*

has been processed. In this illustration the maximum order is 2 and each prediction has a count c and a prediction (conditional) probability p. $|A|$ is the size the alphabet which determines the probability for each unseen character.

Suppose the character following *dealornodeal* is o. Since the order-2 context is al and the upcoming symbol o has already seen in this context, the order-2 model is used to encode the symbol. The encoding probability is 1/2. If the next character were i instead of o, it has not been seen in the current order-2 context (al). Then an order-2 escape event is emitted with a probability of 1/2 and the context truncated to l. Checking the order-1 model, the upcoming character i has not been seen in this context, so an order-1 escape event is emitted with a probability of 1/2 and the context is truncated to the null context, corresponding to the order-0 model. As i has not appeared in the string *dealornodeal*, a final level of escape is emitted with a probability of 7/19 and the i will be predicted with a probability of 1/256 in the order- –1, assuming that the alphabet size is 256 for ASCII. Thus i is encoded with a total probability of $\frac{1}{2} * \frac{1}{2} * \frac{7}{19} * \frac{1}{256}$. In this example the Witten-Bell smoothing makes better prediction of i than escape method D with a probability of $\frac{1}{2} * \frac{1}{2} * \frac{7}{24} * \frac{1}{256}$.

As we use the integer counts instead of log_2 probabilities, this make it likely that they will overflow when the model adapts bigger. A easy way to prevent this is to half all counts when the size of learning data threatens to exceed the maximum allowed number. It is important to make sure that none of the counts is left at zero either by rounding up or by deleting the n-grams. This count scaling (Moffat 1990) can be used for model reduction. The process of adaption is to learn new data and diminish the outdated data.

5.4 Experiment and Result

We use 220MB People's Daily (91-95) as the training corpus and 58M People's Daily (96), modern novel (500K), martial art novel (2M), 34.4M Xinhu(95), 22.7M Xinhua(96) as the test corpus.

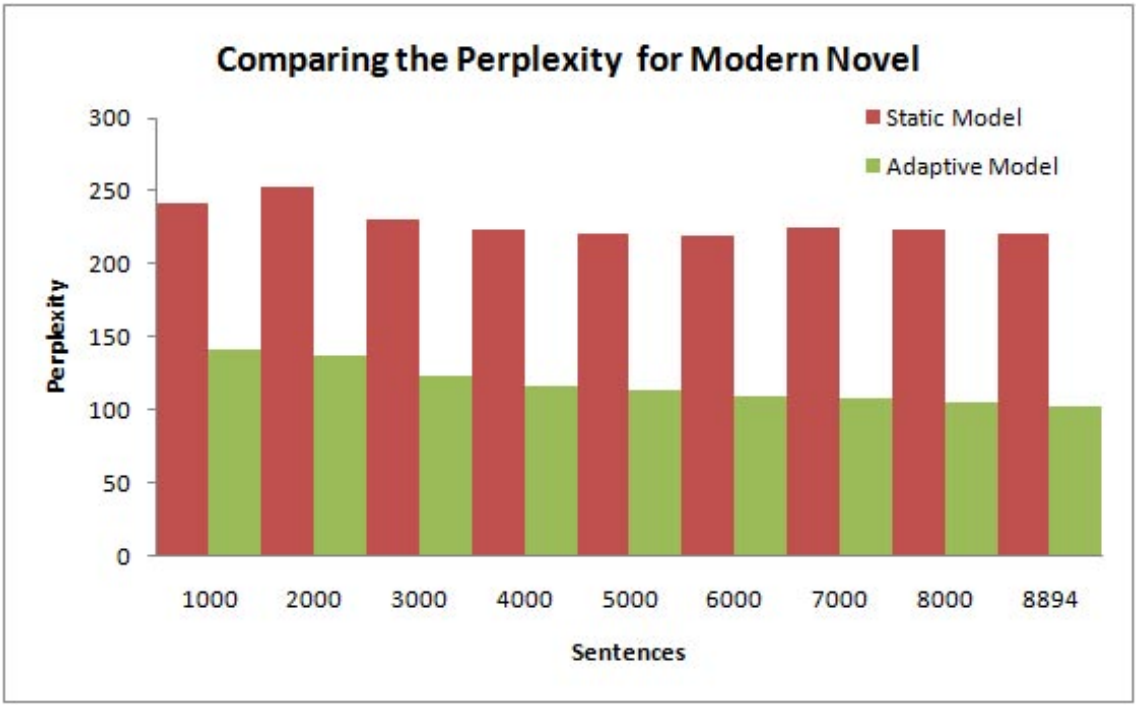

Fig. 5.1: Perplexity compare between static and adaptive model on Modern Novel

We used SRILM language tools (Stolcke 2002) to collect trigram counts and applied different smoothing methods to build the corresponding SLMs. Then we used *disambig* to convert Pinyin sequences to Chinese characters. The *disambig* is a procedure in SRILM tools to disambiguate text tokens using an n-gram model. Table 5.2 shows there is not a smoothing method significantly better than others on CER although there is a great different in perplexity. Only Add one smoothing is slightly worse than others.

In error-driven adaptive model we used the same count data collected by SRILM tools. We chose a trie structure to store the symbols and counts. The adaptive model updates the counts according to the error function after Pinyin-to-Character conversion. It is similar to a cache model (Kuhn & De Mori 1990). We tested both static and adaptive models on test corpus. Fig.5.1,Fig.5.2 and Fig.5.3 show there is a significant reduction in perplexity particularly on a different domain such as Martial Art Novel. The two-tail p values for paired samples t-test are $1.5901E-57$ (People's Daily), $2.15731E-11$ (Modern Novel) and

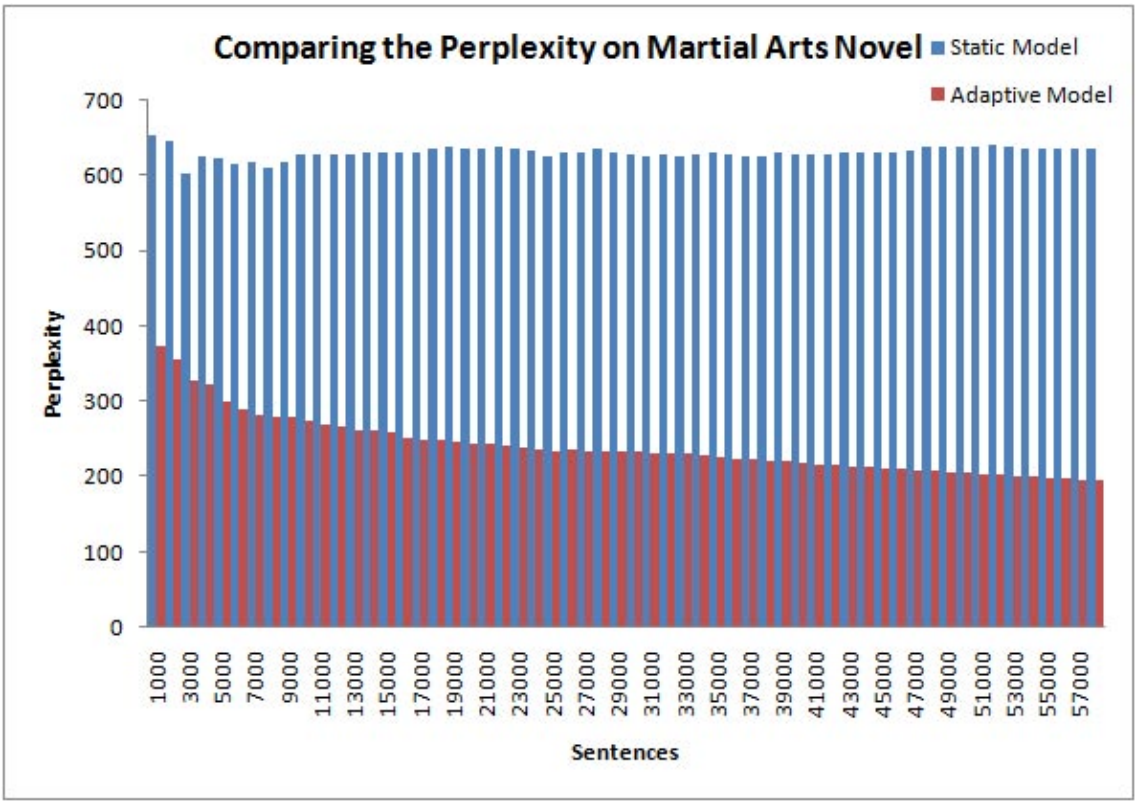

Fig. 5.2: Perplexity compare between static and adaptive model on Martial Arts Novel

	People's Daily		Modern Novel		Martial Arts	
	Perp.	CER	Perp.	CER	Perp.	CER
Add One	42.56	14.34	199.39	26.92	765.37	41.43
Absolute	40.77	13.71	174.26	25.51	624.76	40.49
Good-Turing	40.85	13.76	175.35	25.52	635.65	40.44
Witten-Bell	40.66	13.71	163.26	25.65	541.78	40.43
Modified KN	39.16	14.45	144.56	25.08	453.09	40.70
Kneser-Ney	38.69	14.49	146.12	25.22	471.98	40.54
Error-driven	37.99	05.24	102.79	11.80	196.30	23.38

Tab. 5.2: Comparing perplexity and CER using different smoothing methods on testing corpus

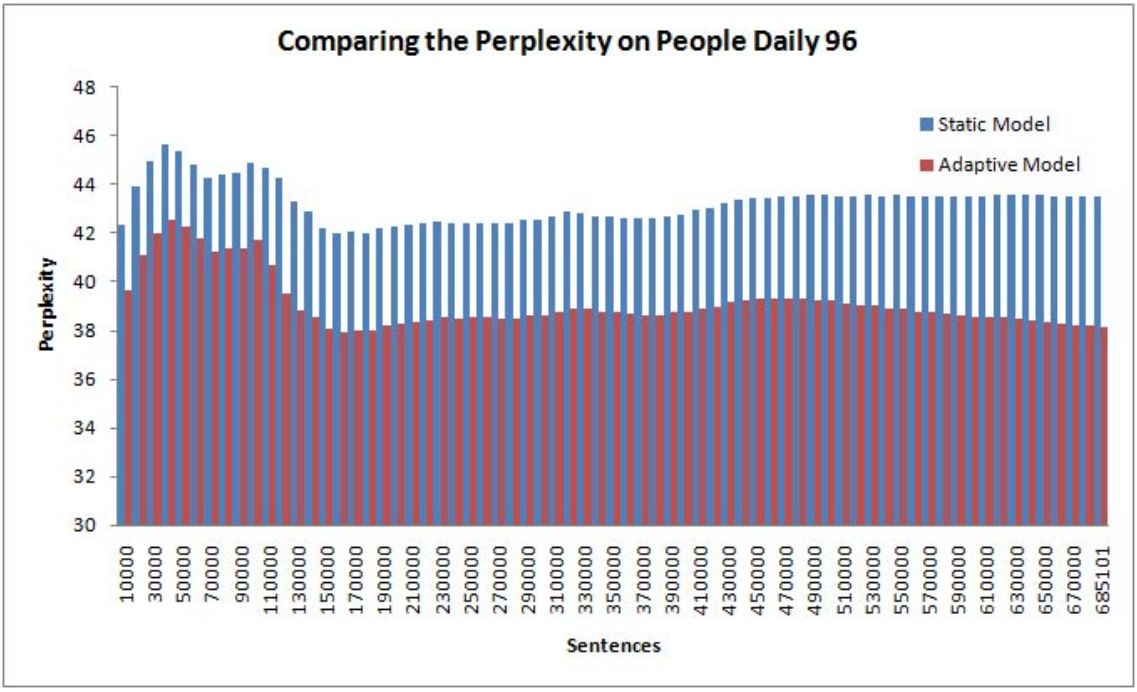

Fig. 5.3: Perplexity compare between static and adaptive model on People's Daily 96

	Error Driven	Full Adaption	Data Used
People's Daily	5.24%	5.08%	52.31%
Modern Novel	11.80%	10.71%	66.57%
Martial Arts	23.38%	21.26%	82.04%

Tab. 5.3: CER and percentage of data used for adaption

1.44383E-57 (Martial Arts). Table 5.2 shows there is around 50% reduction in term of character error rates even in the similar domain People's Daily 96 comparing with other static models.

Our results are comparable with other results although testing on different data. Wang el at reported 5.3% character error rate using a rough set technique (Xiaolong et al. 2004). Li et al achieved 11.46% CER using conditional random fields method(Li et al. 2009). Xiao et al obtained 5.82% CER by a class-based maximum entropy Markov model approach (Xiao et al. 2007). Gao et al realized the best result of 2.73% based on the minimum sample risk (MSR) method (Gao, Suzuki & Yuan 2006).

Models	CER(Xinhua 96)
Background model (PD 91-95)	13.27%
Mixed model (PD 91-95 + Xinhua 95)	4.44%
Error-driven model (PD 91-95 + 47.55% Xinhua 95)	4.46%

Tab. 5.4: Testing on Xinhua 96 with different mixed models with adaption

Table 5.3 shows the percentage of data used for adaption. The less conversion errors occur the less data is used for adaption. The full adaption model use all data to adapt itself. Although it performs a little better than error-driven adaptive model, it may risk over-fitting the models because of redundancy.

Table 5.4 shows the results of different mixed models testing on the Xinhua 96 without adaption. We use the model trained on the People's Daily 91-95 as the background model and the Xinhua 95 as the adaption data. The mixed model merges both counts of the PD 91-95 and the Xinhu 95. The error-driven model only uses 47.55% of the Xinhua 95 to mix with the background model and achieves the equivalent result as the mixed model.

5.5 Conclusion

We have introduced a method for Pinyin-to-Character conversion based on an error-driven adaptive model. The model can be self-adapted to new domain and improves conversion rate. It can incrementally build models using incoming data for adaptation. Pinyin input is an interaction process between users and system of Pinyin input method. Users always correct characters from the Pinyin input system after conversion. This correction can be served as a reference transcript for the error function to adapt the model.

Part II

CHINESE WORD SEGMENTATION AND CLASSIFICATION

6. CHINESE WORDS AND CHINESE WORD SEGMENTATION

In this chapter we present the problems in Chinese word definition and Chinese word segmentation. Then we review the current research work on Chinese word segmentation.

6.1 Introduction

Words are not readily recognizable because Chinese orthography does not show word boundaries. The definition of Chinese character does not cause any controversy because visually a character is an isolated symbol.

In the Chinese writing system, the characters are monosyllabic, each usually corresponding to a spoken syllable and a morpheme with a basic meaning. However, although Chinese words may be formed by characters themselves with basic meanings, a majority of words consist of two or more characters (thus are polysyllabic) but have meaning that is distinct from but dependent on the characters they are made from, like an English collocation or compound word. However there are exceptions that syllables are simply used to represent sounds in the foreign origin words from other languages such as 咖啡(ka fei, coffee), 巧克力(qiao ke li, chocolate) and 澳大利亚(ao da li ya, Australia).

6.2 The Definition of Chinese Word

Despite of the fact that words are written continuously without any delimiters, Chinese people can agree on most of the boundaries of words according to con-

text. However, the definition of Chinese words is inherently ambiguous. Both boundary between character and word and between word and phrase are fuzzy in Chinese. A compound in English can be easily identified since it contains more than one space-separated words, but it is hard to distinguish a compound phase from a simple word in Chinese. For example, 教育部（education department）may either be identified as a simple word or as a compound, that is 教育(education) + 部(department). In other cases, a derived word is normally a single word in English, but it may be identified as a compound in Chinese. For instance, scientist (科学家) is derived from science to represent the professional who performs research in science. In Chinese the same meaning is formed by combining 科学(science) and 家(a professional in certain field). Even in English, some cases of where to put word boundaries are just orthographic conventions such as notwithstanding vs. not to mention or into vs. out of.

"What is a Chinese word" is still an open issue in Chinese linguistics. The word itself is sometimes difficult to define, even in languages whose word boundaries are marked in the text. This is mainly due to the fact that there are multiple identities or senses of "word" (Packard 2000). Many different criteria have been proposed. For example, Packard listed eight different criteria:

1. Orthographic: A language unit defined by writing conventions. Chinese orthography segments written text into characters.

2. Sociological: The basic linguistic language unit intuitively recognized by native speakers. In Chinese, the sociological word is the 字(zi), meaning either the written character or the spoken morpheme.

3. Lexical: An entry in the lexicon. It is impossible to cover all the words in a lexical dictionary.

4. Semantic: A unitary concept intuitively felt but no uniquely definable either within or among speakers.

5. Phonological: A "word-sized" entity that is defined using phonological criteria such as phonological pause, tone and stress.

6. Morphological: An output of a word-formation rule.

7. Syntactic: An independent occupant of a syntactic form class slot.

8. Psycholinguistic: A construct at roughly the "word" level of linguistic analysis that is salient and highly relevant to language processing.

In English, the sets of words defined by different criteria overlap to a very high extent. In Chinese, however, different criteria could result in very different kinds of "words". For example, the first two criteria basically define the set of characters as words. The sets of units defined by the first two criteria are explicitly accessible. On the other hand, the units defined by the remaining six criteria are more or less in the form of implicit knowledge in the speakers' mind.

Packard argues that the syntactic criterion of "word" is the best, because it is the most common current linguistic characterization of the notion "word", and that it motivates the concept of "word" in most other languages. Moreover, the sets of words defined by the syntactic criterion and by the Chinese technical term for "word" (词, ci) overlap to a very large extent. This coincides with widely accepted Chinese word definition by Zhu (Zhu 1982): "A word is a minimal linguistic entity that is both meaningful and independent".

As a practical matter this probably doesn't matter. After all, the definition of word that is most useful will depend to a large degree upon what one wants to use it for. In Chinese language processing, a phonological word is likely to be of interest if one is interested in an application such as text-to-speech synthesis; if one is interested in machine translation, then it is likely that one would be looking more at semantic words; and if one is constructing lexicons, then some notion of lexical word will be relevant. It really doesn't matter that these are different. However, these different criteria will have the impact on the way to segment the Chinese text (Xu, Zens & Ney 2004)(Gao, Wu, Li, Huang, Li, Xia & Qin 2004)(Dong et al. 2010).

6.3 Chinese Word Segmentation

Although it is controversial in defining what encompasses a word in Chinese, computation and analysis of Chinese text will be impossible without transforming strings of characters into strings of words. The process of breaking text into words is called word segmentation. The problem of finding words in Chinese is analogous to the problem of identifying collocations in English, such as "put up with" or "object oriented".There are two major obstacles for Chinese word segmentation: segmentation ambiguity and unknown words.

6.3.1 Segmentation Ambiguity

Many characters can stand alone as words in themselves, while on other occasions it can be a component of other words at any position. This phenomenon causes obvious ambiguities in word segmentation particularly without inflection. There are mainly two types of segmentation ambiguities.

Overlapping ambiguity

For a string ABC, if AB and BC are both possible words, then ABC shows overlapping ambiguity.

For example, 美国会can be segmented as 美国(America) and 会(will) or 美(American) and 国会(congress) in the following sentences.

美国会打击恐怖分子
America will fight terrorists.
美国会正在讨论预算.
American Congress are discussing budget.

Combinational ambiguity

For a string AB, if A, B, and AB are all possible words, then AB shows combinational ambiguity. For example, 将来can be segmented as 将来(in the future)

or 将(will) and 来(come) in the following sentence.

他将来这里.

He will come here.

他将来要当老师.

He want to be a teacher in the future.

Some ambiguities include both overlapping and combinatorial ambiguity. For examples, out of the string 太平淡(too dull), 太平(peaceful), 平淡(dull), 太(over), 平(flat), 淡(plain) are all possible words.

Despite the multiple segmentation possibilities, there is in reality only one way to segment the sentence in question.

6.3.2 Unknown Words

Every language has a large vocabulary to meet the demands for social communication. However, the words in the vocabulary of a certain language are constantly changing instead of being invariable. With the rapid development of society, technology, Internet and the universal promotion of the globalization, new words constantly appear to meet the need of expressing new ideas or naming new products and new organizations. The 2010 Beijing Language Situation in China released by the National Language Committee, Chinese Ministry of Education reported 500 new words created in 2010. These kind of new words are impossible to be covered by dictionaries. Unknown (out-of-vocabulary, OOV) words fall into four major categories: new words, proper names, derived words and factoid words.

1. New words: They are newly coined words, occasional words, and mostly time-sensitive words. For example, 房奴 (houseslaves), 山寨 copycatting, 团购 (group purchase), 胶囊公寓 (capsule apartment), 富二代 (the rich second generation) and 给力(awesome).

2. Proper names: These include acronyms, Chinese names, location names, organization names and those words that have been borrowed from other

languages: for example, 北大(Peking University) 王府井（Wang Fu Jing, a street in Beijing), 胡锦涛(Hu Jin Tao, president of China), 威廉王子(Prince William) and 谷歌(Google).

3. Morphologically derived words: These are words that have affix morphemes: for example, 现代化(modernization) and 可视化(visualization) both of which contain affix morpheme 化.

4. Factoid words: These are numeric-type words such as time and date expressions. For example 2011年 (year 2011), 第一集 (The first episode), 五千多年 (more than 5000 years), 百分之一 (one percentage) and 5点10分 (ten pass five o'clock). Although these words have specific meanings and are used frequently, most dictionaries do not contain them.

It has recognized that the impact of unknown words on accuracy of word segmentation is much larger than that of segmentation ambiguities (Zhao & Kit 2011).

6.4 Segmentation Standards

Defining what is a word in Chinese is not purely a theoretical debate. It has practical consequences for a wide variety of applications. It is desirable to have a standard to evaluate whether a text is segmented correctly.To address this problem, there have been at least four relative widespread standards (Penn Treebank, China, Taiwan, Hongkong).

1. The segmentation guidelines for the Penn Chinese Treebank (Xia 2000)

2. The guidelines for the Beijing University Institute of Computational Linguistics Corpus (Yu 1999)

3. The ROCLING standard developed at Academia Sinica in Taiwan (Huang, Chan, Chang & Chen 1997).

Form	UPenn	Mainland	ROCLING	Example
ABAB	ABAB	AB-AB	ABAB	研究研究 'research (a bit)'
AA-看	[AA/V kan/V]/V	AA kan	AA kan	说说看 'talk about it and see'
Pers. Names	One Seg	Two Segs	One Seg	温家宝 'Wen Jiabao'
Noun+们	One Seg	Two Segs	Two Segs	朋友们 'friends'
Ordinals	One Seg	Two Segs	Two Segs	第一 'first'

Tab. 6.1: Some differences between the segmentation standards

4. The segmentation guidelines from City University of Hongkong(Guidelines 2005)

There is some disagreement on word segmentation amongst these standards. A detailed comparison between these standards is beyond the scope of this discussion, but it is useful to consider a few differences in Table 6.1 adapted from (Xia 2000).

6.5 Current Research Work

Several approaches have been developed for Chinese word segmentation. In general Chinese word segmentation can be classified into two categories: lexical rule-based methods (Yeh & Lee 1991, Palmer 1997) and statistical machine learning methods(Lua & Gan 1994, Sproat & Shih 1990, Sproat et al. 1996, Teahan et al. 2000, Peng & Schuurmans 2001).

The major concerns of lexical rule-based approach are how to deal with ambiguities in segmentation, and how to extend the lexicon beyond dictionary entries. The lexical rule-based approach is also known as the dictionary-based approach.

The most successful dictionary based methods are variations of the maximum matching algorithm, which greedily searches through a sentence in an attempt to find the longest string starting from a given point in the sentence that matches a word entry in a pre-compiled dictionary (Nie, Jin & Hannan 1994). Typically, a dictionary-based approach addresses the ambiguity problem with some lexical, syntactic or semantic heuristics. There exist two kinds of ambiguities in Chinese word segmentation when using the dictionary based approach: overlapping ambiguity and combination ambiguity. Overlapping ambiguity can be detected by a mismatch from forward maximum matching (FMM) and backward maximum matching (BMM), whereas combination ambiguity can be detected by an uncertain decision to split a character sequence when both the whole character sequence and all its members exist in the dictionary (Tsai 2006*a*). To solve the ambiguity problem well, many techniques have been developed, including various kinds of statistical learning methods (Qiao, Sun & Menzel 2008)(Luo, Sun & Tsou 2002)(Li, Gao, Huang & Li 2003). The performance of dictionary-based methods largely depends upon the coverage of the dictionary. However, it is difficult to compile a complete dictionary due to the appearance of out-of-vocabulary (OOV) words. Fu and Wang (Fu & Wang 1999) tackled OOV word detection based on four word-formation patterns and head-middle-tail structures. Other researchers turn to statistic-based methods to better deal with OOV words and OOV detection. Statistical dictionary-based approaches attempt to get the best of both worlds by combining the use of a dictionary and statistical information. There are two strategies that handle OOV word detection. One strategy handles it separately after segmentation (Chen 2003)(Wu & Jiang 2000). Chen (Chen 2003) assumes that OOV words are usually two or more characters long and are often segmented into single characters. He then uses different components to detect OOV words of different types in a cascaded manner after the basic word segmentation. The other treats it as part of the segmentation (Sproat et al. 1996)(Gao et al. 2005)(Jiang, Guan & Wang 2006). Gao et al (Gao et al. 2005) segmented known words and detected unknown words of different types simultaneously based on the framework of linear mixture models. Jiang et al (Jiang et

al. 2006) divided OOV word recognition into several subtasks and tackle different subtasks with more suitable models based on maximum entropy and conditional random fields.

Statistical machine learning approaches can fall into unsupervised and supervised segmentation. Unsupervised segmentation does not rely on any given language resource such as a pre-defined vocabulary or a pre-segmented corpus. It is intended to estimate the likelihood of a substring being a word by capturing the empirical observations of language characteristics in real data. Sproat and Shih (Sproat & Shih 1990) used mutual information as a statistical measurement of the dependency among characters for word segmentation. De Marcken (de Marcken 1996) applied an minimum description length (MDL) framework and a hierarchical model to learn a word lexicon from raw text for segmentation. Sun et al (Sun, Shen & Tsou 1998) combined mutual information with the difference of t-score between characters. Peng and Schuurmans (Peng & Schuurmans 2001) applied expectation maximization algorithm to learn a probabilistic model of character sequences for segmentation. We (Huang & Powers 2003) proposed contextual entropy of bigrams to discover word boundaries for segmentation. Jin and Tanaka-Ishii (Jin & Tanaka-Ishii 2006) extended contextual entropy to n-grams. Feng et al (Feng, Chen, Kit & Deng 2004) adopted the number of distinct predecessors and successors of a string as the measurement of the context independence of a string for word segmentation. Zhao and Kit (Zhao & Kit 2008*b*) exploited unlabeled text with different unsupervised segmentation criteria for Chinese word segmentation.

Supervised segmentation assumes to train a statistical model on a pre-segmented corpus infer the optimal segmentation for text. Its purpose is to determine the parameters in association with word sequences and the optimal parameters of the model are to be obtained via training. The statistical dictionary-based approach is a form of supervised segmentation. Sproat et al (Sproat et al. 1996) used a stochastic finite-state model for segmentation. Teahan et al (Teahan et al. 2000) trained a compression-based n-gram model for segmentation. Zhang et al (Zhang, Liu, Cheng, Zhang & Yu 2003) used a hierarchical hidden Markov

model incorporating lexical knowledge. Gao et al (Gao, Li & Huang 2003) applied class-based source-channel models to combine word category information for word segmentation.

In recent years the character-based tagging method, another form of supervised segmentation methods, has drawn great attention. Xue (Xue 2003) pioneered this method via maximum entropy (MaxEnt) modeling (Low, Ng & Guo 2005). Various learning models such as support vector machines (Li, Huang, Gao & Fan 2004), maximum entropy (SVMs), perceptron (Li, Miao, Bontcheva & Cunningham 2005, Zhang & Clark 2007), conditional random fields (CRFs) (Peng, Feng & Mccallum 2004)(Zhao et al. 2010)(Tseng, Chang, Andrew, Jurafsky & Manning 2005) have been employed within this framework. Variations of character-based tagging methods particularly based on conditional random fields have dominated the Chinese word segmentation Bakeoffs (international segmentation contest) (Sproat & Emerson 2003)(Emerson 2005)(Levow 2006)(Jin & Chen 2008)(Zhao & Liu 2010). Zhao and Kit (Zhao & Kit 2011) enhance the performance by integrating unsupervised and supervised word segmentation together.

Supervised approaches require a pre-defined vocabulary (dictionary or lexicon) or pre-segmented corpus for training. The coverage of the dictionary or pre-segmented corpus is critical for these approaches. As the pre-segmented corpus will never cover the OOV words, OOV words still remain an unsolved issue. Unsupervised approaches estimate the likelihood of a string being a word by using global statistics derived from a large scale unsegmented corpus. These approaches intend to derive a vocabulary from scratch with little human intervention. It is more effective to solve the segmentation errors of OOV words but supervised approaches tend to achieve better results for in-vocabulary words.

6.6 Conclusion

In this chapter, we have presented the problems in Chinese word definition and word segmentation. We have reviewed current research work on Chinese word

segmentation for the next chapter.

7. CHINESE WORD SEGMENTATION BASED ON CONTEXTUAL ENTROPY

Chinese is written without word delimiters so word segmentation is generally considered a key step in processing Chinese texts. This chapter[1] presents a new statistical approach to segment Chinese sequences into words based on contextual entropy on both sides of a bigram. It is used to capture the dependency with the left and right contexts in which a bigram occurs. Our approach tries to segment by finding the word boundaries instead of the words. Experimental results show that it is effective for Chinese word segmentation.

7.1 Introduction

Unlike English there is no explicit word boundary in Chinese text. Chinese words can comprise one, two, three or more characters without delimiters. But almost all techniques to Chinese language processing, including machine translation, information retrieval and natural language understanding are based on words. Word segmentation is a key step in Chinese language processing.

It has been long known that contextual information can be used for segmentation (Harris 1955). Dai, Kgoo and Loh (Dai, Kgoo & Loh 1999) used weighted document frequency as contextual information for Chinese word segmentation. Zhang, Gao and Zhou (Zhang, Gao & Zhou 2000) used the context dependency for word extraction. Tung and Lee (Tung & Lee 1994) used contextual entropy

[1] This chapter is based on the paper (Huang & Powers 2003)

Huang, J. H. & Powers, D. (2003), Chinese word segmentation based on contextual entropy, in '17th Pacific Asia Conference on Language, Information and Computation', Singapore.

to identify unknown Chinese words. Chang, Lin & Su (Chang, Lin & Su 1995) and Ponte & Croft (Ponte & Croft 1996) used contextual entropy for automatic lexical acquisition. Hutchens & Alder (Hutchens & Alder 1998) and Kempe (Kempe 1999) used the contextual entropy to detect the separator in English and German corpus.

In this chapter we will present a simple purely statistical approach using contextual entropy for word segmentation. Details about our approach are given in section 7.2 and 7.3.

7.2 Contextual Entropy

We use a Markov model to estimate the probabilities of symbols of a corpus. The probability of a symbol w with respect to this model M and to a context c can be estimated by:

$$p(w|M,c) = \frac{f(w,M,c)}{f(M,c)}$$

The information of a symbol w with respect to the model M and to a context c is defined by:

$$I(w|M,c) = -\log_2 p(w|M,c)$$

The entropy of a context c with respect to this model M is defined by:

$$H(w|M,c) = \sum_{w\in\Sigma} p(w|M,c)I(w|M,c)$$

This entropy measures the uncertainty about the next symbol after having seen the context c. We call it contextual entropy. It will be low if one particular symbol is expected to occur with a high probability. Otherwise it will high if the model has no "idea" what kind of symbol will follow the context. Across a word boundary there is a significant increase in the contextual entropy as we are not sure what kind of character will appear after a word boundary.

Fig. 7.1 shows the contextual entropy and mutual information for sentence:

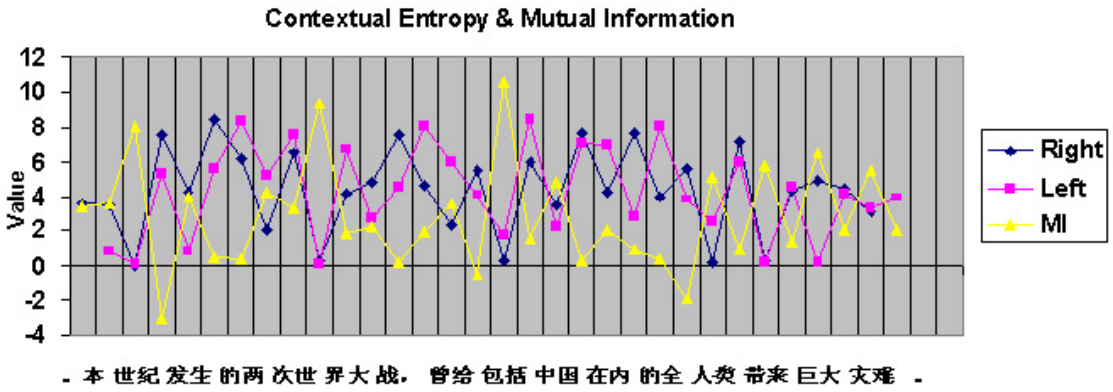

Fig. 7.1: Contextual Entropy and Mutual Information for "The two world wars happened this century had brought great disasters to human being including China."

ben shi ji fa sheng de liang ci shi jie da zhan
"本世纪发生的两次世界大战，
ceng gei bao han zhong guo zai nei de quan ren lei dai lai ju da zai nan
曾给包含中国在内的全人类带来巨大灾难"

Monitoring entropy in the Fig. 7.1 above shows regions of high entropy correspond with word boundary. Given the left context, a word boundary will follow the context. Given the right context, a boundary is followed by the context. In other words, the beginning and the end of a boundary are often marked by high entropy as any symbol can follow a boundary and occur before a boundary.

Contextual entropy finds a left boundary if there is a high branching factor (perplexity & choice) to the left and a right boundary if there is a high branching factor.

7.3 Algorithm

7.3.1 Contextual Entropy

To find Chinese words we look for character sequences that are stable in the corpus in the sense that the components of a word are strongly correlated but

appear in various contexts in the corpus. Contextual entropy among components of a word is low. High entropy appears at word boundaries.

We calculate both left and right contextual conditional entropy values for each bigram occurring in the corpus.

$$RH(x_1, x_2) = -\sum_{x_0 \in \Sigma} p(x_0|x_1, x_2) \log_2 p(x_0|x_1, x_2)$$

$$LH(x_1, x_2) = -\sum_{x_3 \in \Sigma} p(x_3|x_1, x_2) \log_2 p(x_3|x_1, x_2)$$

We only store the positive contextual entropy value. An entropy of zero indicates there is no boundary before or after the context given the right or left context. We assume the value for the bi-grams which do not appear in the corpus is zero as we can still predict the boundary according to the left or right adjacent context. This can save a lot of space to store bigrams with zero value.

From Figure 7.1 above we note that there is a word boundary at a peak for both entropy values. On the contrary there is no boundary at a trough. For a punctuation mark or a Chinese word marker, there is a peak preceding it given the right context and a peak following it given the left context. In other words, after having seen a punctuation mark or a word marker we do not know what occurs before and after it. This is very useful for detecting punctuation marks and word markers. Most other work did not treat the punctuation as an unknown character (Peng & Schuurmans 2001, Dai et al. 1999) or could not detect word markers well based on statistical methods (Ge, Pratt & Smyth 1999). They treated punctuation marks or characters as separators for sentences.

In order to segment the text we simply need to find the word boundaries. Across a word boundary there is a significant change in the contextual entropy. We apply the following thresholds to determine whether there is a word boundary between C and D for a string $ABCDEF$.

$$LH_{BC} - LH_{AB} > h_1 \tag{7.1}$$

$$LH_{BC} - LH_{CD} > h_2 \tag{7.2}$$

$$RH_{DE} - RH_{EF} > h_3 \tag{7.3}$$

$$RH_{DE} - RH_{CD} > h_4 \tag{7.4}$$

For each word markers or punctuation mark, there is a boundary before and after it. We call these function characters and apply the following thresholds to determine if C is a function character in the string $ABCDE$.

$$LH_{BC} - LH_{AB} > h_5 \tag{7.5}$$

$$LH_{BC} - LH_{CD} > h_6 \tag{7.6}$$

$$RH_{CD} - RH_{DE} > h_7 \tag{7.7}$$

$$RH_{CD} - RH_{BC} > h_8 \tag{7.8}$$

where LH is the left contextual entropy, RH is the right contextual entropy. $h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8$ are the threshold values.

$$LH_{BC} > h_9 \tag{7.9}$$

$$RH_{DE} > h_{10} \tag{7.10}$$

$$LH_{BC} - RH_{DE} > h_{11} \tag{7.11}$$

For a boundary between BC and DE, the contextual entropy given left context BC or right context DE are very high. We try to test whether there is a threshold for boundaries and non-boundaries.

7.3.2 Mutual Information

The work by Sproat and Shih (Sproat & Shih 1990) has a similar goal using a different measure, Mutual Information.

$$MI(x, y) = -\log_2 \frac{p(x, y)}{p(x)p(y)}$$

From Fig.7.1 we note that there is a high mutual information between characters in a word and a low mutual information across a boundary (Magerman &

Marcus 1990). Sproat and Shih (Sproat & Shih 1990) found the pair of adjacent characters with mutual information greater than some threshold (2.5) is a word and grouped them together. They iterated it until there were no more characters to group.

We formulate this in our model as well and consider it on its own and in combination with Contextual Entropy. Instead of just grouping characters together as a word we try to find the boundary between characters. We use Eq7.127.137.14 to test whether there is a minimum value at a boundary between C and D for a string $ABCDE$.

$$MI_{CD} < m_1 \tag{7.12}$$

$$MI_{BC} - MI_{CD} > m_2 \tag{7.13}$$

$$MI_{DE} - MI_{CD} > m_3 \tag{7.14}$$

where MI is the mutual information, m_1, m_2 are the threshold values.

7.4 Experiment Results

We trained the bi-directional 2nd order Markov model on 220MB corpora mainly news from People's Daily (91-95). We obtained about 1M pairs of bigrams with positive entropy. We stored the mutual information for the bigram at the same time.

In order to validate variations on our algorithm, we used a small corpus 100 articles of 325 articles from People Daily (94-98) included in the Penn Treebank Tagged Chinese Corpus (3.3M) to set the thresholds *h1 .. h11, m1 .. m3* and find the best way of combining these. Then we tested on the rest of the articles. We used recall and precision to measure our performance both on discovering word boundaries and words. A word is considered correctly segmented only if there is a word boundary in front of and at the end of the word and these is no boundary among the word. The following Table 7.1 7.2 7.3 7.4 show the testing result for our algorithms.

	Boundaries			Words		
	R	P	F	R	P	F
$1(h_1 = 0)$	75.8%	85.7%	80.5%	53.5%	60.5%	56.8%
$2(h_2 = 0)$	72.6%	84.8%	78.2%	43.5%	50.8%	46.9%
$3(h_3 = 0)$	73.0%	85.0%	78.6%	44.7%	52.1%	48.1%
$4(h_4 = 0)$	78.0%	87.5%	82.5%	56.2%	63.0%	59.4%
$AND(1,2,3,4)(h_{1...4} = 0)$	36.4%	96.0%	52.8%	17.5%	46.1%	25.3%
$OR(1,2,3,4)(h_{1...4} = 0)$	97.0%	77.1%	85.9%	75.7%	60.1%	67.0%
$OR(1,2,3,4)(h_{1...4} = 1)$	94.1%	82.5%	**87.9**%	76.1%	66.7%	**71.1**%
$OR(1,2,3,4)(h_{1...4} = 2)$	89.6%	87.0%	**88.3**%	72.3%	70.3%	**71.3**%
$OR(1,2,3,4)(h_{1...4} = 3)$	82.0%	90.7%	86.1%	63.9%	70.7%	67.1%
$AND(1,2)(h_{1...4} = 0)$	59.2%	90.0%	71.5%	29.2%	44.5%	35.3%
$AND(1,2)(h_{1...4} = 1)$	48.5%	94.3%	64.1%	22.0%	42.9%	29.1%
$AND(3,4)(h_{1...4} = 0)$	62.3%	93.3%	74.7%	33.8%	50.6%	40.5%
$AND(3,4)(h_{1...4} = 1)$	49.8%	96.1%	65.6%	25.1%	48.4%	33.1%
OR(AND(1,2),AND(3,4))h=0	85.0%	90.1%	**87.5**%	67.5%	71.5%	**69.5**%
OR(AND(1,2),AND(3,4))h=1	71.5%	94.0%	81.2%	50.6%	66.5%	57.4%

Tab. 7.1: Validation results based on Recall, Precision and F-Measure for Eq. 7.1 7.2 7.3 7.4

	Boundaries			Words		
	R	P	F	R	P	F
$AND(5,6,7,8) : 0(h_{5...8} = 0)$	29.9%	94.8%	45.5%	16.5%	52.3%	25.1%
$AND(5,6,7,8) : 1(h_{5...8} = 1)$	18.5%	**99.0**%	31.3%	9.6%	51.5%	16.2%

Tab. 7.2: Validation results based on Recall, Precision and F-Measure for Eq. 7.5 7.6 7.7 7.8

	Boundaries			Words		
	R	P	F	R	P	F
$9(h_9 = 3)$	89.4%	79.4%	84.1%	66.0%	58.6%	62.1%
$9(h_9 = 4)$	79.1%	85.5%	82.2%	57.4%	62.1%	59.7%
$10(h_{10} = 3)$	88.5%	79.0%	83.5%	63.1%	56.3%	59.5%
$10(h_{10} = 4)$	82.0%	86.9%	84.4%	60.7%	64.3%	62.5%
$OR(9, 10)(h_9, h_{10} = 3)$	98.4%	72.7%	83.6%	68.8%	50.8%	58.4%
$OR(9, 10)(h_9, h_{10} = 4)$	94.5%	80.7%	87.1%	73.9%	63.1%	68.1%
$OR(9, 10)(h_9, h_{10} = 5)$	86.6%	89.5%	**88.0**%	69.3%	71.6%	**70.5**%
$AND(9, 10)(h_9, h_{10} = 3)$	79.5%	89.1%	84.0%	61.6%	69.1%	65.1%
$AND(9, 10)(h_9, h_{10} = 4)$	66.6%	95.4%	78.5%	47.7%	68.3%	56.2%
$11(h_{11} = 6)$	94.2%	82.2%	87.8%	75.2%	65.6%	70.1%
$11(h_{11} = 7)$	90.2%	87.2%	**88.7**%	74.2%	71.7%	**72.9**%
$11(h_{11} = 8)$	85.0%	91.0%	87.9%	69.4%	74.3%	71.8%

Tab. 7.3: Validation results on Recall, Precision and F-measure according to Eq. 7.9 7.10 7.11

	Boundaries			Words		
	R	P	F	R	P	F
$12(m_1 = 2)$	70.1%	91.7%	79.5%	46.7%	61.1%	53.0%
$12(m_1 = 3)$	82.6%	88.9%	85.7%	62.8%	67.5%	65.1%
$12(m_1 = 4)$	90.0%	83.2%	86.5%	69.9%	64.7%	67.2%
$12(m_1 = 5)$	95.1%	77.6%	85.5%	71.6%	58.5%	64.4%
$13(m_2 = 0)$	84.7%	79.2%	81.8%	59.3%	55.4%	57.3%
$13(m_2 = 1)$	77.3%	82.6%	79.9%	51.9%	55.5%	53.7%
$14(m_3 = 0)$	89.8%	88.0%	88.9%	72.9%	71.5%	72.2%
$14(m_3 = 1)$	86.0%	91.4%	88.7%	69.4%	73.8%	71.5%
$OR(13, 14), m_{2,3} = 0$	94.3%	78.2%	85.5%	70.5%	58.5%	63.9%
$OR(13, 14), m_{2,3} = 1$	91.6%	82.6%	86.9%	70.2%	63.4%	66.6%
$OR(13, 14), m_{2,3} = 2$	86.6%	86.7%	86.6%	66.3%	66.4%	66.3%
$AND(13, 14), m_{1,2} = 0$	80.1%	90.6%	85.1%	61.2%	69.2%	65.0%
$AND(13, 14), m_{1,2} = 1$	71.7%	93.4%	81.1%	50.4%	65.6%	57.0%
$AND(13, 14), m_{1,2} = 2$	61.1%	95.4%	74.5%	37.0%	57.8%	45.2%

Tab. 7.4: Validation results based on Recall, Precision and F-Measure for Eq. 7.12 7.13 7.14

	Boundaries			Words		
	R	P	F	R	P	F
100 articles (Penn)	93.2%	93.1%	93.1%	81.1%	81.2%	81.1%
225 articles (Penn)	92.4%	93.3%	92.3%	80.4%	81.3%	80.8%
Peking corpus	91.5%	89.4 %	90.4%	76.8%	75.0%	75.9%

Tab. 7.5: Results based on Recall, Precision and F-Measure on testing corpus

From Table 7.1 we know there is a significant change in contextual entropy across a word boundary. Either side of contextual entropy change is useful to detect the word boundary. If we use F-measure:

$$F = \frac{2 * p * r}{p + r}$$

as a testing metric, using a threshold value around 2 with an "OR" relationship among Eq. 7.1 7.2 7.3 7.4 we achieve the best result for the validation corpus.

Table 7.2 shows properties of Eq. 7.5 7.6 7.7 7.8 are useful to detect a single character word marker in Chinese or punctuation. We obtained the highest precision under the four conditions. Table 7.3 shows using the sum (Eq. 7.13) of both left and right contextual entropy is better than either left (Eq. 7.11) or right (Eq. 7.12) contextual entropy . Table 7.4 shows the best threshold for grouping characters together is 4 for Penn Treebank corpus, greater than 2.5 that Sproat and Shih (Sproat & Shih 1990) used in their work.

Different criteria search the word boundaries from different aspects using different information sources. Combination of different criteria will enhance the recall of the whole segmentation. The higher precision of each criterion is, the higher accuracy of the whole segmentation will be achieved. To achieve higher F-measure of the whole segmentation, we have to increase the thresholds for each criterion to enhance the accuracy and combine the criteria to enhance the recall.

From the results above, the following conditions and thresholds we achieve the best results on the validation corpus (100 articles):

$$OR(AND(1,2), AND(3,4)) \quad (h_{1,2,3,4} = 2) \tag{7.15}$$

$$11 \quad (h_{11} = 9) \tag{7.16}$$

$$AND(5,6,7,8) \quad (h_{5,6,7,8} = 0) \tag{7.17}$$

$$AND(13,14) \quad (m_{2,3} = 3) \tag{7.18}$$

We obtained 93.2% precision with 93.1% recall on discovering word boundaries and 81.2% precision with 81.1% recall on discovering words. And we got 93.3%

precision with 92.4% recall on discovering word boundaries and 81.3% precision with 80.4% recall on discovering words. We tested on another corpus tagged by Peking University from People Daily (Jan 1998, 8.8M). We obtained 89.4% precision with 91.5% recall on discovering word boundaries and 75.0% precision with 76.8% recall on discovering words.

Sproat (Sproat & Shih 1990) obtained 94% precision and 90% recall but only considered the correctness of two-character words. Fu and Wang (Fu & Wang 1999) claimed 99.25% accuracy (recall, divided by total number of correct words in the test corpus) but did not provide precision. Peng and Schuurmans (Peng & Schuurmans 2001) used successive EM phases to learn a probabilistic model of character sequences and pruned the model with a mutual information selection criterion. They achieved 75.1% precision with 74.0% recall on discovering words by repeatedly applying lexicon pruning to an improved EM training. Their results are tested on the same corpus as ours. Compared with their approaches, our approaches are simpler, faster and achieved better results.

So far we have mainly comparing with unsupervised word segmentation. However unsupervised methods are not as effective as supervised learning methods. The dictionary based approach is the most popular Chinese word segmentation method. Given a pre-compiled dictionary, a heuristic method, such as longest word match, is then used to segment text. In our experiments we used the longest forward match method in which text is searched sequentially and greedily and the longest word that matches a word entry in a pre-compiled dictionary is taken at each successive location. A small dictionary (National-Language-Committee 2008) that contains 56,008 words and phrases is used as the pre-compiled dictionary. The coverage of the dictionary is 40.21% in the corpus from Peking University and 49.95% in the Penn Treebank corpus. A perfect lexicon (Ponte & Croft 1996) that consists of all and only words that occurred in the test corpus is used to compare the results. Table 7.6 shows the results of longest forward match method tested on both Treebank and Peking University corpus. We obtain best results 97% F-measure in the perfect lexicon experiments. However, in practice it is impossible to obtain such a perfect lexicon. In our small

	Boundaries			Words		
	R	P	F	R	P	F
Word Based (Penn)	97.75%	78.09%	86.82%	79.36%	63.40%	70.49%
Perfect Lexicon (Penn)	98.55%	99.64%	99.09%	97.15%	98.23%	97.68%
Word Based (Peking)	98.54%	82.60%	89.87%	82.97%	69.55%	75.67%
Perfect Lexicon (Peking)	98.50%	99.24%	98.87%	97.04%	97.78%	97.41%

Tab. 7.6: Results based on Recall, Precision and F-Measure for the Longest Forward Match Method

dictionary experiments, we achieve over 70% F-measure in both corpus. Our unsupervised approach is comparable to the dictionary based approach.

We had the same errors as Peng and Schuurmans (Peng & Schuurmans 2001) mentioned and had the same errors as most segmenters had to recognize the Chinese names. Most errors caused with our approaches relate to numbers and dates. In the training corpus, numbers written in full-width Arabic digits were replaced by a special character but in Penn corpus numbers are written in Chinese character. The other main kind of errors concerns compound nouns. We segmented "开发区"(development area) as "开发 区". But note that there is no standard definition for Chinese words as discussed in Section 6.2. It should be noted that there is poor agreement on word segmentation amongst human annotators and at least three relative widespread conventions (China, Taiwan, Penn Treebank) (GB/T13715-92 1993, Huang et al. 1997, Xia 2000). Our results are as expected lower than those judged by hand (which can bias judgements) and tested on non-standard corpora. The segmentation in Penn Treebank Chinese tagged corpus is also sometimes debatable and the different conventions differ in treatment of compound nouns and person names shown in Table 6.1. Penn Treebank segments "开发区"(development area) as one word and both China and Taiwan segment it as two words. An example in English of the arbitrariness of word segmentation is "cannot" vs "do not" or "into" vs "out of".

Although our approach only used a 2nd order Markov model, we still can find words longer than 2 characters as we only used our model to identify the word

boundaries rather than words.

7.5 Conclusion

This chapter describes a new approach for Chinese word segmentation based contextual entropy from an unsegmented corpus. Contextual entropy is used to capture the dependency with the both contexts in which a word occurs. We used a relative short order Markov model to train our model and tried to identify the word boundary rather than the word. Our approach is simple and fast, and although it is unsupervised it gives very competitive results.

7.6 Reflections

Many new machine learning algorithms (Su, Wang & Dai 2004, Liu 2005, Tanaka-Ishii & Jin 2006, Tanaka-Ishii & Jin 2008, Zhao & Kit 2008*b*, Zhao & Kit 2008*a*, Zhikov, Takamura & Okumura 2010, Zhao & Kit 2011) have been applied to Chinese word segmentation since this paper (Huang & Powers 2003) was published, but most algorithms are mainly supervised due to availability of large tagged corpora and the international contest on word segmentation Bakeoffs. Variations of conditional random field (Lafferty et al. 2001) based approaches have achieved great success in Chinese word segmentation Bakeoffs. But out-of-vocabulary issues remain in word segmentation and the existing lexical resources are hence never sufficient, unsupervised approaches never stop playing a role in word segmentation. Jin and Tanaka-Ishii (Jin & Tanaka-Ishii 2006) extended contextual entropy to n-grams for Chinese word segmentation. Recently Zhikov, Takamura and Okunmura (Zhikov et al. 2010) combined contextual entropy with minimum description length (MDL) for English, Thai and Japanese segmentation. Tang al et (Tang, Geva, Trotman & Xu 2010) used n-gram mutual information to detect word boundaries, similar to our approaches and achieved reasonable results in Chinese word segmentation Bakeoff 2010 (Zhao & Liu 2010). Several researcher have integrated unsupervised and supervised approaches together to improve the

performance of word segmentation (Zhao & Kit 2011, Xu, Zhu, Fei & Zhu 2010).

Our work didn't test on the SIGHAN shared task datasets mainly because our work predates the development of these datasets. Secondly, these pretaged datasets are prepared for supervised task.

8. UNSUPERVISED CHINESE WORD SEGMENTATION AND CLASSIFICATION

There are several problems encountered for Chinese language processing as Chinese is written without word delimiters. The difficulty in defining a word makes it even harder. This chapter[1] explores the possibility of unsupervised segmentation of Chinese character sequences into words and classification of these words through distributional analysis in contrast with the usual approaches that depend on dictionaries. With limitation of unsupervised word segmentation, it should be possible to improve segmentation by attuning the task of word classification. The results of word classification should be regarded as preliminary as it is a hard task to evaluate them in the absence of a benchmark.

8.1 Introduction

There is no explicit word boundary in Chinese text. Chinese orthography fails to represent word boundaries. The definition of a word is very important in Chinese language processing as these standards result in different segmentations and classifications. Chinese words are not inflected with respect to tense, case, person and number. As a result, Chinese word segmentation and classification is more difficult.

A given word, in a given syntactic context, has a grammatical role. A word such as "惩罚(Chengfa)" in Chinese can be translated into punish / punishes /

[1] This chapter is based on the paper (Huang & Powers 2002)

Huang, J. H. & Powers, D. (2002), Unsupervised Chinese word segmentation and classification, in 'First Student Workshop in Computational Linguistics', Beijing, China.

punished / punishing / punishment in English. We cannot tell the grammatical role solely based on the word, but have to look beyond the word. In other words, we must observe how the given word functions in its given context.

Zellig Harris (Harris 1951) proposed the substitutability of linguistic entity from the same class. Several researchers (Finch 1993, Brill 1993, Powers 1997*b*) have worked on learning grammatical properties of words on English. Languages with a less overt morphology like Chinese may be simpler to analyze than English since with fewer tokens per type, there is less data on which to base a categorization decision. For example, a noun or pronoun can be substituted in a Chinese sentence without regard to its number or gender. The verb remains unchanged whether the noun is singular or plural as the substitution is made possible by the fact that Chinese verbs are not conjugated.

In our experiments we explored the possibility of segmenting Chinese character sequences into words and classifying them by their syntactic distribution.

8.2 Word Classification

Part of speech is not well defined in Chinese and a dictionary can not include all words and all conceivable usages. Although Yu (Yu, Zhu, Wang & Zhang 1998) built the grammatical knowledge-based dictionary it only includes 70,000 words at this stage. And building such a dictionary is time consuming without machine assistance. Brown (Brown et al. 1992) argued that letting the machine infer the classes rather than relying on dictionaries or other human-derived artifact may result in more robust systems.

As with other natural languages, the same Chinese words can have more than one part of speech. Without the inflection we cannot tell the grammatical role solely based on the word. We have to observe how the given word functions in its given context through word order or particles. Chinese word "发展"(fazhan) can be a verb (develop) or a noun (development) in a sentence. In the following two sentences, the subjects are the phrases formed by "发展" and "经济"(jing

ji, economy) in different orders. In the first one, "发展" preceding the noun "经济" forms a verb-object compounds. "发展" is a verb. In the other one, "发展" following the noun "经济" forms a modified noun compounds. "发展" is a noun.

- 发展经济可以改善生活 Developing economy can improve life.
- 经济发展可以改善生活 Economic development can improve life.

The POS of "发展" in the sentence "经济发展很快" is legal but ambiguous without adding particles. It can be interpreted as

- 经济(的)发展很快 Economic development is very fast.
- 经济发展(得)很快 The economy is developing very fast.

Most approaches in the previous work in English (Finch 1993, Brill 1993, Powers 1997*a*, Clark 2002, Yang & Powers 2008) classify words instead of individual occurrences. Distributional similarity is often calculated in the high dimensional vector space model (VSM). The dimensionality of word space can be syntactically conditioned (i.e. grammatical relations) or unconditioned (i.e. 'a bag of words') (Clark 2001). They make the assumption that any given word will only belong to one category. Given the widespread part-of-speech ambiguity of words in Chinese this is problematic. Chang and Chen (Chang & Chen 1994) make the same assumption. They present a method to classify Chinese words automatically into a predetermined number of categories. They try to find a class assignment for word that maximizes the probability of a corpus. A bigram approximation of this would state that the estimated probability of the text T of length L is modeled as the product over each word of the probability of w given the inferred class along with probability of given the previous.

$$P(T) = \sum_{i=1}^{L} p(W_i|C_i)p(C_{i-1}|C_i) \tag{8.1}$$

They optimize the class assignment of words so that probability the text is maximized using a simulated annealing approach.

We adopt the substitutional approach in the experiments. Words of the same class are syntactically substitutable although they may not be appropriate semantically. Words in Chinese can frequently function in more than one lexical class. But in a given sentence, it is usually clear how a word is functioning. We construct the environments of each target word with respect to its left and right context. For the left context "深化(deepen)" and right context "改革(reform)", "足球(soccer) 职称(academic) 征管(collection) 这项(this) 渔村(fishing village) 训练(training) 新闻(news) 卫生(health) 铁路(railway) 体制(structure system) 体育(physical education) 税制(taxation) 水利(water conservancy) 商业(business) 企业(enterprize) 配套(complement) 农业(agriculture) 农村(village) 内部(internal) 林业(forestry) 科技(science and technology) 经济(economy) 金融(finance) 教育(education) 教学(teaching) 价格(price) 机构(organization) 各项(each) 高校(university) 高教(tertiary education) 党校(party school) 出版(press) 城市(city) 财政(finance and administration) 财税(finance and taxation) 部队(army)" can be substituted by each other in the middle. Sun (Sun 1998) suggests that the best watch window in collocation extraction is [-2, +1], [-3, +4], [-1, +2] for noun, verb and adjective respectively in Chinese. We use two words on each side of the concept as an environment for classification.

We use the following algorithms for word classification:

1. Segment raw text
2. Collect 5-grams for words
3. Produce context-word sets
4. Group words according to identical context and produce initial word classes
5. Merge word classes according to overlap
6. Replace the words with word classes in the contexts
7. Repeat 4,5,6 until no more classes can merge

8.3 Experiments and Future Work

We segmented the same $220MB$ corpora mainly news from People's Daily (91-95) based on contextual entropy described in the previous chapter and applied some rules for our model to complement our purely statistical model. We treated adjacent numbers and single word markers as words. We obtained segmented corpus with about 96% precision. We applied suffix array again to collect 5-gram word strings on the segmented corpus. We merged two classes if two classes overlap 75%. If the adjacent words are both common to two contexts, we merge even if there is only 50% overlap. From experiments we get more than 2000 classes and cannot merge any more. Following lists show some of our results. From lists we note words in the same class not only have syntactic similarity but also semantic similarity.

1. Classifiers: 元(Yan) 人次(per person) 人(person):平方米(square meter) 亩(a unit of area) 名(classifier for persons) 美元(US dollar) 件(piece) 家(classifier for family) 公斤(kg) 个(for nouns without specific classifier) 吨(ton)

2. Conjunctions: 及其(and) 及(and) 和(and) 各(various) 等(and so forth) 的(adjective marker) 与(and) 和(and)

3. Prepositions: 至于(as for) 因为(because (of)) 在(at) 由(due to) 因(because of) 为(for) 据(according to) 尽管如(in spite of) 鉴于(in the view of) 对(for) 从(from)

4. Number: 9 8 7 6 5 4 30 3 2 100 10 1 一(1) 五(5) 四(4) 三(3) 七(7) 六(6) 九(9) 二(2) 八(8)

5. Adverbs: 越(more and more) 很(very) 相当(quite) 十分(extremely) 日益(increasingly) 更为(more) 更加(much more) 比较(relatively)

6. Adjectives & Adverbs: 进一步(further) 继续向前(continuously forward) 继续(continuous,continuously) 不断(constant, constantly) 积极(active, actively) 大力(vigorous, vigorously)

7. Verbs 组建(organize) 组成(form) 展开(spread) 展出(exhibit) 运转(run) 运营(operate) 运行(move) 营业(business) 议案(motion) 移交(transfer) 邀请(invite) 选举(elect) 宣布(announce) 形成(form) 问世(come out) 推出(launch) 投产(put into production) 通航(operate air services) 通车(open for traffic) 停火(cease fire) 谈判(negotiate) 实行(practise) 实现(achieve) 实施(implement) 施行(put in force) 生效(take effect) 设立(set up) 上任(take a post) 上岗(take a job) 确认(confirm) 签字(sign) 签约(sign a contract) 签署(sign) 签订(sign) 启用(start using) 启动(start operating) 批准(approve) 落成(complete) 立项(set up) 抗议(protest) 开诊(open to see a patient) 开业(open a business) 开通(open a service) 开始(start) 开幕(inaugurate) 开馆(open a place for cultural or sports activities)

8. Nouns & Verbs: 祝贺(Congratulate) 致意(insist) 支持(support) 震惊(shock) 赞同(agree) 赞赏(admire) 赞成(assent) 忧虑(worry) 遗憾(regret) 欣慰(gratify) 欣赏(appreciate) 谢意(gratitude) 慰问(condole) 同意(agree) 钦佩(admire) 满意(satisfy) 理解(understand) 乐观(optimistic) 肯定(affirm) 敬意(respect) 敬佩(admire) 欢迎(welcome) 怀疑(suspect) 关注(pay attention to) 关心(care) 关切(concern) 高兴(happy) 感谢(thank) 愤慨(angry) 反对(oppose)

9. Nouns related government: 组织(organization) 主管(in charge) 种子(seed) 执法(law enforcement) 植保(plant protection) 职能(function) 政治(politics) 政府(government) 政法(political and legislative affairs) 渔政(fishery administration) 有关(related) 邮电(post) 医药(medicine) 宣传(propaganda) 行政(administration) 刑侦(criminal investigation) 信访(petition) 物价(price) 武装(arm) 文化(culture) 卫生(health) 土地(land) 统战(United front work) 统计(statistics) 体育(physical education) 体改(structure reform) 司法(judical administration) 税务(tax) 水利(irrigation) 审计(audit) 涉农(agriculture related) 商业(business) 人武(people's armed force) 人事(human resources) 侨务(overseas Chinese affairs) 气象(meteorology) 农资(agricultural capital) 农业(agriculture) 农牧(agriculture and husbandry) 农经(agriculture economics) 民政(civil administration) 旅游(travel) 领导(leader) 林业(forestry) 粮食(food) 劳动(labor)

10. City Names: 珠海(Zhuhai) 镇江(Zhenjiang) 宜昌(Yichang) 徐州(Xuzhou) 新疆(Xinjiang) 香港(Hongkong) 厦门(Xiamen) 西安(Xian) 武汉(Wuhan) 无锡(Wuxi) 温州(Wenzhou) 天津(Tianjin) 台州(Taizhou) 四川(Sichuan) 沈阳(Shenyang) 深圳(Shenzhen) 绍兴(Shaoxing) 上海(Shanghai) 莆田(Putian) 宁波(Ningbo) 南京(Nanjing) 梅州(Meizhou) 丽水(Lishui) 嘉兴(Jiaxing) 济南(Jinan) 吉安(Jian) 湖南(Hunan) 杭州(Hangzhou) 海南(Hainan) 贵阳(Guiyang) 桂林(Guilin)

11. Person Names: 佟志广(Tong Zhiguang) 邹家华(Zou Jiahua) 专家们(specialist) 朱基(Zhu Ji) 张震(Zhang Zhen) 张万年(Zhang Wannian) 张思卿(Zhang Siqing) 于永波(Yu Yongbo) 有人(somebody) 叶选平(Ye Xuanping) 叶利钦(Yeltsin) 杨主席(President Yang) 杨尚昆(Yang Shangkun) 杨福昌(Yang fuchang) 薛驹(XueJu) 宣言(declaration) 徐惟诚(Xu Weicheng) 徐敦信(Xu Dunxin) 消息(message) 西哈努克(Sihanouk) 吴作栋(Wu Zuodong) 吴仪(Wu Yi) 吴学谦(Wu Xueqian) 吴建民(Wu Jianmin) 吴邦国(Wu Bangguo) 文章(article) 文件(document) 温家宝(Wen Jiabao) 尉健行(Wei Jianxing) 王忠禹(Wang Zhongyu) 王震(Wang Zhen) 王兆国(Wang Zhaoguo) 王学贤(Wang Xuexian) 王汉斌(Wang Hanbin) 王丙乾(Wang Bingqian) 汪道涵(Wang Daohan) 万里(Wan Li) 通知(notice) 通报(brief) 田纪云(Tian Jiyun) 陶驷驹(Tao Siju) 她(she) 他们(they) 他(he) 俗话(proverb) 苏哈托(Suharto) 宋平(Song Ping) 宋健(Song Jian) 声明(announcement) 沈国放(Shen Guofang)

12. Incoherent Cluster(Pronoun, Proposition and Conjunction): 有人(somebody) 我(I,me)) 他们(they,them) 他(he,him) 甚至(even) 那种(that) 即(and) 或者(or) 还(also) 而(but)但(but) 并且(and) 并(and)

13. Incoherent Cluster(Verb and Proposition): 以(according to) 稳定(stabilize) 为(for) 提高(increase) 使(if) 实现(realize) 确保(guarantee) 求(request) 靠(by) 看(look) 解决(solve) 加强(strengthen) 加快(speed up) 加大(expand) 搞好(make better) 发展(develop) 对(at) 保证(promise) 把(proposition particle)

14. Verb Cluster for "通过": 通过(pass) 审议(examine) 批准(approve) 否决(veto) 大厦(building) 表决(vote) 实现(realize) 考虑(consider) 坚持(insist on) 发展(develop) 要求(request) 呼吁(appeal) 敦促(urge) 表示(express)

15. Proposition Cluster for "通过":在(at) 与(and) 用(by) 向(towards) 为(for) 通过(through) 因为(because of) 使(if) 靠(by) 对(at, versus) 从(from) 保持(maintain) 把(proposition particle) 由(due to) 同(with) 受到(be subjected to) 认为(consider) 就(with) 由于(because of)

In our approach we will get a large number of classes (clusters). One word maybe belongs to dozens of classes if it has a range of meanings and usages. How to merge the classes is critical in our approach. Many sub-classes embody finer distinctions among them. Some rare words are difficult to classify because of lack of distributional evidence. Bad word segmentation also contributes some errors. Some classes are pure. The class 11 (person names) includes pronouns and nouns related to "notice". It is possible that these words can be used in the subject acting as a person such as "通知说"(the notice says). Class 12 and 13 are incoherent clusters mixing words with different part of speech. Class 14 and 15 show the discrete classes for "通过". It is a hard task to evaluate the clusters carrying both syntactic and semantic similarity without a standard benchmark set. The measurement of the word relatedness is not well defined. For class 11, we only achieve 64% precision in terms of name entity. The precision increases to 74% if we consider the pronouns.

We are currently exploring how to merge the classes more efficiently and reduce the classes as well as the possibility of learning phrase structure through distributional information.

8.4 Conclusion

This chapter presents a distributional approach for word segmentation and classification. We use contextual entropy and mutual information for word segmentation. Mutual information captures the dependency inside the word. Contextual

entropy captures the dependency with the contexts in which the word occurs. Then we classify words according to their occurrence statistics rather than the knowledge of stored in a dictionary. We obtained larger number of classes mainly because of both syntactic and semantic constraints from the context. In addition in our approach words are allowed to belong to multiple independent classes. Each class carries not only syntactic information but also semantic information.

8.5 Reflections

The nature of unsupervised segmentation, and varying conventions about how to divide text up into words, means that evaluation is major issue, and there are several inconsistent national and proprietary standards or models of segmentation. These results should thus be regarded as preliminary as quantitative evaluation will need to be undertaken against at least 3 or 4 of the major conventions, and appropriate tagged corpus would need to be developed.

The unsupervised clustering facet of this approach brings us into realm where word classes are even fuzzier and even cluster size and no. of clusters are arbitrary parameter. It is not a trivial task to evaluate the induced word clusters in the absence of a benchmark set. Subjective assessment seems a plausible approach to assessing the quality of these clusters. It is practically unfeasible to implement it given the size and the no. of the clusters. There is a poor agreement on word relatedness by human subjects. The alternative way of measuring clusters is to contrast them with existing lexical resources but it is subject to the vocabulary size of the lexical resources and not all word relationships could be contained in the lexical resources. For English Yang found there was only around 30% overlap between each pair of clusters derived from Wordnet, Roget and unsupervised clusters induced for corpus with each cluster of 1000 words (Yang & Powers 2008). This is only the preliminary work for Chinese word classification. In future we will try to evaluate our results using Chinese Wordnet.

Part III

CHINESE INFORMATION RETRIEVAL

9. USING SUFFIX ARRAYS TO COMPUTE STATISTICAL INFORMATION

In this chapter we will introduce suffix trees and suffix arrays. Suffix trees and suffix arrays have become important data structures for problems in pattern matching, text indexing, compression, information retrieval and other applications. We will describe how to use suffix arrays to compute the term frequency and document frequency from corpus.

9.1 Suffix Trees

We use $S = T[1 \dots n]$ as our input text. T is a string of length n, over an alphabet Σ. We assume that for a given string the symbol alphabet is fixed and treat the alphabet size as being constant. Let T = A B C, for some strings A, B, and C. The string B is called a substring of T, A is called a prefix of T, while C is called a suffix of T. We will use $s_i = T[i]$ to denote the i-th symbol in T. We use $T_i = T[i \dots n] = t_i t_{i+1} \dots t_n$ to denote the i-th suffix of T. Similarly, we use $T^i = T[1 \dots i] = t_1 t_2 ... t_i$ to denote the i-th prefix of T. We us $T_i^j = T[i \dots j] = t_i t_{i+1} \dots t_j$ to denote the substring from t_i to t_j of T.

The suffix tree is a data structure used to represent the set of all suffixes of a string. Given a string T of length n, its suffix tree τ_T is a rooted tree with n leaves, where the i-th leaf node corresponds to the i-th suffix T_i of T. Except for the root node and the leaf nodes, every node must have at least two descendant child nodes. Each edge in the suffix tree $T\tau_T$ represents a substring of T, and no two edges out of a node start with the same character. For a given edge, the edge label is simply the substring in T corresponding to the edge. We use l_i to

denote the i-th leaf node. Then, l_i corresponds to T_i, the i-th suffix of T. Table 9.1 shows the suffix tree of "to_be_or_not_to_be$"

Before discussing the construction of suffix trees, we summarize their basic properties. Given the string $T = T[1 \ldots n]\$$ of length n, but with the end of string symbol appended to give a sequence with a total length $n + 1$, the suffix tree of the resulting string $T\$$ will have the following properties:

1. Exactly $n + 1$ leaf nodes;
2. At most n internal (or branching) nodes (the root node is considered an internal node);
3. Every distinct substring of T is encoded exactly once in the suffix tree. Each distinct substring is spelled out exactly once by traveling from the root node to some node in the suffix tree;
4. No two edges out of a given node in the suffix tree start with the same symbol;
5. Every internal node has at least two outgoing edges.

Properties (1), (2), (4), and (5) imply that a suffix tree will have at most $2n + 1$ total nodes, and at most $2n$ edges.

Construction of the suffix tree for a string is not difficult but how to construct a suffix tree efficiently is the critical problem of using suffix tree. A simple algorithm that accomplishes this task for any given string is given in the following algorithm.

$Simple - Suffix - Tree - Algorithm(T)$

Create the root node, with empty string

for i ←1 to n do

Traverse current tree from the root

Match symbols in the edge label one-by-one with symbols in the current suffix, T_i

if a mismatch occurs then

Split the edge at the position of mismatch to create a new node, if need

Insert suffix T_i into the suffix tree at the position of mismatch

end if

end for

Since Weiner (Weiner 1973) proposed this data structure on 1973, a lot of work has been done in this area. There are several well known algorithms which have linear complexity with respect to the length of the given string on both time and space (Weiner 1973, McCreight 1976, Ukkonen 1995, Gusfield 1997).

9.2 Suffix Arrays

An important data structure, closely related to the suffix tree, is the suffix array(Manber & Myers 1990). The suffix array simply provides a lexicographically ordered list of all the suffixes of a string.

If the suffixes are sorted, some of them may share common prefixes shown in Table 9.1. These prefixes share a common path from the root as in a PATRICIA tree. Thus the sorted suffixes can be represented by a Trie-like or PATRICIA-like data structure called suffix tree. A given suffix tree can be used to search for a substring, $substr[1..m]$ in $O(m)$ time. There are $n(n+1)/2$ substrings in $str[1..n]$. A substring must be a prefix of a suffix of str, if it occurs in str.

Suffix arrays provide the same function as suffix trees and occupy much less space. A suffix array is simply an array containing all the pointers to the suffixes of a text sorted in lexicographical (alphabetical) order. Each suffix is a string starting at a certain position in the text and ending at the end of the text. Searching a text can be performed by binary search using the suffix array.

The algorithm, $suffix\_array$, presented below takes a string and its length N as input, and outputs the suffix array, s.

$suffix\_array \leftarrow function(string, N)\{$

Initialize s to be a vector of integers from 0 to $N-1$.

Let each integer denote a suffix starting at $s[i]$.

Position	0	1	2	3	4	5	6	7	8
Characters	t	o	_	b	e	_	o	r	_
Position	9	10	11	12	13	14	15	16	17
Characters	n	o	t	_	t	o	_	b	e

Suffix Array	Indexes	S_i	Suffixes
s[0]	0	S_0	to_be_or_not_to_be
s[1]	1	S_1	o_be_or_not_to_be
s[2]	2	S_2	_be_or_not_to_be
s[3]	3	S_3	be_or_not_to_be
s[4]	4	S_4	e_or_not_to_be
s[5]	5	S_5	_or_not_to_be
s[6]	6	S_6	or_not_to_be
s[7]	7	S_7	r_not_to_be
s[8]	8	S_8	_not_to_be
s[9]	9	S_9	not_to_be
s[10]	10	S_{10}	ot_to_be
s[11]	11	S_{11}	t_to_be
s[12]	12	S_{12}	_to_be
s[13]	13	S_{13}	to_be
s[14]	14	S_{14}	o_be
s[15]	15	S_{15}	_be
s[16]	16	S_{16}	be
s[17]	17	S_{17}	e

Tab. 9.1: Suffixes and suffix arrays before sorting

Array	i	S_i	Suffixes	Lcp vector	Length
s[0]	15	S_{15}	_be	lcp[0]	0
s[1]	2	S_2	_be_or_not_to_be	lcp[1]	3
s[2]	8	S_8	_not_to_be	lcp[2]	1
s[3]	5	S_5	_or_not_to_be	lcp[3]	1
s[4]	12	S_{12}	_to_be	lcp[4]	1
s[5]	16	S_{16}	be	lcp[5]	0
s[6]	3	S_3	be_or_not_to_be	lcp[6]	2
s[7]	17	S_{17}	e	lcp[7]	0
s[8]	4	S_4	e_or_not_to_be	lcp[8]	1
s[9]	9	S_9	not_to_be	lcp[9]	0
s[10]	14	S_{14}	o_be	lcp[10]	0
s[11]	1	S_1	o_be_or_not_to_be	lcp[11]	4
s[12]	6	S_6	or_not_to_be	lcp[12]	1
s[13]	10	S_{10}	ot_to_be	lcp[13]	1
s[14]	7	S_7	r_not_to_be	lcp[14]	0
s[15]	11	S_{11}	t_to_be	lcp[15]	0
s[16]	13	S_{13}	to_be	lcp[16]	1
s[17]	0	S_0	to_be_or_not_to_be	lcp[17]	5
				lcp[18]	0

Tab. 9.2: Suffixes and suffix arrays after sorting

Sort s so that the suffixes are in alphabetical order.

return s; }

In order to compute the frequency, an auxiliary array is defined to store LCPs (longest common prefixes). The lcp array is a vector of $N+1$ integers. Each element, $lcp[i]$, denotes the length of the common prefix between the suffix $s[i-1]$ and the suffix $s[i]$. As mentioned above there are $N(N+1)/2$ substrings for a document with the length N. Instead of computing the statistics for all substrings directly the set of all substrings is partitioned by the classes with the same statistics (term frequency tf and document frequency df)(Yamamoto & Church 2001). The set of substrings in a class can be constructed from the lcp vector:

$$\begin{aligned} class(<i,j>) &= \{s[i]_m | LBL(<i,j>) < m \leq \\ &\quad SIL(<i,j>)\} \qquad (9.1) \\ LBL(<i,j>) &= max(lcp[i], lcp[j+1]) \\ SIL(<i,j>) &= min(lcp[i+1], lcp[i+2], \ldots, lcp[j]) \\ tf_{class(<i,j>)} &= j-i+1 \qquad (9.2) \end{aligned}$$

where LBL is longest bounding lcp, SIL is shortest interior lcp and $s[i]_m$ denotes the first m characters of suffix $s[i]$. A $class < i,j >$ exists between interval $< i,j >$ if $LBL < SIL$. Then this interval $< i,j >$ is lcp-delimited. It means all the suffixes in lcp-delimited interval $< i,j >$ share the same longest common prefix and no other suffixes outside the interval shares it. Thus it is not possible for two lcp-delimited intervals to overlap but it is possible to be nested. Table 9.2 shows that lcp-delimited interval $< 0,1 >$ with lcp "_be" is nested in interval $< 0,4 >$ with lcp "_". The term frequency for $class(< i,j >)$ is equal to $j-i+1$. $class(< i,i >)$ has term frequency 1 called trivial class. We are more interested in the nontrivial class with term frequency greater than 1. The number of substrings in a nontrivial class is

$$|class(<i,j>)| = SIL(<i,j>) - LBL(<i,j>). \qquad (9.3)$$

Interval	Class {Substrings}	SIL	LBL	tf
$< 0, 1 >$	_be {_b,_be }	3	1	2
$< 0, 4 >$	_ {_}	1	0	5
$< 5, 6 >$	be {b,be}	2	0	2
$< 7, 8 >$	e {e}	1	0	2
$< 10, 11 >$	o_be {o_,o_b,o_be}	4	1	2
$< 10, 13 >$	o {o}	1	0	4
$< 16, 17 >$	to_be {to,to_,to_b,to_be}	5	1	2
$< 15, 17 >$	t {t}	1	0	3

Tab. 9.3: Nontrivial classes for string "to_be_or_not_to_be"

The substrings in the nontrivial class $class(< i, j >)$ are the first $LBL(< i, j >) + 1, \ldots, SIL(< i, j >)$ characters of suffix $s[i]$, total $SIL(< i, j >) - LBL(< i, j >)$ prefixes of suffix $s[i]$. All substrings in the same class have the same term frequency and document frequency if all suffixes were terminated with the first end of document symbol in multi-document corpus. Table 9.3 shows nontrivial classes for string "to_be_or_not_to_be". For $class < 16, 17 >$ with SIL 5 and LBL 1 there are 4 substrings {to, to_, to_b, to_be} in the class with the same term frequency 2. In Chinese the longest substring in the class is most likely to be a word or a phase. We can use the longest substring to represent the class instead of using all the substring to reduce the size of all substrings.

As two lcp-delimited intervals are not possible to overlap and possible to be nested, there are at most $N - 1$ non-trivial classes with term frequency greater than 1. For trivial classes there are at most N classes with term frequency equal to 1. This significantly reduces the computation of various statistics over substrings $(N(N + 1)/2))$ to a computation over classes $(2N - 1)$.

All the substrings in the same class have the same attributes. The longest substring in the class can be used to represent the class instead of using all the substrings to reduce the size of the substrings. As suffix array is built from one direction, Some substrings of the longest substring in the class with the same attributes are actually not including in the same class. In Table 9.3 there are

4 substrings {to, to_, to_b, to_be} in the $class < 16, 17 >$ with the same term frequency 2. If we construct another suffix from the other direction (from the end to the start), we should get 5 substrings {e, be, _be, o_be, to_be} in the same class with longest substring "to_be". In this case, we can merge classes with same attributes. We can remove these substrings without building bi-directional suffix array. After one suffix array is build and all statistics of substrings is computed, we can compare the attributes of every substring with the substrings of itself. If they have same attributes, we can merge them further.

Once we identify all the classes of a corpus, we can use a straightforward method to compute the term frequency and the document frequency in a corpus. The term frequency is equal to the size of the internal. For the document frequency, we can enumerate the suffixes within the interval and compute their document *ids* and return the number of distinct documents after removing duplicates.

9.3 Computing Term Frequency and Document Frequency

Use a straightforward method to compute the term frequency and and the document frequency for all classes is certainly too slow. Yamamoto and Church (Yamamoto & Church 2001) used a stack to take advantage of the nesting property of lcp-delimited intervals to compute the term frequency and document frequency recursively. The following procedure is based on their work. The original procedure only calculates the total term frequency in the corpus. We modify the procedure to compute the term frequency in each document for information retrieval.

i, the left edge of an interval,
k, the representative ($SIL = lcp[k]$),
df, partial results for dr, counting documents seen thus far, minus duplicates.

$print\_LDIs\_with\_df \leftarrow$ function(N){

    $stack\_i \leftarrow$ an integer array for the stack of the left edges, i.

$stack\_k \leftarrow$ an integer array for the stack of the representatives, k.

$stack\_df \leftarrow$ an integer array for the stack of the df counter.

$doclink[0..D]$: an integer array for the document link initialized with –1.

D = the number of documents.

$doctf[0..D]$: an integer array for the term frequency in each document.

D = the number of documents.

$stack\_i[0] \leftarrow 0$.

$stack\_k[0] \leftarrow 0$.

$stack\_df[0] \leftarrow 1$.

$sp \leftarrow 1$ (a stack pointer).

(1) For $j \leftarrow 0, 1, 2, \ldots, N-1$ do

(2) (Output a trivial lcp-delimited interval $< j, j >$ with tf=1 and df=1.)

(3) $doc \leftarrow$ get_docnum($s[j]$)

(4) if $doclink[doc] \neq -1$, do

(5) let x be the largest x such that $doclink[doc] \geq stack\_i[x]$.

(6) $stack\_df[x] \leftarrow stack\_df[x] - 1$.

(7) $doclink[doc] \leftarrow$ j.

(8) $df \leftarrow 1$.

(9) While $lcp[j+1] < lcp[stack\_k[sp-1]]$ do

(10) $df \leftarrow stack\_tf[sp-1] + df$.

(11) if it is lcp-delimited.

(12) (Output a nontrivial interval $< i, j >$)

(13) Output $tf = j - i + 1$ and df

(14) $doctf[0..D] \leftarrow 0$

(15) For $t \leftarrow i, \ldots, j$ do

(16) $doc \leftarrow$ get_docnum(t)

(17) $doctf[doc] \leftarrow doctf[doc] + 1$

(18) if $doclink[doc] == t$

(19) Output doc and $doctf[doc]$

(20) $sp \leftarrow sp - 1$.

(21) $stack\_i[sp] \leftarrow stack\_k[sp-1]$.

(22) $stack\_k[sp] \leftarrow j + 1$.

For each value of j between 0 and $N - 1$, a trivial interval is reported at $< j, j >$ in line 2. In addition, there could be up to $N - 1$ nontrivial intervals, where k is the representative and $lcp[k]$ is the SIL. Lcp-delimited intervals are uniquely determined by a representative k such that $i < k \leq j$ where $SIL(< i, j >) = lcp[k]$. The stack keeps track of i, k and df on lines 20,21,22. For each j, if $LBL(< i, j >) \geq SIL(< i, j >)$, the values of i, k and df are pushed in the stack otherwise the while-loop pops the stack if $lcp[j] < lcp[k]$. The procedure reports intervals at $< i, j >$ only when $LBL(< i, j >) < SIL(< i, j >)$.

Lines 5 and 6 check duplicate documents. The duplication processing makes use of *doclink*, which keeps track of which suffixes have been seen in which document, *doclink* is initialized with -1 indicating that no suffixes have been seen yet. As suffixes are processed, *doclink* is updated (on line 7) so that $doclink[d]$ contains the most recently processed suffix in document d.

Stack_df keeps track of document frequencies in the corpus as suffixes are processed. The element $stack\_df[x]$ contains the document frequency for suffixes seen thus far starting at $i = stack_i[x]$ where x is a stack offset. When a new suffix is processed, line 5 checks for double counting by searching for intervals on the stack that have suffixes in the same document as the current suffix. If there is any double counting, *stack_df* is decremented appropriately on line 6.

Intervals are processed in depth-first order, so that more deeply nested intervals are processed before less deeply nested intervals. In this way, double counting is only an issue for intervals higher on the stack. The most deeply nested intervals are trivial intervals. They are processed first. They have a *df* of 1 (line 8). For the remaining nontrivial intervals, $stake_df$ contains the partial results for intervals in process. As the stack is popped, the *df* values are aggregated up to compute the *df* value for the outer intervals. The aggregation occurs on line 10 and the popping of the stack occurs on line 12.

Lines 14-19 calculate the term frequency of the suffix in each document. $doctf$

keeps track of term frequency in each document. At beginning it is initialized with 0. For a nontrivial interval $< i, j >$, $doctf[doc]$ increments if the suffix is seen in the document *doc*. If the suffix is recorded in the *doclink*, then this suffix will be the last time to occur in this document. Document *id* and term frequency are obtained.

9.4 Conclusion

Suffix array based approach described above can reduce the computation term frequency and document frequency over substrings $(N(N+1)/2))$ to a computation over classes $(2N-1)$. We have applied the suffix array to compute the term frequency for n-gram language modelling and Chinese segmentation in the previous chapters. The output n-grams with term frequency and document frequency here are used in next chapter for information retrieval.

10. N-GRAM BASED APPROACH FOR CHINESE INFORMATION RETRIEVAL

With the widespread of the Internet, great research interests are being shown in Chinese language information retrieval in recent years. The absence of word boundaries in Chinese language makes Chinese information retrieval(IR) different to European IR. In order to apply traditional IR approaches to Chinese language, sentences have to be segmented into words first. Word segmentation is playing a key role in Chinese IR. As word segmentation is not straightforward and the results are sometime ambiguous, n-grams are used as an alternative. In this chapter[1] we investigate performance of Chinese IR using words and n-grams as indexes, the effects of word extraction and stop words on Chinese IR.

10.1 Introduction

The number of electronic documents other than European languages available in the Internet is growing enormously. Traditional information retrieval systems for European languages such as English use words as indexing units can not apply directly to Asian languages such as Chinese because English text is written with delimiters and words can be easily recognized. Chinese text is written as continuous strings of ideographs (or characters). Thus there is major difference between Chinese IR and IR in European language. A pre-processing called segmentation has to be done to determine the boundaries of words before traditional IR approaches (Salton & McGill 1986, Manning, Raghavan & Schutze 2008) based on

[1] This chapter is partly based on the paper (Huang & Powers 2008)

Huang, J. & Powers, D. (2008), Suffix-tree-based approach for Chinese information retrieval, in 'International Conference on Intelligent Systems Design and Applications (ISDA)'.

words can be adapted to Chinese language. Because text segmentation is not straightforward and the process itself can have ambiguous outcomes, n-grams are used as an alternative indexing units instead of words. Several studies (Nie et al. 2000, Nie & Ren 1999) have been carried out to compare these two kinds of indexing approaches. It turns out that using either words or n-grams leads to comparable performances. Further studies (Foo & Li 2004, Peng et al. 2002) have revealed that the accuracy of word segmentation have an impact on IR performance but higher word segmentation accuracy does not necessarily result in better retrieval performance. In this chapter we discuss the reasons that cause different retrieval results using words or n-grams as indexes. Furthermore we investigate the effects of stop words and longer extracting words on retrieval results.

In this chapter, first we introduce the concept of Chinese information retrieval and review the previous work. Then we describe the information retrieval models and evaluation. Then we present the experiments we conducted on the TREC-5 and TREC-6 data sets. Finally, a discussion and conclusions are given in Section 10.6 and 10.7.

10.2 Chinese Information Retrieval

For Chinese IR systems, choosing what kind of indexing units is more problematic than those that dealing with European languages only.

10.2.1 Single-character-based (Uni-gram) Indexing

Typical character-based indexing uses single Chinese characters as index terms. In Chinese language, the Chinese character is the element unit of their writing system. The definition of Chinese character does not cause any controversy because visually a character is an isolated symbol. Although the majority of Chinese words are compound words consisting of free or bound morphemes, the meanings of most compound words can be derived from the meanings of their constituents. Each character has its own semantic and syntactic properties. It

can ensure no information loss and is quite easy to implement. Any document contains the same characters as the query will be retrieved. It causes high recall but low precision as some single characters are polysemantic and homonymic.

10.2.2 Multi-character-based (N-grams) Indexing

This indexing method uses chunks of n consecutive characters as the index term. Neither dictionary nor other linguistic knowledge is required in the processing. This leads to the advantages of reduced costs and minimal overheads in the indexing and querying process. Bi-grams have been often used as indexing terms form Chinese IR as most Chinese words are composed of two characters. In addition to the ease of word identification, bi-grams bear more semantic information than single characters. Bi-gram indexing is exhaustive and avoids the difficult problem of word segmentation. Bi-grams can consider unknown words and abbreviations in a better way than words do. The drawbacks of using bi-grams as indexes are meaningless character chunks are abundant among bi-grams, leading to noisy matching between queries and documents.

10.2.3 Word-based Indexing

Using single characters and n-grams as index terms makes it difficult to incorporate linguistic knowledge into the retrieval processing particularly for multilingual and cross language information retrieval because both of them are not ideal conceptual units. word is a better token representation in terms of end-user satisfaction.

As mentioned above Chinese words are not readily recognizable because Chinese orthography fails to represent word boundaries. Therefore, it is necessary and important that word segmentation has to be carried out to break the original Chinese text into a series of words. Segmentation of Chinese text into words will add extra burden for Chinese IR. It requires linguistic knowledge and great coverage of dictionaries. The segmentation problems of segmentation ambiguity

and unknown words will affect the performance of IR as new words and new occurrences of proper nouns such as names of person names, organizations, places cannot be covered by dictionaries. Furthermore there is poor agreement on word definition and segmentation as described in chapter 6. Many characters form one-character words by themselves, but these characters can also form multi-character words when used with other characters. As each character is meaningful, using word as index will lose the information carried by individual character.

10.2.4 Previous Works

All previous researches are related to how to segment sentence and produce indexes. Numerous different segmentation approaches have been proposed for Chinese IR. These approaches can be basically divided into character-based and word-based approaches. Under these two basic groups, there are many alternatives, such as single-character or multiple-character segmentation, use of dictionary or statistics, or introducing linguistic knowledge for segmentation (Nie & Ren 1999, Nie et al. 2000, Peng et al. 2004, Chen, He, Xu, Gey & Meggs 1997, Foo & Li 2004).

Character-based (N-grams) Approaches

Character-based approaches can be defined as purely mechanical processes that extract certain number of characters from texts. According to the number of characters extracted, character-based approaches can be further divided into single character-based approach and multi-character-based approaches.

Single character-based approach segments Chinese texts into single characters and is the simplest method to segment Chinese text. The majority of today's CIR systems generally do not employ single character-based approach as the main segmentation approach although some research groups have obtained encouraging results by only using this approach (Buckley, Singhal & Mitra 1996, Huang & Roberton 1997, Nie, Chevallet & Bruandet 1997, Smeaton & Wilkinson 1997)

Multi-character-based (or N-grams) approaches segment texts into strings containing two (bigram), three or more characters. Since most of Chinese words are two characters a popular approach is overlapping bigram approach that segments a string ABCD into into AB BC CD. Compared with the single character approach, the multi-character approaches consistently yield superior CIR results (Kwok 1999, Nie & Ren 1999, Nie et al. 2000). In applying these character-based approaches to IR, the most obviously advantage is its simplicity and ease of application without a pre-defined lexicon. This in turn leads to other advantages of reduced costs and minimal overheads in the indexing and querying process. As such, multi-character-based approaches, especially the bigram approaches, have been found to be practical options that are implemented in many CIR systems. However, the disadvantages include the requirement of a huge index file and the fact that it is difficult to incorporate linguistic information of any kind.

Word-based Approaches

Word-based approaches attempt to extract complete words from sentences. They can be further categorized as statistics-based, dictionary-based and hybrid approaches.

Statistics-based approaches use the statistical information of Chinese characters in the corpus to mark word boundaries. The lexical statistics include the occurrence frequency of each character in the corpus, and the co-occurrence frequency of each pair of characters in the corpus. This approaches are significantly dependent on a training corpus so that the index terms produced are more sensitive and useful for particular domains that are similar to the training corpus (Nie, Brisebois & Ren 1997, Peng et al. 2002, Chen et al. 1997).

Dictionary-based approach is commonly used in most current systems utilizing the word based approach for text segmentation. It predefines a lexicon containing a large number of Chinese words and then uses heuristic methods such as maximum matching to segment Chinese sentences. According to matching heuristic methods, the dictionary-based approach can be further divided into

longest match (by scanning the text sequentially to match the dictionary and choosing the longest strings as index term), shortest match (by scanning the text sequentially to match the dictionary and choosing the first matched word as index term) and overlap match (He, Xu, Chen, Meggs & Gey 1996, Nie et al. 2000). In overlap match approach, words generated from texts can overlap each other across the matching boundary or characters and shorter words in the longest match words can overlap each other.

The word based approach has the advantage of requiring a smaller inverted index file and allowing additional linguistic information to be incorporated in the retrieval system. The most prominent disadvantage is that it requires a large predefined lexicon, which normally must be constructed by hand with a significant amount of time and labor. Moreover it is virtually impossible to list all Chinese words in a dictionary. An additional shortcoming of the traditional maximum matching method used in the segmentation is that a character sequence is always segmented the same way regardless of context.

All these segmentation approaches have advantages and disadvantages. A hybrid approach combining different approaches and taking into account the strength of various techniques has been reported by many researchers (Nie, Brisebois & Ren 1997, Kwok 1999, Nie et al. 2000, Smeaton & Wilkinson 1997). They usually combine statistic-based and dictionary-based approaches in an attempt to merge the benefits of general and domain-specific knowledge. Although the hybrid approaches take advantage of different approaches to obtain more accurate segments, these are achieved against the expense of more complex processing time, disk space and cost requirements.

Comparing Different Segmentation Approaches in IR Experiments

Both the word based and character-based approaches have been successfully applied to Chinese information retrieval. Some researchers (Buckleym, Walz, Mitra, & Cardie 1997, Huang et al. 1997) obtained better results using character-based approaches while others (Nie, Chevallet & Bruandet 1997, Wilkinson 1998) ob-

Research Team	Better Run
City University	Character better (5%)
Claritech Corporation	Character only
Cornell University	Character only
Information Technology Institute	Character only
Institute of Systems Science	Character better (18%)
Royal Melbourne Institute of Technology	Word better (1%)
Queens College, CUNY	Word better (4%)
Swiss Federal Institute of Technology	Character only
University of California, Berkeley	Word only
University of Massachusetts, Amherst	Character better (2%)
University of Montreal	Word better (1%)
University of Waterloo	User selected, Manual run

Tab. 10.1: Comparison of segmentation approaches in TREC 6

tained better results using word-based approaches.

A lot of studies have been done to compare the effectiveness of these approaches particularly with the induction of "standard" Chinese IR test in TREC 5 (Voorhees & Harman 1996) and 6 (Wilkinson 1998). This allowed participating researchers to implement various segmentation approaches in their experiments and evaluate the performance. Most of participants of TREC-6 Chinese track devoted their studies to comparing different approaches. Out of the six groups that have compared the character-based approach with word-based approach, half have found n-gram indexing is better while the other half report that word indexing is better shown in table 10.1. It reported that bi-gram approaches are comparable with any other individual technique and have the advantage of no requiring the difficult task of segmentation in order to employ word-based approaches in TREC 6 (Wilkinson 1998).

In TREC 9 Fudan University (Wu, Huang, Guo, Liu & Zhang 2000) found that the word segmented index yielded better results than n-gram index. In contrast, IBM (Franz, McCarley & Zhu 2000) found that the character-based results were

better than word-based results across both types of Chinese-English translation and monolingual retrieval. Johns Hopkins University (McNamee, Mayfield & Piatko 2000) assessed the use of 3-grams in comparison with bigrams. They found that 3-grams performed appreciably worse than those using bi-grams. They observed that this trend seemed to hold both in monolingual retrieval with natural language queries and in bilingual retrieval using word-based translations. The experiments at Microsoft Research China (Gao, Xun, Zhou, Huang, Nie, Zhang & Su 2000) reported that the result using both bigram and unigram is comparable to the best performance using their word-based approach. They further used an index comprising bi-grams and words and found slight improvements of 2.6% over the uncombined case.

Therefore there remains a question of whether word segmentation is necessary at all for Chinese text retrieval. However, some researchers(Nie & Ren 1999) have argued that there exist some inherent difficulties in the n-gram based approach. For example, a modern Chinese information retrieval system should be able to take into account more than just character information, but should also be able to exploit sophisticated techniques such as latent semantic indexing (Hofmann 1997).

The relationship between Chinese word segmentation accuracy and information retrieval performance has been investigated by other researchers (Foo & Li 2004, Palmer & Burger 1997, Peng et al. 2002). Foo and Li (Foo & Li 2004) have conducted a series of experiments which suggest that the word segmentation approach does indeed have effect on retrieval performance. Specifically, they observe that recognizing words of length two or more can produce better retrieval performance, and the existence of ambiguous words resulting from the word segmentation process can decrease retrieval performance. Similarly, Palmer and Burger (Palmer & Burger 1997) observe that accurate segmentation tends to improve retrieval performance. Furthermore Peng and Schuurmans (Peng & Schuurmans 2001) used the self-supervised expectation maximization(EM) based segmentation method to control the segmentation precision for Chinese information retrieval. They found that the relationship between word segmentation

accuracy and retrieval performance is not obvious. Retrieval performance increases as word segmentation accuracy increases, but it begins to plateau after some point and eventually decrease when segmentation accuracy is too high.

10.3 Retrieval Models

A retrieval model specifies how the content of a document and user's information need is represented in an IR system, how the documents and the information needs are matched so the relevant items can be retrieved. We used a vector space model, which view documents and queries as vectors in an n-dimension vector space and use distance as a measure of similarity. There are some other probabilistic models used in information retrieval such as the 2-Poisson model (Robertson & Sparck Jones 1976) in OKAPI systems (Huang & Robertson 2000)(Robertson & Walker 1994), the logistic-regression (LR) model (Cooper, Chen & Gey 1994) and Pircs system (Kwok 1996). Luk and Kwok (Luk & Kwok 2002) compared different retrieval models and their effects on Chinese information retrieval.

10.3.1 Vector Space Model

In the Vector Space Model (VSM) (Salton & McGill 1986), both queries and documents are transformed into vector in an n-dimensional space (n denotes the number of indexes). Dimensions are actually represented by the weighted index terms.

Suppose there is a document D_i in collection D and a query Q_j, then the vector D_i and Q_j can be represented respectively as follows:

$$D_i = (d_{i1} \cdots d_{im}) \tag{10.1}$$

$$Q_j = (q_{j1} \cdots q_{jm}) \tag{10.2}$$

where d_{ik} is the weight of term t_k in the document D_i, q_{ik} is the weight of term t_k in the query Q_i,and m is the size of the vector space (the number of different

terms, words or ngrams).

10.3.2 Term Weighing

The weight of a term in a document is calculated according to its occurrence frequency in the document (term frequency) and its distribution in the entire collection. Term weighting in VSM makes use of the three factors: the term frequency(tf), the inverse document frequency(idf) and normalization factor. Some weighting schemes (Sparck Jones 1997) even include the document length to provide additional evidence. The most popular used equation for calculating weights is the $tf * idf$ function (Salton & Buckley 1988):

$$d_{ij} = tf_{ij} * idf_{ij} \tag{10.3}$$

where

$$idf_{ij} = \log(N/df_j) \tag{10.4}$$

where i denotes the ith document, j denotes the jth term in the document D_i, N is the total number of documents in the collection and df_j is the number of documents that contain the term t_j.

This weighting scheme assigns to term a weight in document that is highest when occurs many times within a small number of documents. Thus it lends high discriminating power to those documents with such terms. Such terms are most likely to be content words. Whereas it assigns the lowest weight to the term which occurs in virtually all documents. Such terms tends to be function words. In other words content words give more weight than function words.

For assigning a weight for each term in each document, a number of alternatives to tf and tf-idf have been considered in SMART system(Buckley 1985) shown in Table 10.2, where $tf_{t,d}$ is the term frequency of term t in document d, N is the total number of documents in the collection. df_t is the number of documents that contain the term t. .

We adopted the following formula from the Smart system in our experiments

Term Frequecy	Document Frequency	Normalization
$n(natural) : tf_{t,d}$	$n(no) : 1$	$n(none) : 1$
$l(logarithm) : 1 + \log(tf_{t,d})$	$t(idf) : \log \frac{N}{df_t}$	$c(cosine) : \frac{1}{\sqrt{w_1^2+w_2^2+...+w_M^2}}$
$a(augmented) : 0.5 + \frac{0.5 \times tf_{t,d}}{max(tf_{t,d})}$	$p(prob\, idf) :$	$b(bytesize) :$
$b(boolean) : 1\,(tf_{t,d} > 0)\, or\, 0$	$\max(0, log \frac{N-df_t}{df_t})$	$1/CharLength^{\alpha}, \alpha < 1$
$L(logaverage) : \frac{1+log(tf_{t,d})}{1+log(ave_{t \epsilon d}(tf_{t,d}))}$		

Tab. 10.2: Term Weighting in the Smart system

.

$$d_{ik} = \frac{(\log(tf_{ik}) + 1.0) * \log(\frac{N}{df_k})}{\sqrt{\sum_j ((\log(tf_{jk}) + 1.0) * \log(\frac{N}{df_k}))^2}} \tag{10.5}$$

where tf_{ik} is the occurrence frequency of the term t_k in the document D_i, N is the total number of documents in the collection. df_k is the number of documents that contain the term t_k.

10.3.3 Query and Document Similarity

The main power of VSM is that it enables the system to measure the proximity between any two vectors, that is, it can decide the degree of similarity between the query and the document. The most commonly used similarity function is the cosine correlation. Similarity between D_i and Q_j is calculated as the inner product of their vectors as following:

$$Sim(D_i, Q_j) = \sum_k (d_{ik} * q_{jk}) \tag{10.6}$$

10.3.4 Evaluation

The standard approach to information retrieval system evaluation revolves around the notion of relevant and nonrelevant documents. A retrieved document is relevant if it addresses the query, not because it contains all the words in the query. It has to be judged by the users.

IR systems are usually evaluated in terms of their effectiveness. Retrieval effectiveness refers to the ability of a system to pick up most relevant documents from the collection to reject those that bear little relevance to the user's information need. The two most frequent and basic measures for information retrieval effectiveness are precision and recall. Precision (P) denotes how many documents in all the retrieved documents are relevant. Recall (R) tells how many relevant documents have been retrieved from all the relevant documents in the collection. They can be formulated as follows:

$$Precision = \frac{\text{the number of relevant items retrieved}}{\text{total number of retrieved items}} = P(relevant|retrieved) \quad (10.7)$$

$$Recall = \frac{\text{the number of relevant items retrieved}}{\text{total number of relevant items}} = P(retrieved|relevant) \quad (10.8)$$

The advantage of having the two numbers for precision and recall is that one is more important than the other in different circumstances but there is tradeoff between precision and recall. Recall is a non-decreasing function of the number of documents retrieved. On the other hand, precision usually decreases as the number of documents retrieved is increased. In general we want to get some amount of recall with a reasonable precision.

Precision and recall are set-based measures computed on unordered sets of documents. We need to extend these measures to evaluate the ranked retrieval results that are now standard with modern IR tasks. In a ranked retrieval context, appropriate sets of retrieved documents are naturally given by the top k retrieved documents. For each such set, precision and recall values can be plotted to give a precision-recall curve. Precision-recall curves have a distinctive jagged shape: if the $(k+1)$th document retrieved is nonrelevant then recall is the same but precision has dropped. If it is relevant, then both precision and recall increase. It is often useful to remove these jiggles and the standard way to do this is with an interpolated precision: the interpolated precision p_{interp} at a certain recall level r is defined as the maximum precision found for any recall level $r' \geq r$:

$$p_{interp}(r) = \max_{r' \geq r} p(r') \tag{10.9}$$

The traditional way of doing this is the 11point interpolated average precision. For each query, the interpolated precision is measured at the 11 recall levels of $0.0, 0.1, 0.2, ..., 1.0$. For each recall level, the interpolated precisions at that recall level for all queries in the test collection are then averaged. These values are used for recall-precision graphs.

In recent years, average precision have become more common among the TREC community (Voorhees & Harman 2005). It provides a single-figure measure of discrimination and stability. For a query, average precision is the average of the precision value obtained for the set of top k documents existing after each relevant document is retrieved. Conceptually for each query, this is the area underneath the recall-precision graph. Then this value is averaged over all queries. If the set of relevant documents for a query $q_j \in Q$ is $d_1, \ldots d_{m_j}$ and R_{jk} is the set of ranked retrieval results from the top to document d_k, then

$$AP(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} Precision(R_{jk}) \tag{10.10}$$

When a relevant document is not retrieved, the precision value in the above equation is set to 0. The average precision value for a test collection is the arithmetic mean of average precision values for the individual queries.

The above measures factoring in precision at all recall levels may not enough for some users who only care about the top several retrieved pages after web searching. This leads to measuring precision at fixed low levels of retrieved results. This is referred to as "Precision at X", for example "Precision at 10". The precision after X documents (whether relevant or nonrelevant) have been retrieved. Values averaged over all queries for a test collection. If X docs were not retrieved for a query, then all missing docs are assumed to be non-relevant. It does not require any estimate of the size of the set of relevant documents but it is the least stable of the commonly used evaluation measures since the total

number of relevant documents for a query has a strong influence on precision at X.

An alternative is R-precision. It requires the total number (R) of relevant documents for a query to be known, from which we calculate the precision of the top R documents retrieved. Thus if a query has 50 relevant docs, then precision is measured after 50 docs, while if it has 1000 relevant docs, precision is measured after 1000 docs. This avoids some of the averaging problems of the "precision at X" values. If R is greater than the number of docs retrieved for a query, then the nonretrieved docs are all assumed to be nonrelevant.

10.4 Experimental Setup

10.4.1 TREC Data

The tests are conducted on TREC-5 (Voorhees & Harman 1996)and TREC-6 (Wilkinson 1998) Chinese corpus. The documents in the collection consist of approximately 170 megabytes of articles drawn from the People's Daily newspaper from 1991 to 1993 and the Xinhua newswire in 1994 and 1995. There are 164,789 documents in the collection. It consist of 139,801 articles from the People's Daily newspaper and 24,988 articles from the Xinhua newswire. with 0 bytes as the minimum file size, 294,056 bytes as the maximum size and 1014 bytes as the average file size. A set of 54 queries (28 for TREC 5 and 26 for TREC 6) with average length 338 bytes (Chinese characters only) has been set up and used to evaluate by by people in the NIST (National Institute of Standards and Technology) for Chinese information retrieval task. The document collection used in TREC-6 Chinese track was identical to the one used in TREC-5. All of the original articles were tagged using SGML. The Chinese characters in these articles were encoded using the GB (GuoBiao) coding scheme.

10.4.2 Measuring Retrieval Performance

The TREC relevance judgments for each topic came from the human assessors of the National Institute of Standards and Technology (NIST). Statistical evaluation was done by means of the TREC evaluation program from SMART system (Salton & McGill 1986). Several measures described in previous section are used to evaluate the retrieval result which is an ordered set of retrieved documents. The measures include 11point interpolated average precision, average precision (11 recall points without interpolation, 0.0, 0.1, 0.2,..., 1.0), R precision (precision after the number of documents retrieved is equal to the number of known relevant documents for a query) and Precision at X documents.

10.5 Experiments and Discussion

We will conduct the following tests to find out the factors affecting the performance of Chinese IR. Title, Description and Narrative are all used as the text of the queries. Details of the 54 queries are shown in Appendix. Queries are segmented in the way as the documents are segmented in the same test.

1. Using the longest matching with a dictionary
2. Using full word segmentation with the unsupervised method described in Chapter 7
3. Using different n-grams and its combinations
4. Adding words from word extraction
5. Removing the stop words

10.5.1 Using Dictionary-based Approach

We use the lexicon of common words in contemporary Chinese (National-Language-Committee 2008) released by State Language Committer of China in 2007 as the

dictionary for segmentation. It contains 56,008 common used words consisting of 3,181 single character words, 40,351 two-character words, 6,459 three-character words, 5,855 four-character words and 162 words more than five characters. We use the forward longest matching method to segment the TREC corpus. If characters are not in the dictionary, single character is produced except numbers. In other words, unknown words are segmented as different single characters. We obtained average precision of 0.3669 and R-precision of 0.3932.

10.5.2 Statistical Segmentation Approach

We use the contextual entropy and mutual information approach described in Chapter 7 to segment the TREC corpus. As TREC data is sourced from the People's Daily newspaper 1991-1993 and Xinhua newswire 1994-1995 and our segmentation model is trained in People's Daily 91-95, we can apply our segmentation approach without adjusting any parameters. We treated adjacent numbers, single word markers and punctuations as words. Our approach tries to discover word boundaries instead of words based on contextual entropy. Precision of segmentation is higher than small dictionary used in the previous section as unknown words are not segmented as single characters. We obtained average precision of 0.3957 and R-precision of 0.4084.

Figure 10.1 and Table 10.3 show the average precision at 11 recall points for dictionary-based and statistical segmentation approaches. Statistical segmentation approach is slightly (7%) better than dictionary based approach. Figure 10.2 and Table 10.4 show the R-precision and precision at different number of documents retrieved. As the average number of relevance documents for each query is around 130 documents, we see two lines converge after 100 documents in the figure. Average precision and R-precision are shown more reliable than precision at X documents returned. As we do not have large dictionary, here we may consider our statistical segmentation approach similar to using larger dictionary. We can see a better segmentation can increase IR effectiveness to some extent.

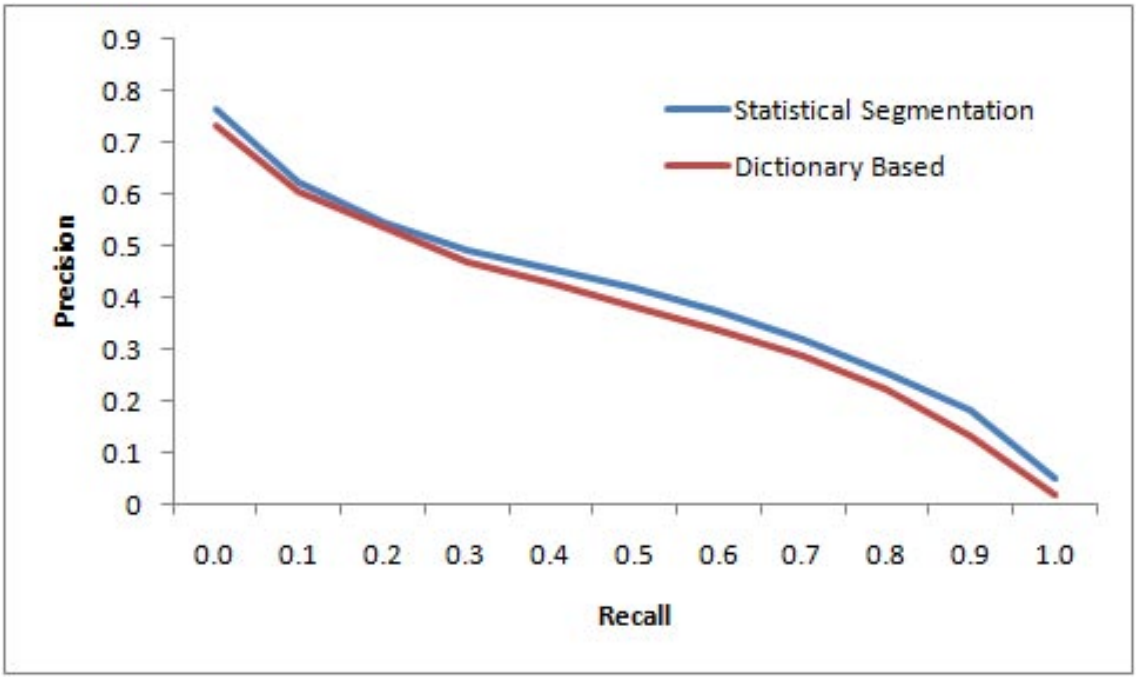

Fig. 10.1: Average Precision At Different Recall For Dictionary-based and Statistical Segmentation Approaches

	Statistical Segmentation Approach	Dictionary Based Approach
0.0	0.7639	0.7331
0.1	0.6231	0.6051
0.2	0.5483	0.5384
0.3	0.4919	0.4686
0.4	0.4535	0.4272
0.5	0.4196	0.3800
0.6	0.3751	0.3363
0.7	0.3194	0.2865
0.8	0.2564	0.2220
0.9	0.1804	0.1314
1.0	0.0505	0.0190
Average Precision	0.3957	0.3669

Tab. 10.3: Average Precision for Dictionary-based and Statistical Segmentation Approaches

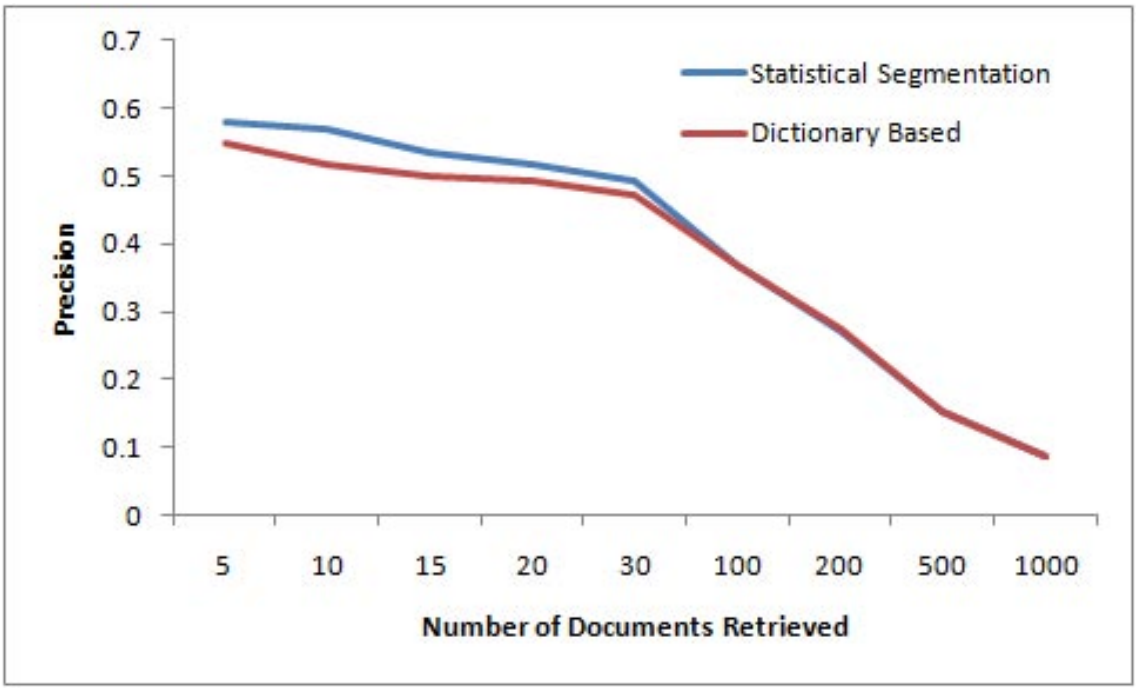

Fig. 10.2: Average Precision At X Documents Retrieved For Dictionary-based and Statistical Segmentation Approaches

	Statistical Segmentation Approach	Dictionary Based Approach
	Statistical Segmentation	Dictionary Based
5	0.5778	0.5481
10	0.5685	0.5167
15	0.5358	0.5012
20	0.5157	0.4944
30	0.4932	0.4722
100	0.3665	0.3663
200	0.2722	0.2741
500	0.1524	0.1511
1000	0.0852	0.0854
R-Precision	0.4084	0.3932

Tab. 10.4: Average Precision for Dictionary-based and Statistical Segmentation Approaches

10.5.3 Using Different N-grams

We used the suffix array based approach described in Chapter 9 to compute the term frequency and document frequency for all n-grams in TREC data. It can reduce the computation term frequency and document frequency over substrings ($N(N+1)/2$)) to a computation over classes ($2N-1$). The following algorithms are used to limit the size of the n-grams and filter out the substrings which are incomplete and lack of representative.

The iterative occurrences of substring in multi-documents is most likely to be a word or a phrase. Low frequency substrings are more likely to occur by chance. We only use n-grams occurring in more than two documents .

The length of n-gram will dramatically increase the number of n-grams produced. The longer n-gram, the larger space and longer time required to process. Most Chinese words are 2 character and less than 4 characters but some name entities can be more than 4 characters such as "亚太经济合作组织" (APEC) in query 17 and "菲律宾皮纳图博火山" (Philippine Mount Pinatubo Volcano) in query 47. We set the length of n-grams no more than 10 characters to reduce computing complexity.

We use different length of n-grams as indexes. The query are segmented as n-grams with the same length. Table 10.5 and 10.6 show the results of average precision and R-precision using different n-grams and its combinations as indexes. From the tables we note that a combination of uni-grams and bi-grams (<= 2) achieved best results of average precision of 0.4372. The precision slowly decreases as the length of n-grams grows. Using all n-grams with 10,694,137 indexes (0.3909) is only 8.8% better than single character around 7,000 (0.3593). This is possible that no all the longer n-grams carry so useful information as it is treated because of *idf* factor. Most of longer n-grams occur by chance. We have to prune them and extract useful information from them.

Table 10.5 shows that 2-grams with average precision of 0.4030 is best indexes without any combination. It is about 12.11%, 31.09% and 44.74% higher than 1-gram (0.3593), 3-grams (0.2777) and 4grams (0.2259) respectively. Figure 10.3

	1	2	3	4	<=2	<=3	<=4	All	>=3
0.0	0.7318	0.7964	0.7422	0.7187	0.7913	0.7961	0.8135	0.8138	0.7470
0.1	0.5977	0.6591	0.5429	0.4544	0.6827	0.6658	0.6594	0.6282	0.4897
0.2	0.5403	0.5812	0.4491	0.3661	0.6038	0.5886	0.5705	0.5452	0.4063
0.3	0.4769	0.5302	0.3728	0.2866	0.5522	0.5283	0.5061	0.4874	0.3355
0.4	0.4199	0.4800	0.3125	0.2470	0.5029	0.4752	0.4618	0.4401	0.2901
0.5	0.3776	0.4214	0.2705	0.2094	0.4583	0.4338	0.4197	0.4021	0.2476
0.6	0.3309	0.3723	0.2271	0.1738	0.4078	0.3903	0.3730	0.3571	0.2084
0.7	0.2770	0.3009	0.1785	0.1442	0.3607	0.3288	0.3125	0.3060	0.1653
0.8	0.2101	0.2343	0.1238	0.1053	0.2901	0.2582	0.2460	0.2404	0.1194
0.9	0.1193	0.1654	0.0509	0.0299	0.2061	0.1793	0.1768	0.1759	0.0515
1.0	0.0197	0.0329	0.0020	0.0016	0.0554	0.0530	0.0511	0.0478	0.0028
AP	0.3593	0.4030	0.2777	0.2259	0.4372	0.4162	0.4032	0.3909	0.2594

Tab. 10.5: Precision Recall and Average Precision For Different Length Of N-grams

	1	2	3	4	<=2	<=3	<=4	All	>=3
5	0.5370	0.6000	0.5481	0.4704	0.6185	0.5963	0.6000	0.6037	0.5407
10	0.5093	0.6074	0.5019	0.4593	0.6056	0.5963	0.5852	0.5685	0.4815
15	0.5074	0.5778	0.4728	0.4296	0.5815	0.5741	0.5494	0.5383	0.4506
20	0.4972	0.5537	0.4481	0.4009	0.5593	0.5528	0.5287	0.5157	0.4241
30	0.4698	0.5179	0.4204	0.3593	0.5272	0.5117	0.5019	0.4815	0.3870
100	0.3507	0.3707	0.2924	0.2448	0.3941	0.3794	0.3693	0.3572	0.2656
200	0.2575	0.2796	0.2156	0.1815	0.2958	0.2876	0.2798	0.2727	0.2021
500	0.1412	0.1527	0.1243	0.1071	0.1614	0.1589	0.1571	0.1547	0.1187
1000	0.0812	0.0851	0.0711	0.0624	0.0888	0.088	0.0868	0.0861	0.0689
RP	0.3900	0.4146	0.3092	0.2661	0.4408	0.419	0.4076	0.3928	0.2883

Tab. 10.6: Precision At X Documents and R-Precision For Different Length Of N-grams

Query 6: 国际社会对中共加入世界贸易组织所给予之支持 International Support of China's Membership in the WTO Description: 世界贸易组织，关贸总协，市场准入，世界贸易体系，多边贸易，成员（国） World Trade Organization (WTO), GATT, market access, world trade structure, multilateral trade, member nation
Query 7:中国大陆与台湾对南海诸岛的立场 Claims made by both PRC and Taiwan over islands in the South China Sea Description: 南沙（群岛），东沙（群岛），西沙（群岛），中国，台湾，主权 The Spratly Islands, the Dongsha Islands, the Xisha Islands, China, Taiwan, sovereignty
Query 29: 信息高速公路的建设 Building the Information Super Highway Description: 信息高速公路，建设 Information Super Highway, building
Query 34: 旱灾在中国造成的影响 The Impact of Droughts in China Description:旱灾，干旱地区，救灾款，粮食总产，面积，雨量，中国 drought, arid region, relief assistance, food production, area, rainfall, China
Query 51: 中国对保护环境的政策 China's Policy of Protecting the Environment Description:中国，环境，保护，酸雨，大气污染，水污染，空气污染，经济 China, environment, protection, acid rain, air pollution, water pollution, air pollution, economy 若文件提及世界性的环境问题或中国以外的环境污染问题则属非相关文件 Non-relevant documents discuss global environmental problems or problems with environmental pollution in other countries

Tab. 10.7: Title and Description of the Query 6,7,29,34,51

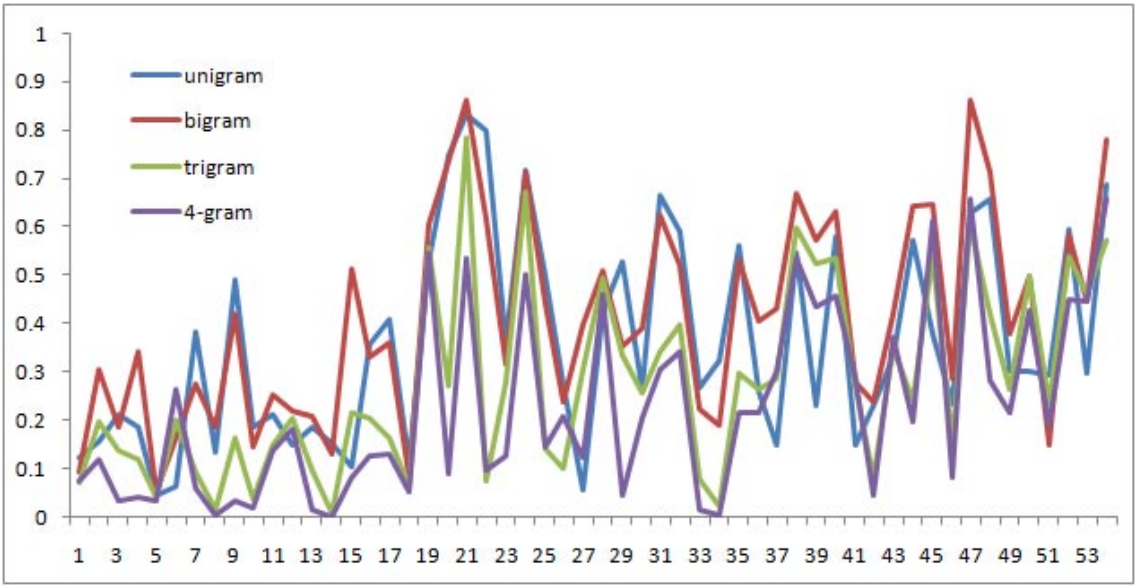

Fig. 10.3: Average Precision for 54 Queries Using 1-gram, 2-grams, 3-grams and 4-grams

shows the trend lines of average precision for 4 different n-grams across 54 queries. We might expect the similar results for each query. Indeed the trend line of bi-grams is almost the upper boundary of the others. This shows that bi-grams perform better across the 54 queries. The details of performance across 54 queries are shown in Table A.1. Bi-grams perform better in 32 queries comparing with uni-gram and 51 queries comparing with 3, 4 grams respectively out of total 54 queries. Bi-gram perform worst in Query 7 (-40.31), 29 (-49.73), 34 (-71.00) and 51 (-95.23) comparing with uni-gram and in Query 6 (-24.63, -64.56) and 51 (-56.85, -27.38) comparing with 3, 4-grams. Table 10.7 shows title and description of the Query 6,7,29,34,51. Details of each query are attached in the Appendix.

By inspecting the relevant documents retrieved, we find out the performance drop for bi-gram comparing with uni-gram mainly due to the inherent nature of Chinese language and the weakness of vector space model.

Query 7 is related to claims made by both PRC and Taiwan over islands in the South China Sea. The word "立场(claims)" is a weak index and islands in the South China Sea such as "南沙（群岛）,东沙（群岛）,西沙（群岛）" are

stronger indexes. Most retrieval errors in Query 7 are caused by documents retrieved relevant to South China Sea and its island but not to the claims. Modern information retrieval views documents as a bag of words. But not all documents containing the words of the query are relevant to the query. Document pd9107-2348 is the typical case of this weakness. Document pd9107-2348 reports a seminar discussing South China Sea and its island by scholars from Taiwan, China and ASEAN countries but not relevant to query.

For Query 29, most non-relevant documents retrieved are relevant to the information of building super highway, not building information super highway as bi-grams only capture the shorter information of "信息","高速","公路" and "建设". Table A.1 shows using n-grams greater than 4 even performs better than bi-gram with average precision of 0.3651.

Query 51 is related to China's policy of protecting the environment and documents discussing global environmental problems or problems with environmental pollution in other countries are considered non-relevant. For documents reported by People's Daily and Xinhua Newswire the word "中国" (China) will not appear in the documents if documents are related to China or the evens reported in the document occurred in China. Normally "我国"(my country), "全国"(entire country), the place of events occurred are used instead. Both relevant documents pd9104-2092 and pd9104-211 do not contain the word "中国"(China). Here the word "中国"(China) has negative impact on the query. The documents retrieved by the query are either relevant to China through the index word "中国"(China) or relevant to the environment through the index words "环境"(environment) or "污染"(pollution). We cannot use the word "中国"(China) to extract the documents relevant to China from those documents relevant to the environment as these documents may not contain the word "中国"(China). Because of these double impacts, bi-gram performs worst with average precision of 0.1490 in this query comparing with others.

For Query 34 there are many Chinese words in relevant documents that are closely related to word "旱灾"(drought disaster). These alternative words

are “春旱”(spring drought), “干旱”(dry drought), “旱区”(drought area), “旱情”(drought situation), “旱魔”(drought monster), “抗旱”(counter drought), “受旱” (being drought), “防旱” (avoid drought) etc. “旱灾” (drought disaster) does not even appear in the relevant document pd9105-832. So using single character “旱”(drought) performs better than using bi-grams “旱灾”(drought disaster). There are two reasons that lead to the average precision being lower. The retrieved documents are not relevant to queries due to negative impact of certain indexes. This is a form of type II error (false negative). The relevant documents could not be retrieved due to multi-representatives of objects. This is a form of type I error (false positive). Using “旱灾”(drought disaster) could not retrieve documents containing other forms of “旱”(drought). This shows using single character will improve the recall of information retrieval but it may have negative impact on the average precision as “旱”(drought) will carry different meanings when used together with other characters such as “旱冰”(roller skating inferring from the meaning of dry ice) and “旱烟”(dry tobacco) etc.

Comparing with 3,4-grams, bi-grams perform worse in Query 6 and 51. For Query 51 as mentioned above the documents relevant to China without the word “中国”(China) in it may not be retrieved. The documents retrieved relevant to the environment but not relevant to China will penalize the precision. For Query 6 “世界贸易组织”(WTO) can be segmented to “世界/贸易/组织” but “世界”(world), “贸易”(trade) and “组织”(organization) do not carry the meaning of WTO individually. From Table A.1 we note longer n-grams (4-grams: 0.2633) significantly perform better than shorter n-grams (1: 0.0598, 2: 0.1600, 3: 0.1994) in Query 6. This shows that using longer n-grams will increase the precision of information retrieval.

10.5.4 Word Extraction

Mutual information is commonly used to evaluate the correlation of substrings. We adapted the same mutual information metric as (Chien 1997) by observing mutual information of two overlapped patterns.

$$\begin{aligned} MI_{ab} &= \frac{Pr(c)}{Pr(a)+Pr(b)-Pr(c)} \\ &= \frac{\frac{f_c}{F}}{\frac{f_a}{F}+\frac{f_b}{F}-\frac{f_c}{F}} \\ &= \frac{f_c}{f_a+f_b-f_c} \end{aligned}$$

where c is the substring to be estimated, $c = c_1, c_2, \ldots, c_n$, a and b are the two longest composted substrings of c with the length $n-1$, i.e. $a = c1, \ldots, c_{n-1}$, $b = c_2, \ldots, c_n$, f_a,f_b and f_c are the frequencies of a, b and c. If MI_{ab} is large, it can be found that more of the time substrings a and b have to occur together. It seems that c is more complete in semantics than either a or b. For example a =”世界贸易组”, b =“界贸易组织” and c =“世界贸易组织”. A string can be extracted as long as it appears in the corpus and its value is large enough. It can effectively remove most of incomplete lexical patterns. For example, the incomplete lexical patterns such as ”世界贸易组” and “界贸易组织” can be removed, but “世界贸易组织” can be extracted. It is especially useful in extracting words like names, locations and technical terms reported by Chien (Chien 1997).

We use it to extract words and start from longest to shortest n-grams searching through suffix array. If longer substring is extracted, we excludes the shorter substrings of itself. Here we only consider words longer than 3 as we would like to evaluate the impact of the longer words on bi-grams at the same time. This algorithm depends mainly on the context rather than the frequencies of the patterns. Furthermore term frequency and document frequency are used in the extraction. We setup a threshold of 0.8 for MI, term frequency of 10 and document frequency of 5. In our experiment, we added 500 extracted words to the indexes through manual selection. Most of the words are related to topics. Here we will not evaluate the extraction algorithm but concentrate on the impact of these longer n-grams (words, compounds or collocations) on information retrieval. We use the statistics of these words from suffix array produced previously instead of re-segmenting the corpus and add them into the indexes (extra dimensions for these words in the document vectors). Extra data is collected for these words

	dic+word	1+word	2+word	1+2+word	seg+word
0.0	0.7436	0.7327	0.8033	0.8045	0.7746
0.1	0.6539	0.6299	0.6684	0.6841	0.6737
0.2	0.5704	0.5690	0.5858	0.6004	0.5989
0.3	0.5020	0.5205	0.5234	0.5524	0.5271
0.4	0.4655	0.4764	0.4741	0.5074	0.4756
0.5	0.4253	0.4179	0.4276	0.4685	0.4363
0.6	0.3748	0.3604	0.3801	0.4131	0.3922
0.7	0.3269	0.3127	0.3194	0.3599	0.3376
0.8	0.2625	0.2577	0.2516	0.2875	0.2688
0.9	0.1716	0.1686	0.1621	0.1983	0.191
1.0	0.0416	0.0409	0.0348	0.0626	0.0507
AP	0.4019	0.3963	0.4088	0.4398	0.4209

Tab. 10.8: The Impact of Word Extraction on TREC

only over the queries and added to original query vectors.

Table 10.8 and Figure 10.4 show that extra longer words extracted from the corpus help to improve the average precision with 9.54% for the dictionary based approach, 6.39% for statistical segmentation based approach, 10.30% for uni-gram, 1.44% for bi-grams and 0.59% for a combination of uni-gram and bi-grams. Table A.2 shows the details of the impact on individual queries. There is consistent improvement over the queries for dictionary based and statical segmentation approaches. Different words have different impact based on different approaches. Most long words have positive impact but some words have negative impact because of overweight of the words in the query. Table 10.9 show the comparison results of paired t-test obtained by corresponding to the different IR results over 54 queries for the different approaches before and after adding the extracted words in the indexes. In this table, the values of 2-tailed significance indicate the computed p-value. Both the dictionary-based approach and statistical segmentation approach after adding the extracted words performed significantly better than the original approaches at 0.0000883 and 0.000436 level. Although adding

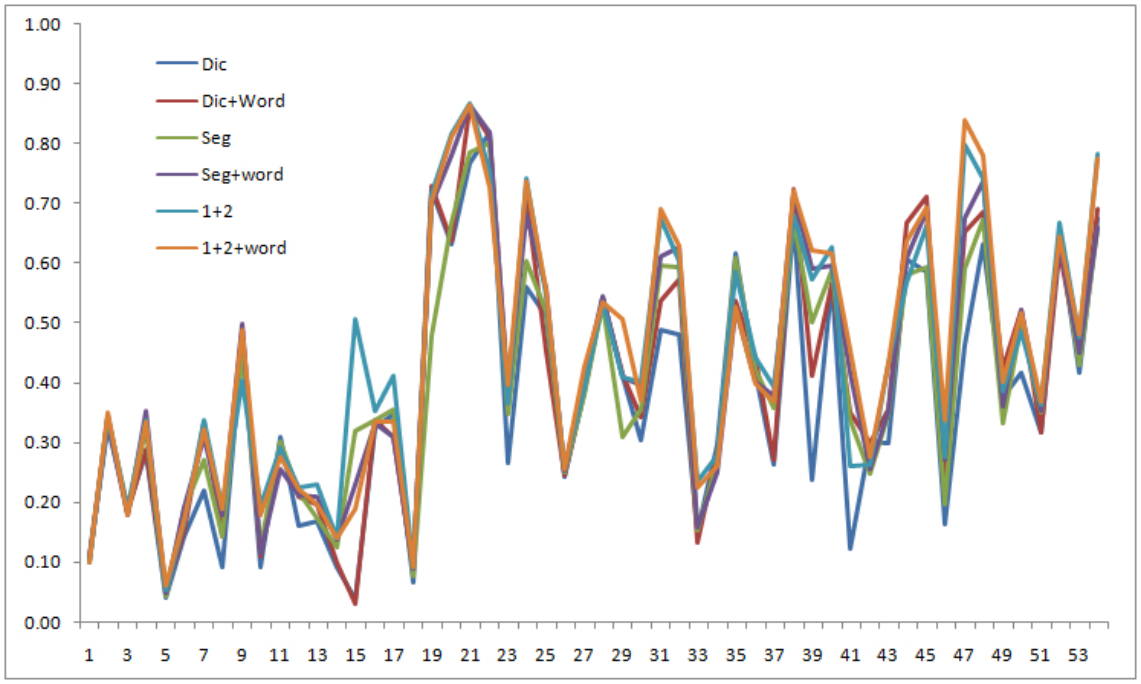

Fig. 10.4: The Impact of Extracted Words on 54 Queries

extra longer extracted words slightly improves the combination of uni-gram and bi-grams approach by 0.59%, it fails on paired samples t-test over all 54 queries and is not significantly better than the pure combination approach. However by observing the table A.2 there are some queries on which the performance is significantly improved by adding longer words in indexes.

Table 10.10 shows some examples of words and their impact on individual queries. The average precision of Query 15 and Query 17 are badly affected by adding extracted longer words for all three approaches. "联合国"(UN) and "维和部队"(Peace-keeping troops) make the retrieval precision decrease 62.47% for the approach of uni-gram and bi-grams. The title of Query 15 is "the UN peace-keeping troops help Haiti return to democracy". Adding these two new words makes the dimension of "海地"(Haiti, country name) weaker in the vector. This brings the retrieval documents more close to the topic of UN and Peace-keeping troops. In dictionary-based approach, the retrieval precision is only 0.0378 for Query 15 because "海地"(Haiti) is not in the dictionary and it is segmented as "海"(sea) and "地"(adverb marker or terra, ground). They are high frequent

53	Mean	Standard Deviation	Standard Error	95% confidence Lower	95% confidence Upper	t	sig (2-tailed)
Dic Dic+W	3.50E-02	6.05E-02	8.23E-03	1.84E-02	5.15E-02	-4.25	8.83E-05
Seg Seg+W	2.53E-02	6.75E-03	4.96E-02	1.18E-02	3.89E-02	3.75	4.36E-04
1+2 1+2+W	2.61E-03	6.03E-02	8.21E-03	-1.39E-02	1.91E-02	0.32	7.52E-01

Tab. 10.9: Paired samples t-test on IR results of combining extracted words (df=53)

Q_{id}	Words	Improvement(%) Dic	Seg	1+2
15	联合国UN 维和部队Peace-keeping troops	-23.02	-28.45	-62.47
17	亚太经济合作组织APEC	-11.53	-13.49	-18.66
19	希望工程Project Hope	1.01	46.31	-1.51
23	海湾战争Gulf War	47.33	9.19	8.53
29	信息高速公路Information Super Highway	0.41	32.77	23.93
41	特大桥very larger bridge			
	京九铁路Beijing-Kowloon Railway	189.18	23.49	75.34
45	红十字会Red Cross	21.77	16.33	4.63
46	中越关系Sino-Vietnamese relations	43.83	36.55	21.84
47	皮纳图博火山Mount Minatubo	40.92	14.29	5.21

Tab. 10.10: Improved Query Samples

characters. This causes their weight in the cosine product less because of *idf* factor. For unknown words such as "海地" a hybrid token representation including "海地","海" and "地" as indexes is likely to improve the precision. Obviously this happens to Query 17 "中国对亚太经济合作组织的期望"(China's Expectations about APEC). Correct segmentation of "亚太经济合作组织" in the corpus will result in the poor precision of the Query 17 as "亚太经济合作组织"(APEC) are not segmented as a compound for all three approaches. This may explain why higher precision of segmentation does not necessarily result in higher precision in Chinese IR. Different terms in query give the different weight in the vector. Some are more important and some are less important. These is a discrepancy for the term weight in the queries and in the corpus.

Other words in the table 10.10 have positive impact on the precision for three approaches. "希望工程"(Project Hope) and "信息高速公路"(information super highway) are already in the dictionary so there is not improvement in the Query 19 and Query 29 for dictionary-based approach. But if it is not in the dictionary, there is great improvement for Query 23, 45, 46 and 47. Statistical segmentation approach segments "希望工程" (Project Hope) and "信息高速公路" (information super highway) as "希望/工程"(hope/project) and "信息/高速/公路" (information/super/highway) incorrectly, thus adding the correct segmented words greatly improves the precision for statical segmentation approach. There is inconsistency for the bi-gram with uni-gram approach. There is not much improvement for Query 19, 45 and 47 but in contrast there is significant improvement for Query 29, 41 and 46. By observing these two groups of longer words, we find out that if any bi-grams (substring of the longer extracted word) can represent the statistics of its parent word, then no improvement will be achieved by adding the longer words. Otherwise adding longer extracted will improve the precision on bi-grams with uni-gram approach. Table 10.11 shows the term frequency and document frequency of the extracted longer words and the bi-grams of themselves. All bi-grams "希望" "望工" "工程" are substrings of word "希望工程", but only meaningless bi-gram "望工" inherits the statistical attributes from its parent string "希望工程". Thus the bi-grams "望工" functions as the word "希

Word/Bi-grams	Term Frequency	Document Frequency
希望工程(Project Hope)	1164	387
希望(hope)	22867	14692
望工(*meaningless bi-gram*)	1193	413
工程(project)	32099	12220
特大桥(very large bridge)	147	79
特大(very large)	2293	1654
大桥(large bridge)	2048	886

Tab. 10.11: Term Frequency and Document Frequency of Extracted Word and its Bi-grams

望工程" in the bi-grams approach. This is the robustness of bi-grams approach to unknown words comparing with word segmentation based approaches.

Adding extracted words to indexes will generally improve the performance of IR but it may have some side effects on some queries. It will drag the retrieved documents close to the dimension of the new words. Bi-grams are more robust to unknown words comparing with words.

10.5.5 Removing Stop Words

In modern information retrieval systems, more efficient indexing can be achieved by removal of stop words. Chinese language hardly has any grammatical inflections but it does makes use of grammatical particles to indicate aspect and mood. They carry little information to the document for purpose of IR. We normally call the set of such functional words "stop words" in information retrieval (Manning et al. 2008, Fox 1992, Van Rijsbergen 1979).

But some Chinese particles could be interpreted differently when used together with different characters. The character "地" (de, di) is equivalent to the suffix "-ly" in English and it carries a different meaning in the combination with different characters, such as "地面"(ground), "海地"(Haiti), etc. This brings some cautions using Chinese stop words in information retrieval. Several researchers

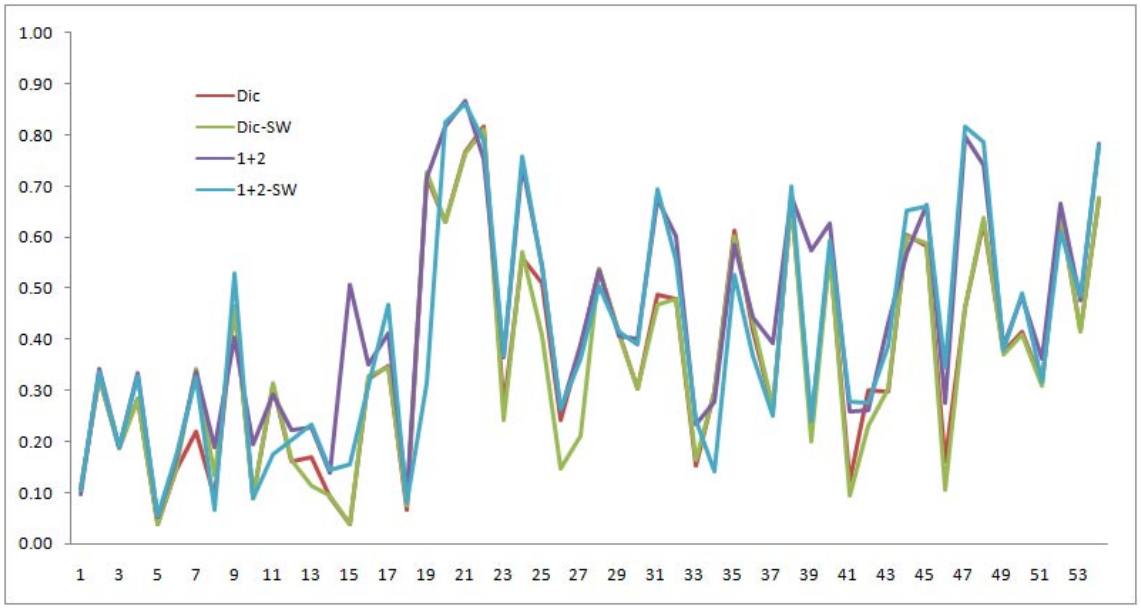

Fig. 10.5: The Impact of Stop Words on 54 Queries

have successfully applied Chinese stop words in information retrieval (Du, Zhang, Sun, Sun & Han 2000, Chen & Chen 2001), segmentation (Zou, Wang, Deng & Han 2006), word extraction (Nakagawa, Kojima & Maeda 2005)and document classification (Yang 1995, Hao & Hao 2008) to improve the performance but others (Kwok 1997) take cautions about using stop words in information retrieval.

A stop word list about 504 Chinese words from Center for Information Retrieval, Harbin Institute of Technology, China is used to evaluate its effect on IR performance. For the dictionary-based approach, we delete the word from the indexes if the word is in the stop word list. For the bi-grams with uni-gram approach, we delete the n-grams from the indexes if it and substrings are in the word list.

Table A.3 and Figure 10.5 show that there is negative impact on information retrieval using stop words for dictionary-based and bi-grams with uni-gram approaches. This is mainly caused by deleting some crucial words in documents and queries unexpectedly. The impact on dictionary-based approach (-2.16%) is far less than the bi-grams with uni-gram approach (-7.32%) because all the entries of the stop words are removed from the indexes. This causes more infor-

mation loss than the dictionary-based approach. It is surprising in Query 8 “地震在日本造成的损害与伤亡数据” (Numeric Indicators of Earthquake Severity in Japan) that dictionary-based approach gains 43.49% but in contrast bi-grams with uni-gram approach loses 180.30% of average precision. By looking through the documents retrieved, relevant documents, the query and the stop word list, we find out that both “地”(adverb marker, terra, ground) and “本”(classifier,current, root) are in the stop word list. For the bi-grams with uni-gram approach, the removal of all the entries related these two characters/words makes the critical words “地震”(earthquake) and “日本”(Japan) unindexable. Similarly the removal of “海地”(Haiti) in Query 15 results in 221.07% loss in average precision. As for the dictionary-based approach, “地震”(earthquake) is in the dictionary but “日本”(Japan) isn’t. “日本”(Japan) is segmented to “日”(date, day) and “本”(classifier, current, root). Most of documents have its published date in them. This dismisses importance of “日”(date, day) in the vector. But “本”(classifier, this, root) can be used together with other characters to be interpreted into different meanings. Particularly it is regularly used with “报”(newspaper) as “本报”(This Newspaper) at beginning of some documents in the corpus. All the documents with “本报”(This Newspaper) at beginning and related to earthquake will be retrieved with documents relevant Japan earthquake. Furthermore there are only 43 documents relevant to Query 8 in the dataset. Thus it is very sensitive to any change in the indexes. Removing “本” from the entries may make the precision close to the distribution of the documents relevant to Japan earthquake in all documents relevant to Earthquake. Although this is a great improvement of 43.49%, it only achieved 0.1323 precision. In other words, it only retrieves 5 relevant documents. This may happen by chance. Here we will not analyze any further for each queries. Significant precision loss is mainly caused by removal of the critical indexes. Removal of stop words will result in the reduction of indexes. For bi-grams with uni-gram approach, the indexes reduced 86.45% from 5,205,728 to 705,359 with 504 stop words but for the dictionary-based approach there is only the size of stop word list reduction in the indexes. Removal of stop words will improve the information retrieval with higher accuracy of segmentation but

it still runs the risks of accidentally deleting some unknown words maybe not in the dictionary. Furthermore most of the stop words are grammatical particles and occur frequently in the documents, the importance of these words will be diminished because of *idf* factor in information retrieval.

10.6 Discussion

From the previous experiments we note that there is inconsistent performance for each approach over 54 queries. Some approaches may work effectively in some queries but work poorly in others. This is mainly caused by the weakness of vector space model and the fuzzy nature of Chinese language.

Modern information retrieval treats documents and queries as bags of words. Each document is presented by a vector of words. The retrieval results are based on the ranked cosine product of document and query vectors. For three words "海地"(Haiti),"维和部队"(Peace-keeping troops) and "联合国"(UN) in Query 16 "联合国维和部队如何帮助海地恢复民主制度" (The UN peace-keeping troops help Haiti return to democracy), the documents containing "维和部队"(Peace-keeping troops) and "联合国"(UN) may rank before the documents containing either "海地"(Haiti) and "维和部队"(Peace-keeping troops)" or "海地"(Haiti) and "联合国"(UN). Incorrect segmentation of either "维和部队"(Peace-keeping troops) or "联合国"(UN) may help improve the performance of Query 15. This is proved by the sharp drop of precision for the bi-grams with uni-gram approach after adding "维和部队"(Peace-keeping troops) and "联合国"(UN). But incorrect segmentation of "联合国"(UN) will harm the precision of Query 16 "联合国对伊拉克经济制裁的辩论"(The Debate of UN Sanctions Against Iraq). Note that the concepts of "维和部队"(Peace-keeping troops) and "联合国"(UN) are not independent.

To approve our assumption first we make sure "海地"(Haiti), "伊拉克" (Iraq), "联合国"(UN) and "维和部队"(Peace-keeping troops) in the dictionary. Then we use the retrieval results as the base line and compare the results achieved after each removal of the word "联合国"(UN), "维和部队"(Peace-keeping troops) or

Q_{id}	Baseline	“联合国” (UN)	%	“维和部队” (P. K. troops)	%	Removing both	%
15	0.1899	0.3938	107.37	0.4720	148.55	27.96	47.24
16	0.3361	0.3245	-3.45	0.3361	0.00	0.3245	-3.45

Tab. 10.12: The Impact of Removal of “UN” and “Peace-keeping troops” on Queries 15 & 16

both from dictionary. Without the word in the dictionary, it will cause incorrect segmentation of words “联合国”(UN) and “维和部队”(Peace-keeping troops) as “联合/国”(unite, country) and “维/和/部队”(dimension, and, troop). Table 10.12 shows there is 107.37% gain of average precision for Query 15 and -3.45% loss of average precision for Query 16 after removal of the word “联合国”(UN). This means incorrect segmentation of “联合国”(UN) will improve the precision of Query 15 but it will hurts the precision of Query 16. In the other words, correct segmentation of “联合国”(UN) will make the other way around. For Query 16 the values of “联合国”(UN) and “维和部队”(Peace-keeping troops) dominate the cosine product of document vector and query vector. Breaking the domination actually helps the precision. Even incorrect segmentation of both words improves the precision. This shows the mismatch between tasks of word segmentation and informant retrieval. This inconsistent impact of segmentation makes hard to compare and evaluate the IR approaches.

The importance of words in corpus is weighed by inverse document frequency (idf) and is a fixed value, but the importance of different words in the same query and the importance of the same word in the different queries is varied. The importance of “联合国”(UN) in Query 15 and Query 16 is different. This may result in inconsistent results for different queries using the same approach. Incorrect segmentation of “联合国”(UN) improves the performance of Query 15 but in contrast it harms the performance of Query 16.

The importance of “海地”(Haiti), “联合国”(UN) and “维和部队”(Peace-keeping troops) in the Query 16 is different. This difference may be inconsistent with the value weighed by *idf*. In Query 16, we suppose to prefer the retrieved

documents more relevant to “海地”(Haiti) but “联合国”(UN) is weighed more important than “海地”(Haiti) by *idf* factor. Thus it may cause the retrieved documents relevant to “联合国”(UN) be ranked before documents relevant to “海地”(Haiti).

Human language is fuzzy and ambiguous. There may be many different words (synonyms) having the similar meaning or one word form (homonyms) having different meanings. This will affect the performance of IR. Synonym will reduce the recall as the documents containing different words which have same meaning will not be retrieved. Homonyms will reduce the precision as documents containing the same words which have different meaning will be retrieved. Each Chinese character has its own meaning. It may be interpreted to different meanings when used with other characters. Longer words increase the precision but risk the recall. Longer “希望工程”(Project Hope) in Query 19 and “信息高速公路”(information super highway) in Query 29 both improve the precision of the queries. Shorter words increase the recall but risk the precision. In Query 34, using “旱灾”(drought disaster) could not retrieve documents containing “干旱”(dry drought). Using shorter word “旱” will improve the recall but risk the precision as it may retrieve documents containing “旱冰”(roller skating inferred from the meaning of dry ice). There is some similar phenomenons in English such as “ guide dog”, “dog” and “hot dog”. To improve the performance further, we may need incorporate word sense information in future information retrieval.

10.7 Conclusion

In this chapter we have investigated performance of Chinese IR using words and n-grams as indexes. N-grams based approach (especially bi-grams with uni-gram) has the advantage of simplicity and robustness based on statistical information from corpus for Chinese IR. Word based approach relies on linguistic knowledge of the dictionary to segment the texts. It inherits the segmentation problems of segmentation ambiguity and unknown words. Longer words will improve the precision of IR. In contrast shorter words will increase the recall of the IR. The

fundamental problem of information retrieval is the balance between precision and recall. Although the bi-grams with uni-gram approach performed better than dictionary based approach in our experiment, with a larger dictionary and a better segmentation algorithm better IR results will be achieved. Higher accuracy of segmentation will improve the performance of IR to some extend but correct segmentation doesn't always have positive impact on the Chinese information retrieval.

Removal of stop words will improve the information retrieval with higher accuracy of segmentation but it still runs the risks of accidentally deleting some unknown words maybe not in the dictionary as some stop words maybe be used with other characters to form new unknown word. We suggest to keep them as they are to avoid such unexpected results.

Instead of focusing on accurate word segmentation, one should pay more attention to issues such as term weighting, word extraction, other retrieval models (Ponte & Croft 1998, Song & Croft 1999, Zhai 2008), morphological analysis and word sense disambiguation(Gao, Zhang, Liu & Liu 2006, Schutze & Pederson 1995, Krovetz & Croft 1992, Sanderson 1994) for information retrieval.

11. CONCLUSIONS

This chapter briefly reviews the thesis and gives an outlook for further research.

11.1 Thesis Review

We have proposed using a number of new learning algorithms for natural language processing and information retrieval based on unsupervised statistical n-gram models and machine learning. Statistical language modeling has a strong basis in information theory and removes many ad hoc procedures from traditional language learning systems. In this thesis, we have achieved improvements in context sensitive spelling correction in English, adaptive language modeling for Chinese Pinyin-to-character conversion, Chinese word segmentation and classification, and Chinese information retrieval.

1. **Context sensitive spelling Correction**

   We proposed a dynamic methodology for using n-gram contextual information to detect and correct real-word errors in English text. Contextual information is represented by the most frequent words and affixes to avoid the problems of high-dimension feature space. It gives us the syntactic cues to discriminate the confused words. These confused words are modeled by keyboard adjacency. We learn the contextual information surrounding the confused words from corpus and record the contexts which are significant and likely enough to disambiguate the confused words. Both significance and probability can be used as a function for detection and correction. We built an interface based on Word 97 (Huang & Powers 2001) and let users decide the balance between being bothered by false errors notifications and

missing some true errors according to the levels of significance and probability. Our spelling correction even found real errors in the WSJ corpus.

2. **Error-Driven Adaptive Language Modeling for Chinese Pinyin-to-character Conversion**

   We proposed using an error-driven adaptive approach based on n-gram models for automatic Chinese Pinyin-to-character conversion. The process of Pinyin-to-character conversion is to decode the Pinyin sequences into corresponding characters based on statistical n-gram models. The n-gram model will adapt itself if it can not make correct conversion, and diminish the outdated data by count scaling. Compression based smoothing method is used to solve the data sparseness. It has the advantage of updating the model incrementally and efficiently. Our approach can be applied to the traditional adaptive maximum a posteriori (MAP) models, and we mix the task independent model with task dependent model by error-driven adaption instead of a mixture coefficient. The Pinyin Chinese input is an interaction process between users and system of Pinyin input method. Users always correct characters from the Pinyin input system after conversion. This correction can be served as a reference transcript for the error function to adapt the model. Experiments show that the adaptive model significantly reduces Pinyin-to-Character conversion error rate.

3. **Unsupervised Chinese Word Segmentation**

   We proposed using contextual entropy (branching or boundary entropy) for Chinese word segmentation. Contextual entropy measures the branching factor after having seen the n-grams. It will be high if they are many possible symbols following the n-grams with none especially likely. As most Chinese words are only two characters, we only use a bi-gram model in our experiment. We compare the context entropy across the boundaries using different thresholds. Various thresholds are obtained from training on a small segmented data. We segment Chinese text by both trying to discovering the word boundaries and the non-boundaries. Our approach is

unsupervised without prior lexical knowledge and easy to adapt to other Asian languages such as Japanese.

4. **N-grams for Chinese Information Retrieval**

   We compared the performance of different length of n-gram based approaches, traditional dictionary based maximum match approach and statistical segmentation approach, adding longer unknown words and removal of stop words on Chinese information retrieval. Experiments on standard Chinese TREC 5 and TREC 6 data sets show different approaches perform inconsistently on the 54 queries. Some approaches work effectively in some queries but work poorly in others queries. Correct segmentation dose not necessarily lead to improve the precision of Chinese IR. This is mainly caused by the ambiguous nature of language and weakness of retrieval model. In our experiments a bi-gram approach with uni-grams and longer extracted words achieves best results by combining the advantage of shorter and longer words. It strikes a balance between the precision and recall. This analysis is of theoretical and practical importance to Chinese information retrieval.

11.2 Future Work

We have demonstrated the success of statistical n-gram language modeling and statistical language learning in solving various problems of Chinese natural language processing. However, there are still many research questions that need to be addressed in the near future.

The context sensitive spelling correction we developed here can only resolve lexical ambiguity in the syntactic sense as the semantic information are not represented in the n-gram context which we use to detect and correct real-word errors. Semantic information can be captured by word association in longer distance but it will significantly increase the learning complexity. As large web n-gram (Google and Microsoft) data are available recently, we should exploit more efficient learn-

ing algorithms on these data for context sensitive spelling correction. In addition we only model the confused words by keyboard adjacency. Better models can be built by analyzing real data such as misspelling searching queries entered by internet users.

Unsupervised word segmentation makes use of global statistics derived from a whole large scale unsegmented corpus to estimate the likelihood of a string being a word. It intends to derive a vocabulary from scratch and can be effective to solve the segmentation errors of unknown words. Due to recent rapid growth of segmented corpora and competition on Chinese Word Segmentation Bakeoffs (Sproat & Emerson 2003, Emerson 2005, Levow 2006, Jin & Chen 2008, Zhao & Liu 2010), supervised word segmentation has become the dominant approach for Chinese word segmentation. It trains a statistical model on a pre-segmented corpus to infer the optical segmentation. It only makes use of local information about individual characters and/or words within scope of one sentence. As the pre-segmented corpus will never cover the unknown words, unknown words still remain a prominent issue. It is worthwhile to explore the feasibility of integrating both supervised and unsupervised word segmentation for enhancing the performance.

The main purpose of word segmentation is to serve other applications such as POS tagging and machine translation for post-processing. Different applications have different interests in words. It is interesting to explore whether performing post-processing strictly after segmentation or performing segmentation and post-precessing simultaneously is more effective for the applications.

The goal of all information retrieval system is to rank documents accurately for a given query. Word sense ambiguity is a major cause of inconsistent performance in Chinese IR systems, it is thus important to further investigate the relationship between word sense disambiguation and IR.

APPENDIX

A. THE APPENDIX: TABLES FOR TREC 5 & 6 CHINESE INFORMATION RETRIEVAL RESULTS

Tab. A.1: Average Precision for 54 Queries Using 1,2,3,4-grams

Q_{id}	1 gram	2 grams	3 grams	4 grams	>4 grams	1 vs 2 (%)	2 vs 3 (%)	2 vs 4 (%)
1	0.1222	0.0924	0.0685	0.0729	0.0037	-32.25	25.87	21.10
2	0.1538	0.3032	0.1950	0.1187	0.0052	49.27	35.69	60.85
3	0.2105	0.1841	0.1382	0.0327	0.0345	-14.34	24.93	82.24
4	0.1840	0.3397	0.1197	0.0408	0.0161	45.83	64.76	87.99
5	0.0434	0.0508	0.0419	0.0332	0.0276	14.57	17.52	34.65
6	0.0598	0.1600	0.1994	0.2633	0.2426	62.63	-24.63	-64.56
7	0.3822	0.2724	0.0921	0.0601	0.0183	-40.31	66.19	77.94
8	0.1334	0.1849	0.0141	0.0019	0.0000	27.85	92.37	98.97
9	0.4890	0.4202	0.1642	0.0330	0.0105	-16.37	60.92	92.15
10	0.1848	0.1432	0.0362	0.0178	0.0136	-29.05	74.72	87.57
11	0.2037	0.2489	0.1475	0.1378	0.0705	18.16	40.74	44.64
12	0.1475	0.2173	0.2044	0.1817	0.1544	32.12	5.94	16.38
13	0.1838	0.2056	0.0944	0.0150	0.0000	10.60	54.09	92.70
14	0.1510	0.1280	0.0087	0.0005	0.0004	-17.97	93.20	99.61
15	0.0988	0.5136	0.2147	0.0799	0.0311	80.76	58.20	84.44
16	0.3550	0.3304	0.2025	0.1244	0.0610	-7.45	38.71	62.35
17	0.4070	0.3592	0.1629	0.1299	0.1597	-13.31	54.65	63.84
18	0.1098	0.0795	0.0625	0.0509	0.0253	-38.11	21.38	35.97

Continued on Next Page...

Table A.1 – Continued

Q_{id}	1 gram	2 grams	3 grams	4 grams	>4 grams	1 vs 2 (%)	2 vs 3 (%)	2 vs 4 (%)
19	0.5190	0.5997	0.5575	0.5458	0.0840	13.46	7.04	8.99
20	0.7484	0.7351	0.2690	0.0887	0.0047	-1.81	63.41	87.93
21	0.8334	0.8633	0.7856	0.5353	0.2084	3.46	9.00	37.99
22	0.7994	0.6182	0.0744	0.0958	0.0667	-29.31	87.97	84.50
23	0.3588	0.3170	0.2779	0.1254	0.0447	-13.19	12.33	60.44
24	0.7168	0.7115	0.6706	0.5030	0.2364	-0.74	5.75	29.30
25	0.4968	0.4410	0.1425	0.1459	0.0686	-12.65	67.69	66.92
26	0.2744	0.2362	0.1014	0.2062	0.0964	-16.17	57.07	12.70
27	0.0558	0.3979	0.3038	0.1211	0.0358	85.98	23.65	69.57
28	0.4230	0.5090	0.4931	0.4596	0.2791	16.90	3.12	9.71
29	0.5239	0.3499	0.3328	0.0449	0.3651	-49.73	4.89	87.17
30	0.2582	0.3877	0.2562	0.2047	0.0868	33.40	33.92	47.20
31	0.6656	0.6245	0.3398	0.3054	0.0230	-6.58	45.59	51.10
32	0.5889	0.5217	0.3974	0.3423	0.3558	-12.88	23.83	34.39
33	0.2667	0.2229	0.0782	0.0131	0.0082	-19.65	64.92	94.12
34	0.3208	0.1876	0.0215	0.0044	0.0005	-71.00	88.54	97.65
35	0.5609	0.5305	0.2976	0.2159	0.0236	-5.73	43.90	59.30
36	0.2488	0.3984	0.2624	0.2146	0.0462	37.55	34.14	46.13
37	0.1421	0.4275	0.2857	0.3000	0.0788	66.76	33.17	29.82
38	0.5471	0.6689	0.5961	0.5349	0.3616	18.21	10.88	20.03
39	0.2289	0.5711	0.5235	0.4333	0.2250	59.92	8.33	24.13
40	0.5780	0.6297	0.5365	0.4573	0.4276	8.21	14.80	27.38
41	0.1463	0.2759	0.2678	0.2841	0.1076	46.97	2.94	-2.97
42	0.2293	0.2362	0.0803	0.0431	0.0053	2.92	66.00	81.75
43	0.3289	0.4183	0.3503	0.3728	0.2539	21.37	16.26	10.88
44	0.5731	0.6424	0.2420	0.1964	0.0183	10.79	62.33	69.43
45	0.3752	0.6459	0.5378	0.6133	0.6187	41.91	16.74	5.05

Continued on Next Page...

Table A.1 – Continued

Q_{id}	1 gram	2 grams	3 grams	4 grams	>4 grams	1 vs 2 (%)	2 vs 3 (%)	2 vs 4 (%)
46	0.2253	0.2784	0.1521	0.0797	0.0647	19.07	45.37	71.37
47	0.6271	0.8615	0.6069	0.6558	0.8166	27.21	29.55	23.88
48	0.6549	0.7118	0.4208	0.2810	0.1073	7.99	40.88	60.52
49	0.3022	0.3795	0.2624	0.2150	0.2299	20.37	30.86	43.35
50	0.2984	0.4971	0.4991	0.4253	0.3798	39.97	-0.40	14.44
51	0.2909	0.1490	0.2337	0.1898	0.0917	-95.23	-56.85	-27.38
52	0.5916	0.5761	0.5375	0.4491	0.3836	-2.69	6.70	22.04
53	0.2952	0.4434	0.4622	0.4442	0.3166	33.42	-4.24	-0.18
54	0.6866	0.7783	0.5702	0.6559	0.5511	11.78	26.74	15.73
AP	0.3593	0.4030	0.2777	0.2259	0.1472	10.84	31.09	43.95

Tab. A.2: The Impact of Extracted Words on 54 Queries

Q_{id}	Dic	D+W	%	Seg	S+W	%	1+2	1,2+W	%
1	0.1069	0.1076	0.65	0.1022	0.1048	2.54	0.0985	0.0994	0.91
2	0.3271	0.3406	4.13	0.3465	0.3344	-3.49	0.3433	0.3505	2.10
3	0.1899	0.1831	-3.58	0.1823	0.1802	-1.15	0.1911	0.1786	-6.54
4	0.2829	0.2872	1.52	0.3244	0.3524	8.63	0.3360	0.3348	-0.36
5	0.0391	0.0514	31.46	0.0444	0.0487	9.68	0.0535	0.0616	15.14
6	0.1436	0.1492	3.90	0.1894	0.1918	1.27	0.1649	0.1705	3.40
7	0.2185	0.3328	52.31	0.2706	0.3103	14.67	0.3362	0.3237	-3.72
8	0.0922	0.1427	54.77	0.1440	0.1791	24.38	0.1892	0.1896	0.21
9	0.4651	0.4886	5.05	0.4503	0.4991	10.84	0.4037	0.4898	21.33
10	0.0910	0.1078	18.46	0.1256	0.1156	-7.96	0.1948	0.1791	-8.06
11	0.3081	0.2982	-3.21	0.3014	0.2559	-15.10	0.2916	0.2765	-5.18
12	0.1605	0.2082	29.72	0.2174	0.2135	-1.79	0.2246	0.2222	-1.07
13	0.1695	0.1957	15.46	0.1740	0.2107	21.09	0.2301	0.1964	-14.65
14	0.0907	0.0979	7.94	0.1264	0.1378	9.02	0.1411	0.1418	0.50
15	0.0378	0.0291	-23.02	0.3202	0.2291	-28.45	0.5060	0.1899	-62.47
16	0.3239	0.3353	3.52	0.3369	0.3331	-1.13	0.3517	0.3361	-4.44
17	0.3478	0.3077	-11.53	0.3566	0.3085	-13.49	0.4126	0.3356	-18.66
18	0.0663	0.0856	29.11	0.0773	0.0896	15.91	0.0919	0.0925	0.65
19	0.7200	0.7273	1.01	0.4776	0.6988	46.31	0.7150	0.7042	-1.51
20	0.6306	0.6352	0.73	0.6676	0.7796	16.78	0.8165	0.8109	-0.69
21	0.7666	0.8617	12.41	0.7852	0.8649	10.15	0.8664	0.8646	-0.21
22	0.8180	0.8094	-1.05	0.7994	0.8199	2.56	0.7534	0.7255	-3.70
23	0.2658	0.3916	47.33	0.3483	0.3803	9.19	0.3658	0.3970	8.53
24	0.5604	0.7101	26.71	0.6021	0.6813	13.15	0.7406	0.7374	-0.43
25	0.5103	0.4526	-11.31	0.5197	0.5568	7.14	0.5437	0.5562	2.30
26	0.2424	0.2435	0.45	0.2504	0.2529	1.00	0.2591	0.2549	-1.62
27	0.3850	0.3842	-0.21	0.3811	0.4123	8.19	0.3909	0.4262	9.03

Continued on Next Page...

Table A.2 – Continued

Q_{id}	Dic	D+W	%	Seg	S+W	%	1+2	1,2+W	%
28	0.5390	0.5422	0.59	0.5316	0.5443	2.39	0.5352	0.5339	-0.24
29	0.4121	0.4138	0.41	0.3088	0.4100	32.77	0.4078	0.5054	23.93
30	0.3028	0.3425	13.11	0.3551	0.3957	11.43	0.4009	0.3713	-7.38
31	0.4880	0.5371	10.06	0.5960	0.6104	2.42	0.6742	0.6909	2.48
32	0.4800	0.5726	19.29	0.5935	0.6270	5.64	0.6028	0.6306	4.61
33	0.1515	0.1316	-13.14	0.1546	0.1588	2.72	0.2350	0.2243	-4.55
34	0.2968	0.2828	-4.72	0.2699	0.2506	-7.15	0.2795	0.2610	-6.62
35	0.6151	0.5355	-12.94	0.6087	0.5246	-13.82	0.5845	0.5278	-9.70
36	0.4172	0.4411	5.73	0.4218	0.4027	-4.53	0.4421	0.3999	-9.55
37	0.2638	0.2693	2.08	0.3594	0.3779	5.15	0.3937	0.3676	-6.63
38	0.6704	0.7232	7.88	0.6706	0.7158	6.74	0.6804	0.7204	5.88
39	0.2385	0.4102	71.99	0.5021	0.5913	17.77	0.5734	0.6210	8.30
40	0.5684	0.5649	-0.62	0.5875	0.5975	1.70	0.6272	0.6162	-1.75
41	0.1211	0.3502	189.18	0.3380	0.4174	23.49	0.2595	0.4550	75.34
42	0.3016	0.2984	-1.06	0.2474	0.2548	2.99	0.2635	0.2752	4.44
43	0.2979	0.3557	19.40	0.3486	0.3526	1.15	0.4319	0.4330	0.25
44	0.6055	0.6665	10.07	0.5795	0.6096	5.19	0.5684	0.6373	12.12
45	0.5844	0.7116	21.77	0.5927	0.6895	16.33	0.6629	0.6936	4.63
46	0.1622	0.2333	43.83	0.1978	0.2701	36.55	0.2770	0.3375	21.84
47	0.4631	0.6526	40.92	0.5915	0.6760	14.29	0.7965	0.8380	5.21
48	0.6312	0.6862	8.71	0.6708	0.7357	9.68	0.7417	0.7799	5.15
49	0.3753	0.4228	12.66	0.3329	0.3597	8.05	0.3865	0.4009	3.73
50	0.4164	0.5216	25.26	0.4992	0.5214	4.45	0.4853	0.5177	6.68
51	0.3156	0.3168	0.38	0.3514	0.3532	0.51	0.3633	0.3681	1.32
52	0.6448	0.6199	-3.86	0.6471	0.6350	-1.87	0.6672	0.6437	-3.52
53	0.4172	0.4449	6.64	0.4290	0.4505	5.01	0.4754	0.4802	1.01
54	0.6743	0.6902	2.36	0.6595	0.6612	0.26	0.7823	0.7752	-0.91
AP	0.3669	0.4019	9.54	0.3957	0.4210	6.39	0.4372	0.4398	0.59

Tab. A.3: The Impact of Stop Words on 54 Queries

Q_{id}	Dic	Dic-SW	%	1+2	1+2-SW	%
1	0.1069	0.1075	0.56	0.0985	0.1068	7.77
2	0.3271	0.3274	0.09	0.3433	0.3382	-1.51
3	0.1899	0.1865	-1.79	0.1911	0.1911	0.00
4	0.2829	0.2835	0.21	0.3360	0.3299	-1.85
5	0.0391	0.0398	1.79	0.0535	0.0558	4.12
6	0.1436	0.1470	2.37	0.1649	0.1738	5.12
7	0.2185	0.3421	56.57	0.3362	0.3261	-3.10
8	0.0922	0.1323	43.49	0.1892	0.0675	-180.30
9	0.4651	0.4668	0.37	0.4037	0.5319	24.10
10	0.0910	0.0883	-2.97	0.1948	0.0888	-119.37
11	0.3081	0.3163	2.66	0.2916	0.1767	-65.03
12	0.1605	0.1631	1.62	0.2246	0.2047	-9.72
13	0.1695	0.1150	-32.15	0.2301	0.2353	2.21
14	0.0907	0.0949	4.63	0.1411	0.1457	3.16
15	0.0378	0.0385	1.85	0.5060	0.1576	-221.07
16	0.3239	0.3285	1.42	0.3517	0.3098	-13.52
17	0.3478	0.3453	-0.72	0.4126	0.4703	12.27
18	0.0663	0.0735	10.86	0.0919	0.0820	-12.07
19	0.7200	0.7288	1.22	0.7150	0.3149	-127.06
20	0.6306	0.6302	-0.06	0.8165	0.8278	1.37
21	0.7666	0.7653	-0.17	0.8664	0.8630	-0.39
22	0.8180	0.8121	-0.72	0.7534	0.7901	4.64
23	0.2658	0.2415	-9.14	0.3658	0.3695	1.00
24	0.5604	0.5725	2.16	0.7406	0.7599	2.54
25	0.5103	0.4076	-20.13	0.5437	0.5418	-0.35
26	0.2424	0.1482	-38.86	0.2591	0.2651	2.26

Continued on Next Page...

Table A.3 – Continued

Q_{id}	Dic	Dic-SW	%	1+2	1+2-SW	%
27	0.3850	0.2130	-44.68	0.3909	0.3634	-7.57
28	0.5390	0.5378	-0.22	0.5352	0.5055	-5.88
29	0.4121	0.4098	-0.56	0.4078	0.4175	2.32
30	0.3028	0.3043	0.50	0.4009	0.3904	-2.69
31	0.4880	0.4698	-3.73	0.6742	0.6954	3.05
32	0.4800	0.4809	0.19	0.6028	0.5551	-8.59
33	0.1515	0.1635	7.92	0.2350	0.2438	3.61
34	0.2968	0.2975	0.24	0.2795	0.1426	-96.00
35	0.6151	0.6015	-2.21	0.5845	0.5271	-10.89
36	0.4172	0.4360	4.51	0.4421	0.3677	-20.23
37	0.2638	0.2663	0.95	0.3937	0.2523	-56.04
38	0.6704	0.6728	0.36	0.6804	0.7002	2.83
39	0.2385	0.1999	-16.18	0.5734	0.2404	-138.52
40	0.5684	0.5689	0.09	0.6272	0.5946	-5.48
41	0.1211	0.0931	-23.12	0.2595	0.2801	7.35
42	0.3016	0.2306	-23.54	0.2635	0.2760	4.53
43	0.2979	0.3041	2.08	0.4319	0.3871	-11.57
44	0.6055	0.6015	-0.66	0.5684	0.6523	12.86
45	0.5844	0.5877	0.56	0.6629	0.6632	0.05
46	0.1622	0.1069	-34.09	0.2770	0.3465	20.06
47	0.4631	0.4629	-0.04	0.7965	0.8172	2.53
48	0.6312	0.6380	1.08	0.7417	0.7871	5.77
49	0.3753	0.3705	-1.28	0.3865	0.3804	-1.60
50	0.4164	0.4103	-1.46	0.4853	0.4910	1.16
51	0.3156	0.3096	-1.90	0.3633	0.3194	-13.74
52	0.6448	0.6511	0.98	0.6672	0.6105	-9.29
53	0.4172	0.4157	-0.36	0.4754	0.4856	2.10
54	0.6743	0.6775	0.47	0.7823	0.7816	-0.09

Continued on Next Page...

Table A.3 – Continued

Q_{id}	Dic	Dic-SW	%	1+2	1+2-SW	%
AP	0.3669	0.3590	-2.16	0.4372	0.4074	-7.32

B. THE APPENDIX: EXAMPLES OF TREC 5 & 6 CHINESE QUERIES

Number: CH1

Title: U.S. to separate the most-favored-nation status from human rights issue in China.

美国决定将中国大陆的人权状况与其是否给予中共最惠国待遇分离.

Description: most-favored nation status, human rights in China, economic sanctions, separate, untie

最惠国待遇，中国，人权，经济制裁，分离，脱钩

Narrative: A relevant document should describe why the U.S. separates most-favored nation status from human rights. A relevant document should also mention why China opposes U.S. attempts to tie human rights to most-favored-nation status.

相关文件必须提到美国为何将最惠国待遇与人权分离；相关文件也必须提到中共为什么反对美国将人权与最惠国待遇相提并论.

Number: CH2

Title: Communist China's position on reunification

中共对于中国统一的立场

Description: China, one-nation-two-systems, Taiwan, peaceful reunification, economic and trade cooperation, cross-strait relationship, science and technology exchanges

中国，一国两制，台湾，和平统一，经贸合作，两岸关系，科技丶文化交流

Narrative: A relevant document should describe how China wishes to reach reunification through the implementation of "one-nation-two-systems." If a document

merely states a foreign nation's support of China's sovereignty over Taiwan or discusses trade cooperation as well as cultural and technical exchanges between China and a country other than Taiwan, then the document is irrelevant.
相关文件必须提到中共如何经由实现一国两制来达到台湾与大陆统一的目的.如果文件只是外国政府重申支持中共对台湾拥有主权或提到中共与其他国家之经贸、科技、文化交流，则为不相关文件.

Number: CH3
Title: The operational condition of nuclear power plants in China.
中共核电站之营运情况
Description: nuclear power plant, Daya Bay (nuclear power plant), Qinshan (nuclear power plant), safety
核电站，大亚湾，秦山，安全
Narrative: A relevant document should contain information on the current safety practices in China's nuclear power plants. Any article on safety regulations, accident reports and safety practices are relevant.
相关文件必须提到中国目前投产的核电站的安全营运情况．任何有关安全之规则或法令，安全措施之执行，意外事故报告之文件皆属相关文件.

Number: CH4
Title: The newly discovered oil fields in China.
中国大陆新发现的油田
Description: oil field, natural gas, oil and gas, oil reserves, oil quality
油田，天然气，油气，储量，油质
Narrative: A relevant document should contain information on the oil reserves in the newly discovered oil fields in Mainland China, any concrete description of specific oil fields, or China's plan to develop these fields.
相关文件必须提到中国大陆近几发现的油田的储量，各油田的特点，以及中国开发油田的计划.

Number: CH5

Title: Regulations and Enforcement of Intellectual Property Rights in China
中国有关知识产权的立法与政策以及执法情况

Description: intellectual property rights, trade mark, copyright, patent.
知识产权法，商标法，著作权法，专利法

Narrative: A relevant document should describe laws established in China to protect intellectual property rights. If a document contains information such as: China's violation of intellectual property rights as the basis for imposing trade sanctions against China; or, China taking up intellectual property rights as part of its economic reform, then the document is irrelevant.
相关文件必须提到中国有关保护知识产权的法律．非相关文件包括将中国违反知识产权作为对中国贸易制裁之依据或中国以知识产权作为经济改革的项目．

Number: CH6

Title: International Support of China's Membership in the WTO
国际社会对中共加入世界贸易组织所给予之支持

Description: World Trade Organization (WTO), GATT, market access, world trade structure, multilateral trade, member nation
世界贸易组织，关贸总协，市场准入，世界贸易体系，多边贸易，成员（国）

Narrative: A relevant document should indicate support given by specific nation(s) for China's membership in WTO.
相关文件必须提到某一国家或某些国家对中国加入世界贸易组织所给予之支持．

Number: CH7

Title: Claims made by both PRC and Taiwan over islands in the South China Sea
中国大陆与台湾对南海诸岛的立场

Description: The Spratly Islands, the Dongsha Islands, the Xisha Islands, China, Taiwan, sovereignty
南沙（群岛），东沙（群岛），西沙（群岛），中国，台湾，主权

Narrative: A relevant document should include the following information: (1) why the Spratly Islands became the disputed area among China, the Philippines, Vietnam, and Indonesia; or (2) what are the natural resources found in the South China Sea; or/and (3) what are the sovereign rights claimed by the PRC and Taiwan; or/and (4) what are the suggestions proposed by the ASEAN to solve the territorial dispute over the Spratly Islands and South China Sea.
相关文件应包括下列信息: (1)为何南沙群岛成为中国、菲律宾、越南、印尼等国冲突的所在地；(2)南海有那些天然资源: (3)中国大陆与台湾对南海诸岛之主权立场为何；以及(4)东盟国家对解决南沙群岛与南海争端有什么建议.

Number: CH8
Title: Numeric Indicators of Earthquake Severity in Japan
地震在日本造成的损害与伤亡数据
Description: Japan, earthquake, damage, death, injury, Richter scale
日本，地震，损失，死亡，级，受伤, 芮氏地震仪
Narrative: A relevant document should contain numeric indicators such as the magnitude of the earthquake, number of deaths or injuries, or property damage.
相关文件应包括地震的级数以及所造成的实际损害与伤亡数字,诸如地震在芮氏地震仪上的级数,死亡与受伤人数,以及以金钱为单位的财产损失数目.

Number: CH9
Title: Drug Problems in China
中国毒品问题
Description: narcotics, cocaine, heroin, marijuana, ton(s), kilogram(s), drugs use, drugs sale
毒品, 可卡因, 大麻, 海洛因, 吨, 公吨, 吸食毒品, 毒品买卖
Narrative: A relevant document should contain information on drug problems in China, how the government cracks down on illegal drug activities, what types of drug rehabilitation program exist in China, and how the Chinese government cooperates with international organizations to stop the spread of drug trafficking.
相关文件应包括目前毒品在中国所造成的危害,中国打击非法买卖毒品的措

施,有否有戒毒设施,以及中国是否与国际执法组织合作来遏制国际毒贩的走私活动.

Number: CH10

Title: Border Trade in Xinjiang

新疆的边境贸易

Description: Xinjiang, Uigur, border trade, market

新疆,维吾尔,边境贸易,边贸,市场

Narrative: A relevant document should contain information on the trading relationship between Xinjiang, China and its neighboring nations, including treaties signed by China and former Soviet Republics that are bordering China and foreign investment. If a document contains information on how China develops Xinjiang, it is not relevant.

相关文件必须包括中国新疆与其邻近国家的贸易关系,此关系包括中国与前苏联共和国之间所签署的贸易条约以及彼此间的外贸投资.如果文件只论及中国如何建设发展新疆，则属非相关文件.

Number: CH11

Title: UN Peace-keeping Force in Bosnia

联合国驻波斯尼亚维和部队

Description: Bosnia, Former Yugoslavia, Balkan, U.N., NATO, Muslim, weapon sanction, peace-keeping

波斯尼亚，前南斯拉夫,巴尔干,联合国,北约,武器禁运,维和,维持和平

Narrative: A relevant document should contain information on how UN peace-keeping troops carry out their mission in the war-torn Bosnia.

相关文件必须包括联合国和平部队如何在战火柔蔺(足旁roulin)的波斯尼亚进行维持和平的任务.

Number: CH12

Title: World Conference on Women

世界妇女大会

Description: UN, world, women's conference, women's issues, women's status

联合国,世界,妇女大会,妇女问题,妇女地位

Narrative: A relevant document should contain information on the 4th World Conference on Women, especially on ways to improve women's social status and economic situations through education and legislation.

相关文件必须是关于第四届世界妇女大会中讨论的妇女问题,特别是经由教育和立法来改进妇女的社会地位和经济情况的措施.

Number: CH13

Title: China Bids for 2000 Olympic Games

中国争取举办西元2000年奥运

Description: China, economic strength, Olympic games, preparatory work

中国,经济实力,奥运,世界运动大会,奥林匹克,筹备工作

Narrative: A relevant document should contain information on how China bids for the 2000 Olympic Games, China's reasons for sponsoring the 2000 Olympic games.

相关文件必须包括中国如何争取举办西元2000年奥运,中国所持的理由为何. 中国选手在奥运会中的表现属于不相关文件.

Number: CH14

Title: Cases of AIDS in China

中国的爱滋病例

Description: China, Yunnan, AIDS, HIV, high risk group, syringe, virus

中国,云南,爱滋病,HIV,高危险群患者,注射器,病毒

Narrative: A relevant document should contain information on the areas in China that have the highest AIDS cases, how the AIDS virus was transmitted, and how the Chinese government combats AIDS problem.

相关文件应当包括中国那些地区的爱滋病例最多,爱滋病毒在中国是如何传播的,以及中国政府如何监测爱滋病并控制它的传染.

Number: CH15

Title: The UN peace-keeping troops help Haiti return to democracy
联合国维和部队如何帮助海地恢复民主制度

Description: Haiti, UN, U.S., multination-troops, peace-keeping troops, democracy
海地,联合国,美国,多国部队,维和部队,民主

Narrative: A relevant document should contain information on the U.S. efforts to help Haiti resume its democracy, UN resolutions on Haiti, and the Latin-American nations reactions to the UN resolutions.
相关文件必须提到美国如何帮助海地民主政府重建海地;联合国安理会对海地问题之决议,以及拉美国家对联合国决议之反应. 不相关文件则为海地仅为新闻或电视广播提要,或新闻分析中提及海地但新闻主题不在海地.

Number: CH16

Title: The Debate of UN Sanctions Against Iraq
联合国对伊拉克经济制裁的辩论

Description: UN, Iraq, economic sanction
联合国,伊拉克,经济制裁

Narrative: A relevant document should contain information on why the UN carries out economic sanctions against Iraq; the impact of the economic sanctions on Iraq; the UN debate on when to lift the sanctions; Iraq‘s reaction to the sanctions. An irrelevant document is such that it only mentions the UN sanctions against Iraq but does not give any details on the discussions, impact, and Iraq‘s reaction about the sanctions. Non-relevant documents include summaries without any details like the French government’s setting up a representative office in Iraq thus reducing its economic sanctions toward Iraq, Iran‘s criticizing of the UN sanctions when seeking diplomatic relations with Iraq, or UN sanctions against Iraq.
相关文件应提到联合国为何对伊拉克实施经济制裁; 经济制裁对伊拉克的影响;联合国对何时解除此经济制裁的辩论;以及伊拉克对经济制裁的反应.不相关文件为法国为了在伊拉克设代表处而减少其对伊之经济制裁; 中国对联合国在

中东维和行动的评论;伊拉克与伊朗关系正常化中批评联合国之制裁; 或联合国对伊拉克之经济制裁仅为新闻提要而未详细报道.

Number: CH17
Title: China's Expectations about APEC
中国对亚太经济合作组织的期望
Description: APEC, China, GATT, WTO
亚太经济合作组织,中国,关贸总协,世界贸易组织
Narrative: A relevant document should contain information on China's economic growth; the importance of China in the development of economics and trade in the Asian-Pacific region; and China's efforts in resuming its status as a signatory state of GATT and a member nation of the WTO. An irrelevant document only mentions APEC when discussing bilateral trade relations with other nations but does not give details on why China wants to be a member of WTO.
相关文件应提到中国之经济成长;中国在亚太地区经济贸易发展的地位;中国为恢复关贸总协缔约国地位以及成为世界贸易组织成员国所做的努力.不相关文件为中国与外国代表讨论双边经贸关系提及亚太经济合作组织,但未谈具体方案者.

Number: CH18
Title: The Mid-East Peace Talks
中东和平会议
Description: Israel, Palestine, the Mid-East, peace talks
以色列,巴勒斯坦,中东,和平会议
Narrative: A relevant document should contain information on what the United States hopes to achieve in the Mid-East peace talks; how many countries participate in the peace talks; what is the agenda to be discussed; Arab nations positions toward Israel; and the Chinese view on the peace talks. A non-relevant document mentions the support of leaders of the Western nations and China for the Mid-East peace talks, but it does not contain information on the crux of the problem and how to solve it.

相关文件:应包括美国对中东和平会议的期望, 哪些国家出席中东和平会议,主要讨论的议题为何,阿拉伯国家对以阿冲突的态度,以及中国对整个中东问题的看法.不相关文件:如果文件只是西方首脑或中国领导人表示支持召开中东和平会议,但是未提到中东问题的症结和解决中东和平问题的的建议,则属不相关文件.

Number: CH19

Title: Project "Hope"

希望工程

Description: China, Project Hope, educational level, education

中国, 希望工程, 文化程度, 教育

Narrative: A relevant document should contain information on Project Hope's objectives and its results. Any document containing information on raising teachers pay, improving remote areas' education, educational reform laws, or the amount of private contributions to Project Hope is relevant. An irrelevant document mentions Project Hope but does not provide any concrete data on the success of the project such as how each area carries out the Project and how many people have benefited from it. Documents such as letters to the editor asking where to donate money for the Project are irrelevant. Documents that mention educational reform but do not give concrete measures are also irrelevant.

相关文件应提到希望工程是什么,它的目标为何,实施成果如何.有关改进教师待遇,文化扶贫工作与捐款等文件亦属相关文件.不相关文件包括听众信箱之问题,或文件提到教育法但未提具体法案内容,或仅提希望工程之名但没有具体数据以及推行办法者.

Number: CH20

Title: U.S. Military Personnel Missing in Action in Vietnam

越战失踪美军

Description: Vietnam, MIA's

越南,失踪美军

Narrative: A relevant document presents any information on U.S. soldiers missing in action in Vietnam. Document topics include missions to Vietnam, inter-

government cooperation and discussions, effect on lifting the trade embargo, the Vietnamese Government's reaction to U.S. statistics, MIA statistics, resolved cases, etc.
相关文件:应包括任何有关美国军人在越南失踪的信息,包括美军在越南的任务,美越政府间有关此问题的合作与讨论,以及美国停止对越南贸易制裁的影响.此外,越南政府对美国有关在越战中失踪军人的统计数字的反应与已经解决的案件等信息亦属相关文件.

Number: CH21
Title: The Role of the Governor of Hong Kong in the Reunification with the PRC
香港总督彭定康在香港回归中国一事上所扮演的角色
Description: Hong Kong issue, special administrative zone, Peng DingKang, plan, proposal
香港问题, 特别行政区, 彭定康, 计划, 建议
Narrative: A relevant document presents information on the role of the Governor of Hong Kong, Peng DingKang, in the reunification of China. Issues include any of the Governor's announcements, his official visits to China and meetings with Chinese officials, PRC criticism of Peng's legislative plans or proposals, etc. Non-relevant documents discuss any reactions to the Governor's actions or his politics in Hong Kong reunification from sources other than Hong Kong, UK, or the PRC.
相关文件:应包括香港总督彭定康在香港问题上所扮演的角色,包括所有彭定康发表过的声明,彭定康到中国访问与中国政府官员的谈话,以及中国政府对彭定康提出的有关香港立法改革的批评等. 不相关文件:任何非来自香港,英国,或中国的有关彭定康的评论皆属非相关文件.

Number: CH22
Title: The Spread of Malaria Infection in Various Parts of the World
世界各地感染疟疾的情况
Description: malaria, number of deaths, number of infections
疟疾, 死亡人数, 感染病例

Narrative: A relevant document presents numeric information about malaria infection or death rate at a national or international level. Non-relevant documents discuss health policies related to communicable diseases or vaccination against malaria without numeric information.
相关文件应包括有关世界各地感染疟疾的情况, 包括病例统计与死亡人数. 凡属讨论与传染性疾病有关的卫生政策或预防疟疾之疫苗接种而未提及感染或死亡人数的资料则为非相关文件.

Number: CH23
Title: Soviet Union's Mediation Role in the Gulf War
苏联在海湾战争中如何担任调停的角色
Description: Soviet Union, Gulf War, peace proposal, Iraq
苏联, 海湾战争, 和平建议, 伊拉克
Narrative: A relevant document discusses the Soviet Union's mediation in the Gulf War, including communication with Iraq, cease-fire resolution to the UN Security Council and their peace proposal for withdrawal of multi-national troops, etc.
相关文件应提及苏联在海湾战争中如何担任调停的角色,包括与伊拉克之间的沟通,苏联在联合国安理会中提出的停火协议以及要求多国部队从伊拉克撤出的和平建议

Number: CH24
Title: Reaction to Lifting the Arms Embargo for Bosnian Muslims
对取消向波黑穆斯林武器禁运的反应
Description: Bosnia-Herzogovenia, Muslims, arms embargo, United Nation's Security Council
波黑, 波斯尼亚-黑塞哥维那, 穆斯林, 武器禁运, 安理会, 联合国安理会
Narrative: A relevant document discuss international reaction to lifting the international arms embargo against the Former Yugoslavia. Document topics include statements in support or opposition by Government officials or officials of international organizations, pressure from U.S. legislative initiatives, etc.

相关文件应提及国际社会对取消向前南斯拉夫武器禁运的反应.文件内容应包括各国政府或国际组织官员对武器禁运所持的正反意见, 以及美国国会反对武器禁运而对联合国施加压力等.

Number: CH25

Title: China's Protection of Pandas

中国对熊猫的保护

Description: Ecoprotection, panda, nature preserve, endangered species

生态保护, 熊猫, 保护区, 濒临灭绝

Narrative: A relevant document discusses China's protection of pandas, such as how the Government sets up nature preserves for pandas, existing nature preserves, the nature preserve environment, the total number of pandas in China, or increases in the panda population. An irrelevant document covers panda sighting, without any details about protective measures, like how the Government is helping pandas to reproduce.

相关文件应提到中国对熊猫的保护, 比如中国政府如何设立熊猫的保护区, 目前熊猫的保护区包括那些地区; 熊猫的生态环境如何; 目前中国的熊猫总数大约有多少; 以及受到保护后熊猫数量的增长.不相关文件则包括新闻中只提到在某个地区看到熊猫,但是没有提出具体的保护方法,诸如政府如何设立保护区来帮助熊猫的繁殖.

Number: CH26

Title: Measures to Prevent Forest Fires in China

中国森林火灾的防范措施

Description: Mongolia, Manchuria (Northeast China) forest, fire, raging fires,

蒙古, 东北, 森林, 火灾, 大火

Narrative: A relevant document presents causes for forest fires in China, the area affected, acreage damaged, number injured and dead, or preventive measures adopted by the Chinese Government. Any document without the abovementioned information is not relevant.

相关文件应提及中国森林火灾发生的原因, 发生地区,受害面积,受伤与死亡人数,

以及政府采取什么样的防范措施. 如果没有上述的信息则属不相关文件.

Number: CH27

Title: Robotics Research in China

中国在机器人方面的研制

Description: robotics, automation

机器人, 自动控制

Narrative: A relevant document should have the following information: the functions of manufactured robots in China, the universities and institutes that are involved in robotic research, or direction of the research.

相关文件应提供下列的信息: 中国研制成功的机器人主要有什么功用, 有那些大学与研究机构参与机器人的研究设计, 研究的方向为何.

Number: CH28

Title: The Spread of Cellular Phones in China

移动电话在中国的成长

Description: digital, cellular, cellular phone, net, automatic roaming

数字,蜂窝式,移动电话,网络,自动漫游

Narrative: A relevant document contains the following kinds of information: the number of cellular phone users, area coverage, or how PSDN is implemented for national cellular communication. A non-relevant document includes reports on commercial manufacturers or brand name cellular phones.

相关文件应包括下列信息: 中国移动电话用户数, 覆盖地区, 中国如何以数据分组交换网覆盖全国移动电话的通讯. 不相关文件则包括有关制造移动电话厂商的报道, 以及移动电话的厂牌.

Number: CH29

Title: Building the Information Super Highway

信息高速公路的建设

Description: Information Super Highway, building

信息高速公路，建设

Narrative: A relevant document should discuss building the Information Super Highway, including any technical problems, problems with the information infrastructure, or plans for use of the Internet by developed or developing countries.
相关文件应提到信息高速公路的建设，包括任何技术上的，或与信息基础设施有关的问题，以及有关发达国家或发展中国家对国际网络的应用计划.

Number: CH30

Title: The Development of the Tourist Industry in China 1983-1993
中国旅游业的发展，１９８３　１９９３

Description: tourist agency, tourist industry, tourist, revenue, foreign exchange revenue
旅行社，旅游业，旅游客，收入，外汇收入，

Narrative: A relevant document should discuss the growth of the tourist industry in China and quantify that trade in terms of the total number of domestic and foreign tourists and revenue in any year betweeen 1983-1993. Moreover, it should compare the amount of foreign exchange revenue generated by foreign tourists in any year between 1983-1993. Any discussion of the construction of new hotels and service improvement in the Chinese tourist industry, such as providing information and making reservations through the computer network makes the document a relevant one.
相关文件应提及１９８３　１９９３中国旅游业的成长．包括国内外旅客的人数．营业收入，并且就１９８３　１９９３由海外旅客所创的外汇收入作比较．有关新旅馆的建设以及中国旅游服务的改进，如利用电脑网络订位以及提供旅游信息的文件亦属相关文件.

Number: CH31

Title: New U.S. Government policy concerning Cuban Refugees
美国政府对古巴难民的新政策

Description: Cuba, U.S., illegal immigrant, immigration policy, refugee
古巴，美国，非法移民，移民政策，难民

Narrative: A relevant document should discuss the new official U.S. Government policy toward Cuban immigration and Castro's reaction to the policy. Any document that contains statistics of legal and illegal Cuban immigrants in the United States, the differences between the 1960's and 1990's Cuban refugee waves, as well as foreign Government criticism of the Clinton administration's policy on the Cuban refugees is also relevant.
相关文件应提及克林顿政府针对大量古巴难民涌入美国所制定的新难民政策以及卡斯特罗对此政策之批评．有关在美的合法与非法古巴移民人数之统计，６０年代与９０年代古巴难民潮的不同，以及任何外国政府对克林顿政府新古巴难民政策批评的文件亦属相关文件．

Number: CH32
Title: Drug Traffikers in Latin America
拉丁美洲的贩毒集团
Description: Drug traffikers, Cali Cartel, Medina Cartel, Latin America, smuggling, drug selling, drug market, money laundering
贩毒集团，卡利贩毒集团，麦德林贩毒集团，拉丁美洲，走私，贩毒，毒品市场，洗钱
Narrative: A relevant document describe activities related to drug traffikers in Latin America, especially in Colombia, Panama, and Mexico. A document that discusses drug traffiers' activities of arms smuggling and overturning Governments is also relevant.
相关文件应提及贩毒集团在中南美洲（拉丁美洲）的贩毒活动，特别是在哥伦比亚．巴拿马，墨西哥．讨论贩毒集团走私武器与颠覆政府之活动亦属相关文件．

Number: CH33
Title: Airline hijackings between Taiwan and the Mainland
两岸劫机
Description: cross-strait hijackings, hijackers, Strait Exchange Foundation, Association for Relations Across the Strait

两岸，劫机，劫机犯，海基会，海协会

Narrative: A relevant document should describe some aspect of a specific airline hijacking from the Mainland to Taiwan, such as the hijackers motive, casualty or deaths during the hijacking, the sentencing of the hijackers. Discussions about the return of hijackers in the context of Taiwan-Mainland talks are not relevant unless a specific hijack event is described.

相关文件必须提到关于从大陆到台湾某一劫机事件的具体内容，诸如劫机动机，劫机过程中有无伤亡，及对劫机者之判刑．若文件只提及大陆与台湾对两岸劫机犯遣返问题之协商而非针对某一特定劫机事件之处理则属非相关文件．

Number: CH34

Title: The Impact of Droughts in China

旱灾在中国造成的影响

Description: drought, arrid region, relief assistance, food production, area, rainfall, China

旱灾，干旱地区，救灾款，粮食总产，面积，雨量，中国

Narrative: A relevant document should discuss the impact of droughts in China. Concrete indicators of impact include areas, number of people and acerage affected as well as total loss of crops and livestock. Any documents that discuss the Chinese Governmental relief assistance and measures of combating droughts are also relevant.

相关文件应提到旱灾在中国造成的影响，包括受灾地区，受害人数，受灾农地之面积，以及干旱对农作物与畜牧业所造成的损失．讨论政府帮助农牧民救灾的措施亦属相关文件．

Number: CH35

Title: Acts of Violence in South Africa Prior to the 27 April 1994 Presidential Election

一九九四年四月二十七日南非总统大选前之暴力事件

Description: violent events, violent conflict, apartheid, South Africa, April 27, General Election, riot area, massacre, Mandela

暴力事件，暴力冲突，种族隔离，南非，四月二十七日，全民大选，暴乱地区，屠杀，曼德拉，

Narrative: A relevant document should discuss the violence, the causes and areas affected in South Africa prior to the April 27 1994 South African Presidential election. Any document that discusses groups participating in the riots and South African Government's efforts to quell riots is also relevant.

相关文件应提到1 9 9 4 年4 月2 7日南非总统大选之前各地所发生的暴力事件以及暴力事件发生的原因与地区．提及参与暴力事件的团体及南非政府对平息暴力所作的努力的文件亦为相关文件．

Number: CH36

Title: The Growth of China's Foreign Trade

中国对外贸易的成长

Description: foreign trade, exports, imports, total amount of trade, foreign exchange, foreign funds, international market, export product, import product, China

对外经贸, 出口, 进口, 进出总额，外汇，外贸，国际市场，出口商品，进口商品，中国

Narrative: A relevant document should discuss: (1) China's foreign trade policy, (2) total amount of trade, (3) export products, (4) import products, (5) China's competitiveness in the international markets, (6) China's trade relationship with Taiwan and Hong Kong, (7) growth of China's foreign trade in percentage, or (8) major export and import countries.

相关文件应提到中国外贸政策，进出口总额，出口商品，进口商品，中国在国际市场上的竞争力，与台湾香港的贸易关系，对外贸易成长的百分比，以及主要输出国与输入国．

Number: CH37

Title: The Collapse of the Bubble Economy in Japan

日本泡沫经济的破灭

Description: bubble economy, collapse, recession, economic downturn, economic

recovery
泡沫经济，破灭，不景气，经济衰退，经济复苏
Narrative: A relevant document should discuss the economic recession in Japan after the collapse of the bubble economy, especially in the areas of finance, real estate, and industry, and the Japanese government's policy to stimulate economy recovery. Discussions of the predictions of Japanese economic growth are also relevant.
相关文件应提到自泡沫经济破灭後，日本所经历的经济不景气,特别是金融，房地产业与企业的萧条，以及日本政府为刺激经济复苏所采取的政策．对日本经济成长的预测亦属相关文件．

Number: CH38
Title: Protection of Wildlife in China
中国野生动物保护形势
Description: Protection of Wildlife, Legislation Protecting Wildlife, Associations for the Protection of Wildlife, rare and precious animals, endangered species
野生动物保护，《野生动物保护法》，野生动物保护协会，珍稀动物，濒危动物
Narrative: A relevant document should discuss protection of endangered species in China. Relevant documents include the following information: (1) "Legislation protecting endangered species", (2) rare and precious animals, (3) hunting and selling wild animals, (4) adopting measures to rescue rare animals, (5) market survelliance work, or (6) establishing preservation grounds for endangered species.
相关文件应提到中国野生动物保护形势．相关文件包括下列信息:（一）《野生动物保护法》，（二）珍稀动物，（三）捕猎和销售野生动物，（四）采取措施抢救珍稀动物，（五）市场管制工作，或（六）建设濒危动物的保护区．

Number: CH39
Title: Terrorism in Algeria
阿尔及利亚的恐怖主义

Description: Algeria, terrorism, curfew, assasination, opposition party, state of emergency
阿尔及利亚，恐怖主义，宵禁，暗杀，反对党，紧急状态
Narrative: A relevant document should discuss terrorist activity in Algeria, including the Algerian authorities measures against terrorism, discussions with the opposition party, or violent terrorist activity.
相关文件必须提到在阿尔及利亚发生的恐怖活动，此活动包括阿尔及利亚当局对恐怖主义采取的措施,与反对党领袖举行会谈，或者恐怖暴力活动.

Number: CH40
Title: Provincial effective measures to Lighten the Burden for Peasants
某些省采取有效措施减轻农民负担
Description: the burden for peasants, three turmoils, fee collection, lighten, province
农民负担，三乱，收费，减轻，省
Narrative: A relevant document should discuss effective policies to lighten the burdent for peasants in specific provinces. Relevant documents include discussion of changes in taxation policy, indications of reduction of tax burden, or supervision to reduce indiscriminate taxing. If a document merely describes the problem, or reasons for the increased burden, it is not relevant.
相关文件必须提到某些省落实减轻农民负担的切实有效措施，诸如税收政策的改进减轻向农民税收的情形或监督减轻非法收费的执行．如果文件只论及农民负担的问题或增加农民负担的原因，则属非相关文件.

Number: CH41
Title: Bridge and Tunnel Construction for the Beijing-Kowloon Railroad
京九铁路的桥梁隧道工程
Description: Beijing-Kowloon Railroad, bridge, tunnel, connection, very large bridge
京九铁路，桥梁，隧道，贯通，特大桥，
Narrative: A relevant document discusses bridge and tunnel construction for the

Beijing-Kowloon Railroad, including location, construction status, span or length.
相关文件必须提到京九铁路的桥梁隧道工程，包括地点、施工阶段、长度.

Number: CH42
Title: Dikes and Reservoirs in Flood Prevention in the Seven Great Rivers
七大江河的防洪水库和大堤
Description: Flood prevention on the Seven Great Rivers, Yangtze River, Yellow River, Huaihe River, Haihe River, Pearl River, Liaohe River, Songhua River, flood control, reservoir, dike, embankment
长江、黄河、淮河、海河、珠江、辽河、松花江等七大江河的防洪,防汛，水库，堤，坝，
Narrative: A relevant document should discuss specific dikes and reservoirs in the Seven Great Rivers region. Relevant documents discuss the following information: (1) construction projects, (2)measures for flood and rescue work, (3) reservoir water levels, or (4) flood discharging. Documents discussing the Three Gorge Project are non-relevant.
相关文件必须提到七大江河地区的某一些水库与防堤．相关文件应包括下列信息：（一）建设项目；（二）抗洪抢险（三）水库水位以及（四）开闸泄洪．凡三峡工程则属非相关文件.

Number: CH43
Title: The Fourteenth Dali Lama
十四世达赖喇嘛
Description: The fourteenth Dali Lama, Tibet
十四世达赖喇嘛，西藏
Narrative: A relevant document should discuss the life the Dali Lama, his activities, or his stand on Tibetan indepedence. Articles on the position of the Chinese Government toward the Dali Lama are relevant. Documents discussing the position of other Governments toward the Dali Lama are also relevant.
相关文件必须提到十四世达赖喇嘛的生活或活动以及其对西藏独立的立场．任何有关中央政府对达赖喇嘛的立场之文件皆属相关文件．讨论到其他政府对达

赖喇嘛的立场的文件亦属相关文件.

Number: CH44

Title: The Three Gorges Project and Resettlement

三峡工程与移民

Description: The Three Gorges Project, resettlement

三峡工程，移民，

Narrative: A relevant document should discuss the resettlement plan, the implementation, and reaction of the resettlement population. Non-relevant documents discuss the environmental and cultural impact.

相关文件必须提到移民政策，如何执行及移民反应．任何环境及文化的负面影响则属非相关文件.

Number: CH45

Title: China Red Cross

中国红十字会

Description: China Red Cross, providing relief goods and materials, aiding, donating, relieving (disaster victims)

中国红十字会，救济物资，援助，捐款，赈济，

Narrative: A relevant document should discuss the activities of the China Red Cross，including the type of aid and the recipient. For relevant documents in which the China Red Cross is an intermediary, the document should describe the role or function that the China Red Cross is performing and the beneficiary of the activity.

相关文件必须提到中国红十字会的各种活动包括援助及受援的种类，若文件提到中国红十字会作为中间人的活动应描述其所扮演的角色及其功能，以及此活动的贡献.

Number: CH46

Title: New advances in the Relationship between China and Vietnam

中越两国关系的新进展

Description: Sino-Vietnamese relations, Vietnam, normalization, economic cooperation, nongovernmental border and port trade, exchanges, agreements
中越关系，越南，正常化，经济合作，非官方边境贸易，交流，协议

Narrative: A relevant document should discuss new advances in the Sino-Vietnamese relationship after normalization. Relevant documents should identify border trade; basic agreements reached between the two countries; exchanges and cooperation regarding economy and trade, science and technology, or culture and education; or resolution of the Campuchea problem.
相关文件应提到中越两国关系正常化后的新进展．文件须提到非官方边境贸易，原则性协议的达成，经贸、科技、文教等领域的交流与合作，以及柬埔寨问题的解决．

Number: CH47

Title: The Impact of the 1991 Mount Pinatubo Volcano
１９９１年菲律宾皮纳图博火山爆发造成的后果

Description: Philippines, Mount Minatubo, volcanic ash, magma, eruption
菲律宾，皮纳图博火山，火山灰，岩浆，爆发

Narrative: A relevant document should discuss the following kinds of information: weather in the Northern Hemisphere; evacuation of citizens; casualties, deaths, and losses resulting from the eruption; damage to U.S. Subic Bay Naval base and Clark Air Force base; or damage to the ozone layer.
相关文件应提到以下信息:北半球的气侯，火山周围居民的撤离，火山爆发造成的伤亡和损失，对美国苏比克海军基地与克拉克空军基地的损害和臭氧层的破坏．

Number: CH48

Title: Kuwaiti Oil Industry after the Gulf War
海湾战争之后的科威特石油业

Description: Kuwait, Gulf War, oil well, oil production, oil industry
科威特，海湾战争，油井，石油生产，石油业，

Narrative: A relevant document should discuss the economic losses and recovery of the Kuwaiti oil industry after the Gulf War. Economic losses include the number of burning oil fields, the efforts to extinguish fires, and the Chinese firefighters work. The recovery of oil production includes post-war rebuilding such as construction and contracts with various countries.
相关文件应提到海湾战争对科威特石油业所造成的经济损失与科威特如何恢复石油生产两大方面．经济损失方面包括燃烧油井的数量，灭火工作的进行以及中国参加灭火工作的情形．恢复石油生产方面须提及战后重建工作包括各种建设工程，与各国签订合同等．

Number: CH49
Title: China's position on nuclear disarmament
中国对核裁军立场
Description: China, nuclear tests, nuclear disarmament, destruction of nuclear weapons, Non-Proliferation Treaty, START treaty
中国，核试验，核裁军，销毁核武器，《不扩散核武器条约》，《削减战略武器条约》
Narrative: A relevant document should discuss China's position on nuclear disarmament, including how China fulfills its commitment to non-proliferation, how China is developing its own nuclear program and underground nuclear tests, or how China is not helping non-nuclear countries to develop nuclear weapons but promotes international peaceful use of nuclear power. If a document discusses the extension of the non-proliferation treaty or China's approval of a country's becoming a treaty member, it is relevant. Non-relevant documents discuss the START treaty.
相关文件应提到中国对核裁军的立场，包括中国如何履行不扩散核武器的义务；中国如何发展核武器与地下核试验以及中国如何不帮助无核国家发展核武器而促进国际核能的和平利用．若文件提及《不扩散核武器条约》的延长问题或中国对赞成别国加入《不扩散核武器条约》亦属相关文件．提及《削减战略武器条约》的文件则属非相关文件．

Number: CH50

Title: China and Britain Reach an Understanding regarding the New Airport in Hong Kong

关于中国与英国政府在香港新机场问题上所达成的谅解

Description: China, Britain, new airport, construction

中国，英国，新机场，建设

Narrative: A relevant document should discuss the setting in which the problem of the airport arose, why the Chinese opposed the construction of the the new airport, or the contents of the memorandum of agreement regarding issues related to the construction of the new airport in Hong Kong.

相关文件应提到新机场问题产生的背景，中方为何反对香港新机场的建设，以及中英《关于香港新机场建设及有关问题的谅解备忘录》之内容为何.

Number: CH51

Title: China's Policy of Protecting the Environment

中国对保护环境的政策

Description: China, environment, protection, acid rain, air pollution, water pollution, air pollution, economy

中国，环境，保护，酸雨，大气污染，水污染，空气污染，经济

Narrative: A relevant document should discuss China's policy toward environmental protectionism. Relevant documents include the following information: (1) the reasons for and extent of the pollution, (2) the relationship between economic growth and environmental pollution, or (3) the legislation and policies formaulated by the Chinese Government. Non-relevant documents discuss global environmental problems or problems with environmental pollution in other countries.

相关文件应提到中国对环境保护的政策．相关文件应包括下列信息:（一）造成环境污染的因素及其对环境危害的程度，（二）经济成长与环境污染之间的相关性，或（三）中国政府对环境所制定的立法与政策．若文件提及世界性的环境问题或中国以外的环境污染问题则属非相关文件.

Number: CH52

Title: Reform and Growth in China's Real Estate Industry
中国房地产业的改革与发展

Description: China, real estate industry, investment, scale, trade, transferring possession, wild selling, huge profits
中国，房地产业，投资，规模，交易，转让，炒卖，暴利，

Narrative: A relevant document should discuss problems faced in the real estate industry and the various measures adopted by the Government to promote healthy growth for the industry. Problems in the real estate industry include wild buying and selling of real estate, developing at an excessive scale, investors obtaining excessive profits, excessive and indiscriminate selling of public lands, unrestricted trade practices, trade price speculation, etc. Growth policies being adopted by the Government for macro-management of the industry include measures to promote the healthy growth in the industry by collecting a value added tax on land, implementing controls for land use, promulgating the urban control of land use law, etc. Documents about the reform of the housing system are also relevant.
相关文件应提到中国房地产业所面临的问题以及政府采取何种措施来促进房地产业的健康发展．房地产业的问题包括炒买炒卖房地产，开发规模过大，投资者获取超额的利润，国有土地出让过多过滥，交易行为不规矩，交易价格混乱等．政府对房地产业所采取宏观管理则包括开征土地增值税，实施土地使用管制，颁布城市房地产业管理法等来促进房地产业健康发展之措施．有关中国住房制度改革之文件亦属相关文件．

Number: CH53

Title: The Development of the Chinese Auto Industry, and the Chinese Auto Market
中国汽车工业的发展与市场

Description: China, production, manufacture, auto, auto industry, auto market
中国，生产，制造，汽车，汽车工业，汽车市场

Narrative: A relevant document should discuss the Chinese Government's plan

to develop the auto industry, including how to attract foreign investment and technology, or how to plan for production of vehicle types and annual output as well as for the demand in the domestic auto market. Documents which discuss policies formulated to protect the Chinese auto industry are also relevant.
相关文件应提到中国政府对发展汽车工业之计划，包括如何吸引外资与技术，计划生产车辆类型与年产量以及中国国内市场对汽车需求．有关中国政府为保护本国汽车工业发展所制定的政策亦属相关文件．

Number: CH54
Title: China's Reaction to U.S. Sale of F-16 Fighters to Taiwan
中国关于美国政府向台湾出售Ｆ－１６战斗机的反应
Description: China, U.S., Taiwan, F-16 fighter, sale
中国，美国，台湾，Ｆ　１６战斗机，出售
Narrative: A relevant document should discuss the resolution concerning U.S. weapon sales to Taiwan in the Sino-American "8-17" Joint Communique and why the Chinese consider President Bush's decision to sell F-16 fighters to Taiwan to be in violation of the spirit of the Sino-American "8-17" Joint Communique and to be damaging to Sino-American relations.
相关文件应提到中美“八　一七”联合公报中对美国向台湾出售武器之决定，以及为何中国认为布什总统决定售予台湾Ｆ　１６战斗机是违反中美“八　一七”联合公报之精神并损害中美关系．

BIBLIOGRAPHY

Abney, S. (1991), Parsing by Chunks, *in* R. Berwick, S. Abney & C. Tenny, eds, 'Principle-Based Parsing', Kluwer.

Al-Mubaid, H. & Nagula, S. (2005), Machine learning approach for context-sensitive error detection, *in* 'Proceedings of International Conference on Intelligent Computing and Information Systems (ICICIS05)'.

Bahl, L. R., Brown, P. F., de Souza, P. V. & Mercer, R. L. (1989), 'A tree-based statistical language model for natural language speech recognition', *IEEE Transactions on Acoustics, Speech, and Signal Processing* **37**, 1001–1008.

Bahl, L. R., Jelinek, F. & Mercer, R. L. (1983), 'A maximum likelihood approach to continuous speech recognition', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **5**(2), 179–190.

Bell, T., Cleary, J. & Witten, I. (1990), *Text Compression*, Prentice Hall.

Bellagarda, J. (2004), 'Statistical language model adaptation: review and perspectives', *Speech Communication* **42**, 93–108.

Bellegarda, J. (2000), 'Exploiting latent semantic information in statistical language modeling', *Proceedings of the IEEE* **88**(8), 1279–1296.

Bengio, Y.and Ducharme, R. & Vincent, P. (2001), 'A neural probabilistic language model', *Advances in Neural Information Processing Systems (NIPS)* **13**.

Bergsma, S., Lin, D. & Goebel, R. (2009), Web-scale n-gram models for lexical disambiguation, *in* 'Proceedings of the 21st international joint conference

on Artificial intelligence', Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 1507–1512.

Bouchaffra, D. (2005), 'Probabilistic logic with minimum perplexity: Application to language modeling', *Pattern Recognition* **38**, 1307–1315.

Brill, E. (1993), A Corpus-Based approach to language learning, Phd thesis, University of Pennsylvania.

Brown, P., Pietra, V. D., deSouza, P., Lai, J. & Mercer, R. (1992), 'Class-based n-gram models of natural language', *Computational Linguistics* **18**(4).

Buckley, C. (1985), Implementation of the smart information retrieval system, Technical Report 85-686, Cornell University.

Buckley, C., Singhal, A. & Mitra, M. (1996), Using query zoning and correlation within smart: Trec 5, *in* 'Proceedings of the Fifth Text REtrieval Conference (TREC 5)'.

Buckleym, C., Walz, J., Mitra, M., & Cardie, C. (1997), Using clustering and superconcepts within smart: Trec 6, *in* 'Proceedings of Sixth Text REtrieval Conference (TREC 6)'.

Chang, C.-H. & Chen, C.-D. (1994), A study on corpus-based classification of chinese words, *in* 'Proceedings of International Conference on Chinese Computing (ICCC '04)', Singapore.

Chang, J., Lin, Y. & Su, K. (1995), Automatic construction of a chinese electronic dictionary, *in* 'Proceedings of the Third Workshop on Very Large Corpora'.

Chao, Y. R. (1968), *A Grammar of Spoken Chinese*, Berkeley: University of California Press.

Charniak, E. (2001), Immediate head parsing for language models, *in* 'Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL)'.

Chelba, C. & Jelinek, F. (1998), Exploiting syntactic structure for language modelling, *in* 'Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics(ACL/COLING)', Montreal, pp. 225–231.

Chen, A. (2003), Chinese word segmentation using minimal linguistic knowledge, *in* 'Proceedings of the Second SIGHAN Workshop on Chinese Language Processing', pp. 148–151.

Chen, A., He, J., Xu, L., Gey, F. C. & Meggs, J. (1997), 'Chinese text retrieval without using a dictionary', *Sigir Forum* **31**, 42–49.

Chen, K. & Chen, H. (2001), 'Cross-language chinese text retrieval in ntcir workshop: towards cross-language multilingual text retrieval', *ACM SIGIR Forum* **35**(2), pp. 12–19.

Chen, S. & Goodman, J. (1999), 'An empirical study of smoothing techniques for language modeling', *Computer Speech and Language* **13**.

Chien, L.-F. (1997), Pat-tree-based keyword extraction for chinese information retrieval, *in* 'Proceedings of the 20th annual international ACM SIGIR conference on research and development in information retrieval', SIGIR '97, pp. 50–58.

Clark, A. (2001), Unsupervised Language Acquisition: Theory and Practice, PhD thesis, University of Sussex.

Clark, A. S. (2002), 'Unsupervised language acquisition: Theory and practice', *Computing Research Repository* **cs.CL/0212**.

Clarkson, P. (1999), Adaptation of Statistical Language Models for Automatic Speech Recognition, PhD thesis, Cambridge University Engineering Department.

Cleary, J. & Witten, I. (1984), 'Data compression using adaptive coding and partial string matching', *IEEE transactions on Communications* **32**(4).

Collins, M. (2000), Discriminative reranking for natural language parsing, *in* 'International Conference on Machine Learning', pp. 175–182.

Collins, M. (2002), Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms, *in* 'Conference on Empirical Methods in Natural Language Processing', pp. 1–8.

Cooper, W. S., Chen, A. & Gey, F. C. (1994), Full text retrieval based on probabilistic equations with coefficients fitted by logistic regression, *in* 'Proceedings of the Second Text REtrieval Conference (TREC 2)'.

Cucerzan, S. & Brill, E. (2004), Spelling correction as an iterative process that exploits the collective knowledge of web users, *in* 'Proceedings of Conference on Empirical Methods on Natural Language Processing', pp. 293–300.

Dai, Y., Kgoo, C. & Loh, T. (1999), A new statistical formula for chinese text segmentation incorporating contextual information, *in* 'SIGIR'99', Berkley.

de Marcken, C. (1996), Unsupervised Language Acquisition, PhD thesis, MIT.

Dong, Z., Dong, Q. & Hao, C. (2010), Word segmentation needs change - from a linguist's view, *in* 'Proceedings of CIPS-SIGHAN Joint Conference on Chinese Language Processing'.

Du, L., Zhang, Y., Sun, L., Sun, Y. & Han, J. (2000), Pm-based indexing for chinese text retrieval, *in* 'Proceedings of Fifth International Workshop on Information Retrieval with Asian Languages', pp. 55–59.

Emerson, T. (2005), The second international chinese word segmentation bakeoff, *in* 'Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing'.

Entwisle, J. (1997), An Investigative Parser for English Using Constraints on Surface Sentence-Form, PhD thesis, The Flinders University of South Australia.

Feng, H., Chen, K., Kit, C. & Deng, X. (2004), Unsupervised segmentation of chinese corpus using accessor variety, *in* 'the First International Joint Conference on Natural Language Processing (IJCNLP '2004)', pp. 694–703.

Finch, S. P. (1993), Finding structure in language, Phd thesis, University of Edinburgh.

Foo, S. & Li, H. (2004), 'Chinese word segmentation and its effect on information retrieval', *Information Processing and Management* **40**, 161–190.

Fox, C. (1992), *Information Retrieval: Data Structures and Algorithms*, Prentice Hall, chapter Lexical analysis and stoplists, pp. 102–130.

Franz, M., McCarley, J. S. & Zhu, W. J. (2000), English chinese information retrieval at ibm, *in* 'Proceedings of the Nineth Text REtrieval Conference (TREC 9)'.

Fu, G. & Wang, X. (1999), Unsupervised chinese word segmentation and unkown word identification, *in* 'Proceedings of the Fifth Natural Language Processing Pacific Rim Symposium (NLPPRS)', Beijing, China.

Gale, W. A., Church, K. W. & Yarowsky, D. (1994), Discrimination decisions for 100,000 dimensional spaces, *in* 'Current Issues in Computational Linguistics: In Honour of Don Walker', Kluwer Academic Publishers, pp. 429–450.

Gao, J., Goodman, J., Li, M. & Lee, K. F. (2002), 'Toward a unified approach to statistical language modeling for chinese', *ACM transaction on Asian Language information processing* **1**(1).

Gao, J., Li, M. & Huang, C.-N. (2003), Improved source-channel models for chinese word segmentation, *in* 'Proceedings of the 41st Annual Meeting on Association for Computational Linguistics', ACL '03, pp. 272–279.

Gao, J., Li, M., Wu, A. & Huang, C.-N. (2005), 'Chinese word segmentation and named entity recognition: A pragmatic approach', *Computational Linguistics* **31**(4), 531–574.

Gao, J., Li, X., Micol, D., Quirk, C. & Sun, X. (2010), A large scale ranker-based system for search query spelling correction, *in* 'Proceedings of the 23rd International Conference on Computational Linguistics', COLING '10, pp. 358–366.

Gao, J., Suzuki, H. & Yuan, W. (2006), 'An empirical study on language model adaptation', *ACM Transactions on Asian Language Information Processing (TALIP)* **5**(3), 209–227.

Gao, J., Wu, A., Li, M., Huang, C.-N., Li, H., Xia, X. & Qin, H. (2004), Adaptive chinese word segmentation, *in* 'Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics', ACL '04, Association for Computational Linguistics, Stroudsburg, PA, USA.

Gao, J., Xun, E., Zhou, M., Huang, H., Nie, J. Y., Zhang, J. Y. & Su, Y. (2000), Trec 9 clir experiments at msrcn, *in* 'Proceedings of the Nineth Text REtrieval Conference (TREC 9)'.

Gao, L., Zhang, Y., Liu, T. & Liu, G. (2006), Word sense language model for information retrieval, *in* 'Proceedings of Asia Information Retrieval Symposium', pp. 158–171.

GB/T13715-92 (1993), Contemporary chinese language word-segmentation specification for information processing, Technical report, State Technology Supervision Administration, China.

Ge, X., Pratt, W. & Smyth, P. (1999), Discovering chinese words from unsegmented text, *in* 'SIGIR'99', Berkeley.

Golding, A. R. (1995), A bayesian hybrid method for context-sensitive spelling correction, *in* 'Proceedings of the third Workshop on Very Large Corpora', Boston, pp. 39–53.

Golding, A. R. & Roth, D. (1996), Applying winnow to context-sensitive spelling correction, *in* 'Proceedings of 13th International Conference on Machine Learning', San Francisco, pp. 182–190.

Golding, A. R. & Schabes, Y. (1996), 'Combining trigram-based and feature-based methods for context-sensitive spelling correction', *Computing Research Repository* **cmp-lg/960**, 71–78.

Good, I. J. (1953), 'The population frequencies of species and the estimation of population parameters', *Biometrika* **40**, 237–264.

Goodman, J. (2001), A bit of progress in language modeling, Technical Report MSR-TR-2001-72, Microsoft Research.

Gu, H. Y., Tseng, C. Y. & Lee, L. S. (1991), 'Markov modeling of mandarin chinese for decoding the phonetic sequence into chinese characters', *Computer Speech and Language* **5**.

Guidelines (2005), Segmentation guidelines, Technical report, Language Information Sciences Research Centre, City University of Hong Kong.

Gusfield, D. (1997), *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*, Cambridge University Press, Cambridge.

Hao, L. & Hao, L. (2008), 'Automatic identification of stop words in chinese text classification', *IEEE Control Systems Magazine* pp. 718–722.

Harris, Z. (1951), *Structural linguistics*, Chicago: University of Chicago Press.

Harris, Z. (1955), 'From phoneme to morpheme', *Language* **31**(2).

He, J., Xu, J., Chen, A., Meggs, J. & Gey, F. (1996), Berkeley chinese information retrieval at trec 5: Technical report, *in* 'Proceedings of the Sixth Text REtrieval Conference (TREC 5)'.

Hofmann, T. (1997), Probabilistic latent semantic indexing, *in* 'Proceedings of the 22th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval'.

Howard, P. (1993), The design and analysis of efficient lossless data compression systems, PhD thesis, Brown University, Providence, Rhode Island.

Hsieh, M., Lo, T. & Lin, C. (1989), A grammatical approach to converting phonetic symbols into characters, *in* 'Proceedings of National Computer Symposium', Taipei, Taiwan.

Huang, C. N. & Zhao, H. (2007), 'Chinese word segmentation: A decade review', *Journal of Chinese Information Processing* **21**(3).

Huang, C.-R., Chan, K.-j., Chang, L. & Chen, F.-y. (1997), 'Segmentation standard for chinese natural language processing', *International Journal of Computational Linguistics and Chinese Language Processing* **2**(2), 47–62.

Huang, J. H. & Powers, D. (2001), Large scale experiments on correction of confused words, *in* 'Proceedings of the 24th Australasian conference on Computer science', pp. 77–82.

Huang, J. H. & Powers, D. (2002), Unsupervised chinese word segmentation and classification, *in* 'First Student Workshop in Computational Linguistics', Beijing, China.

Huang, J. H. & Powers, D. (2003), Chinese word segmentation based on contextual entropy, *in* '17th Pacific Asia Conference on Language, Information and Computation', Singapore.

Huang, J. H. & Powers, D. (2004), Adaptive compression-based approach for chinese pinyin input, *in* 'Third ACL SIGHAN Workshop on Chinese Processing'.

Huang, J. H. & Powers, D. (2008), Suffix tree based approach for chinese information retrieval, *in* 'International Conference on Intelligent Systems Design and Applications (ISDA)', pp. 393–397.

Huang, J. H. & Powers, D. (2011), Error-driven adaptive language modeling for pinyin-to-character conversion, *in* 'International Conference on Asian Language Processing (IALP2011)', Penang, Malaysia.

Huang, X. & Roberton, S. (1997), Okapi chinese text retrieval experiments at trec 6, *in* 'Proceedings of Sixth Text REtrieval Conference (TREC 6)'.

Huang, X. & Robertson, S. (2000), A probabilistic approach to chinese information retrieval: Theory and experiments, *in* 'Proceedings of 22nd Annual Colloquium on Information Retrieval Research'.

Hutchens, J. L. & Alder, M. D. (1998), Finding structure via compression, *in* 'Proceedings of the Joint Conferences on New Methods in Language Processing and Computational Natural Language Learning', NeMLaP3/CoNLL '98, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 79–82.

Islam, A. & Inkpen, D. (2009), Real-word spelling correction using google web 1t 3-grams, *in* 'Proceedings of the 2009 Conference on Empirical Methods on Natural Language Processing', pp. 1241–1249.

Iyer, R. & Ostendorf, M. (1999), 'Modeling long distance dependence in language: Topic mixtures versus dynamic cache models', *IEEE Transactions on Speech and Audio Processing* **7**(1), 30–39.

J., G., Yu, H., Yuan, W. & Xu, P. (2005), Minimum sample risk methods for language modeling, *in* 'Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)', pp. 209–216.

Jelinek, F. & Mercer, R. (1980), Interpolated estimation of markov source parameters from sparse data, *in* 'International Conference on Pattern Recognition'.

Jiang, W., Guan, Y. & Wang, X.-L. (2006), An improved unknown word recognition model based on multi-knowledge source method, *in* 'Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications', ISDA '06, pp. 825–832.

Jin, G. & Chen, X. (2008), The fourth international chinese language processing bakeoff: Chinese word segmentation, named entity recognition and chinese pos tagging, *in* 'Proceedings of the Sixth SIGHAN Workshop on Chinese Language Processing'.

Jin, Z. & Tanaka-Ishii, K. (2006), Unsupervised segmentation of chinese text by use of branching entropy, *in* 'Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING/ACL '06)'.

Katz, S. M. (1987), 'Estimation of probabilities from sparse data for the language model component of a speech recognizer', *IEEE Transactions on Acoustics, Speech, and Signal Processing* **35**, 400–401.

Kempe, A. (1999), Experiments in unsupervised entropy-based corpus segmentation, *in* 'Ninth Conference of the European Chapter of the Association for Computational Linguistics' 99 Workshop', Bergen, Norway.

Khudanpur, S. & Wu, J. (2000), 'Maximum entropy techniques for exploiting syntactic, semantic and collocational dependencies in language modeling', *Computer Speech and Language* **14**, 355–372.

Kilgaroff, A. (1996), Which words are particularly characteristic of a text?, A survey of statistical approaches, itri technical report, University of Brighton.

Kneser, R. & Ney, H. (1995), Improved backing-off for m-gram language modeling, *in* 'IEEE International Conference on Acoustics, Speech, and Signal Processing', Vol. 1, pp. 181–184.

Krovetz, R. & Croft, W. B. (1992), 'Lexical ambiguity and information retrieval', *ACM Transactions on Information Systems* **10**, 115–141.

Kuhn, R. & De Mori, R. (1990), 'A cache-based natural language model for speech reproduction', *IEEE Transaction on Pattern Analysis and Machine Intelligence* .

Kukich, K. (1992), 'Techniques for automatically correcting words in text', *ACM computing survey* **24**(4), 377–439.

Kuo, J. J. (1995), 'Phonetic-input-to-character conversion system for chinese using syntactic connection table and semantic distance', *Computer Processing and Oriental Languages* **10**(2), 195–210.

Kwok, K. L. (1996), 'A network approach to probabilistic information retrieval', *ACM Transactions on Information Systems* **13**, 325–353.

Kwok, K. L. (1997), Lexicon effects on chinese information retrieval, *in* 'Proceedings of Second Conference on Empirical Methods in Natural Language Processing (EMNLP'.

Kwok, K. L. (1999), 'Employing multiple representations for chinese information retrieval', *Journal of the American Society for Information Science (JASIS)* **50**(8), 709–723.

Lafferty, J. D., McCallum, A. & Pereira, F. C. N. (2001), Conditional random fields: Probabilistic models for segmenting and labeling sequence data, *in* 'Proceedings of the Eighteenth International Conference on Machine Learning'.

Lafferty, J., Sleator, D. & Temperley, D. (1992), Grammatical trigrams: A probabilistic model of link grammar, *in* 'Proceeding of the AAAI Fall Symposium on Probabilistic Approaches to Natural Language', Cambridge, MA.

Levow, G.-A. (2006), The third international chinese language processing bakeoff: Word segmentation and named entity recognition, *in* 'Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing'.

Li, H., Huang, C., Gao, J. & Fan, X. (2004), The use of svm for chinese new word identification, *in* 'International Joint Conference on Natural Language Processing', pp. 723–732.

Li, L., Xuan, W., Wang, X. L. & Yu, Y. B. (2009), 'A conditional random fields approach to chinese pingyin-to-character conversion', *Jounal of Communication and Computer* **6**(4), 25–31.

Li, M., Gao, J., Huang, C.-N. & Li, J. (2003), Unsupervised training for overlapping ambiguity resolution in chinese word segmentation, *in* 'Proceedings of the Second SIGHAN Workshop on Chinese Language Processing', Association for Computational Linguistics, Sapporo, Japan, pp. 1–7.

Li, M., Zhu, M., Zhang, Y. & Zhou, M. (2006), Exploring distributional similarity based models for query spelling correction, *in* 'Meeting of the Association for Computational Linguistics'.

Li, Y., Miao, C., Bontcheva, K. & Cunningham, H. (2005), Perceptron learning for chinese word segmentation, *in* 'Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing'.

Lidstone, G. J. (1920), 'Note on the general case of the bayes-laplace formula for inductive or a posteriori probabilities', *Transactions of the Faculty of Actuaries* **8**, 182–192.

Liu, L. (2005), A Corpus-based Approach to the Chinese Word Segmentation, PhD thesis, Ludwig Maximilian University of Munich.

Low, J. K., Ng, H. T. & Guo, W. (2005), A maximum entropy approach to chinese word segmentation, *in* 'Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing'.

Lua, K. & Gan, K. (1994), 'An application of information theory in chinese word segmentation', *Computer Processing of Chinese and Oriental Languages* **8**.

Luk, R. & Kwok, K. (2002), 'A comparison of chinese document indexing strategies and retrieval models', *ACM Transactions on Asian Language Information Processing* **1**(3), 225–268.

Luo, X., Sun, M. & Tsou, B. K. (2002), Covering ambiguity resolution in chinese word segmentation based on contextual information, *in* 'Proceedings of the 19th international conference on Computational linguistics - Volume 1', COLING '02, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 1–7.

Magerman, D. & Marcus, M. (1990), Parsing a natural language using mutual information statistics, *in* 'Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI 90)'.

Manber, U. & Myers, G. (1990), Suffix arrays: a new method for on-line string searches, *in* 'Symposium on Discrete Algorithms', pp. 319–327.

Mangu, L. & Brill, E. (1997), Automatic rule acquisition for spelling correction, *in* M. Kaufmann, ed., 'Proceedings of International Conference on Machine Learning'.

Manning, C. D., Raghavan, P. & Schutze, H. (2008), *Introduction to Information Retrieval*, Cambridge University Press.

Mays, E., Damerau, F. & Merser, R. (1991), 'Context based spelling correction', *Information Processing and Management* **27**(5), 517–522.

McCreight, E. M. (1976), 'A space-economical suffix tree construction algorithm', *Journal of The ACM* **23**, 262–272.

McNamee, P., Mayfield, J. & Piatko, C. (2000), The haircut system at trec 9, *in* 'Proceedings of the Nineth Text REtrieval Conference (TREC 9)'.

Moffat, A. (1990), 'Implement the ppm data com-pression scheme', *IEEE Transaction on Communications* **38**(11), 1917–1921.

Nakagawa, H., Kojima, H. & Maeda, A. (2005), Chinese term extraction from web pages based on compound word productivity, *in* 'Proceedings of the Second International Joint Conference on Natural Language Processing (IJCNLP '05)', pp. 269–279.

National-Language-Committee (2008), *Lexicon of Common words in Contemporary Chinese* 现代汉语常用词表*(*草案*)*, The Commercial Press 商务出版社.

Ney, H., Essen, U. & Kneser, R. (1994), 'On structuring probabilistic dependences in stochastic language modeling', *Computer Speech and Language* .

Nie, J., Jin, W. & Hannan, M. (1994), A hybrid approach to unknown word detection and segmentation of chinese, *in* 'Proceedings of International Conference on Chinese Computing (ICCC'94)', Singapore.

Nie, J. Y., Brisebois, M. & Ren, X. B. (1997), On chinese text retrieval, *in* 'Proceedings of 19th International Conference on Research and Development in Information Retrieval (SIGIR '96)'.

Nie, J. Y., Chevallet, J. P. & Bruandet, M. F. (1997), Between terms and words for european language ir and between words and bigrams for chinese ir, *in* 'Proceedings of the Sixth Text REtrieval Conference (TREC 6)'.

Nie, J.-Y., Gao, J., Zhang, J. & Zhou, M. (2000), On the use of words and n-grams for chinese information retrieval, *in* 'Proceedings of the Information Retrieval with Asian Languages'.

Nie, J.-Y. & Ren, F. (1999), 'Chinese information retrieval: using characters or words?', *Information Processing and Management* **35**, 443–462.

Packard, J. (2000), *The morphology of Chinese: A Linguistics and Cognitive Approach*, Cambridge University Press, Cambridge.

Palmer, D. (1997), A trainable rule-based algorithm for word segmentation, *in* 'Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics', Madrid.

Palmer, D. & Burger, J. (1997), Chinese word segmentation and information retrieval, *in* 'AAAI Spring Symposium on Cross-Language Text and Speech Retrieval, Electronic Working Notes'.

Peng, F., Feng, F. & Mccallum, A. (2004), Chinese segmentation and new word detection using conditional random fields, *in* 'Coling '04: Proceedings of The 20th International Conference on Computational Linguistics', pp. 562–568.

Peng, F., Huang, X., Schuurmans, D. & Cercone, N. (2002), Investigating the relationship between word segmentation performance and retrieval performance in chinese ir, *in* 'International Conference on Computational Linguistics'.

Peng, F. & Schuurmans, D. (2001), Self-supervised chinese word segmentation, *in* 'Advances in Intelligent Data Analysis, Proceedings of the Fourth International Conference', Cascais, Portugal.

Peterson, J. L. (1986), 'A note on undetected typing errors', *Communications of the ACM* **29**(7), 633–637.

Ponte, J. & Croft, W. B. (1996), Useg: a retargetable word segmentation procedure for information retrieval, *in* 'Symposium on document analysis and information retrieval'.

Ponte, J. M. & Croft, W. B. (1998), A language modeling approach to information retrieval, *in* 'Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval', SIGIR '98, ACM, New York, NY, USA, pp. 275–281.

Powers, D. (1997*a*), Learning and application of differential grammars, *in* 'CoNLL97: Computational Natural Language Learning, ACL', Madrid, pp. 88–96.

Powers, D. (1997*b*), 'Unsupervised learning of linguistic structure: an empirical evaluation', *Int'l Journal of Corpus Linguistics* **2**(1), 91–131.

Powers, D. (2008), Minors as miners: Modelling and evaluating ontological and linguistic learning, *in* 'Proceedings of the 7th Australasian Data Mining Conference'.

Powers, D. (2011), 'Evaluation: From precision, recall and f-measure to roc, informedness, markedness & correlation', *Journal of Machine Learning Technologies* **2**(1), 37–63.

Qiao, W., Sun, M. & Menzel, W. (2008), Statistical properties of overlapping ambiguities in chinese word segmentation and a strategy for their disambiguation, *in* 'Proceedings of the 11th international conference on Text, Speech and Dialogue', TSD '08, Springer-Verlag, Berlin, Heidelberg, pp. 177–186.

Reynaert, M. (2004), Text induced spelling correction, *in* 'Proceedings of the 20th International Conference on Computational Linguistics', COLING '04.

Roark, B., Saraclar, M., Collins, M. & Johnson, M. (2004), Discriminative language modeling with conditional random fields and the perceptron algorithm, *in* 'Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics', Association for Computational Linguistics, Barcelona, Spain.

Robertson, S. E. & Sparck Jones, K. (1976), 'Relevance weighting of search terms', *Journal of the American Society for Information Science* **27**, 129–146.

Robertson, S. E. & Walker, S. (1994), Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval, *in* 'Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval (SIGIR)', pp. 232–241.

Rodriguez, N. J. & Diaz, M. I. (2007), Word processing in spanish using an english keyboard: a study of spelling errors, *in* 'Proceedings of the 2nd International Conference on Usability and Internationalization', pp. 219–227.

Rosenfeld, R. (1994), Adaptive Statistical Language Modeling: A Maximum Entropy Approach, PhD thesis, School of Computer Science, Carnegie Mellon University.

Rosenfeld, R. (2000), 'Two decades of statistical language modeling: where do we go from here?', *Proceedings of The IEEE* **88**, 1270–1278.

Salton, G. & Buckley, C. (1988), 'Term-weighting approaches in automatic text retrieval', *Information Processing and Management* **24**, 513–523.

Salton, G. & McGill, M. J. (1986), *Introduction to Modern Information Retrieval*, McGraw-Hill, Inc., New York, NY, USA.

Sanderson, M. (1994), Word sense disambiguation and information retrieval, *in* 'Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval', SIGIR '94, Springer-Verlag New York, Inc., New York, NY, USA, pp. 142–151.

Schutze, H. & Pederson, J. (1995), Information retrieval based on word senses, *in* 'Proceedings of the Fourth Annual Symposium on Document Analysis and Information Retrieval', pp. 161–175.

Shannon, C. E. (1948), 'A mathematical theory of communication', *Bell Systems Technical Journal* **27**, 379–423,623–656.

Siu, M. & Ostendorf, M. (2000), 'Variable n-gram language modeling and extensions for conversational speech', *IEEE Transactions on Speech and Audio Processing* **8**(1).

Smeaton, A. & Wilkinson, R. (1997), Spanish and chinese document retrieval in trec 5, *in* 'Proceedings of the Sixth Text REtrieval Conference (TREC 6)'.

Song, F. & Croft, W. B. (1999), A general language model for information retrieval, *in* 'Proceeding of International Conference on Information and Knowledge Management', pp. 316–321.

Sparck Jones, K. (1997), *Search term relevance weighting given little relevance information*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 329–338.

Sproat, R. & Emerson, T. (2003), The first international chinese word segmentation bakeoff, *in* 'Proceedings of the Second SIGHAN Workshop on Chinese Language Processing'.

Sproat, R. & Shih, C. (1990), 'A statistical method for finding word boundaries in chinese text', *Computer Processing of Chinese and Oriental Languages* **4**(4).

Sproat, R. & Shih, C. (2002), Corpus-based methods in chinese morphology and phonology, *in* 'International Conference on Computational Linguistics'.

Sproat, R., Shih, C., Gale, W. & Chang, N. (1996), 'A stochastic finite-state word-segmentation algorithm for chinese', *Computational Linguistics* **22**(3).

Stehouwer, H. & den Bosch, A. V. (2007), Putting the t where it belongs: Solving a confusion problem in dutch, *in* S. Verberne, H. van Halteren & P.-A. Coppen, eds, 'Computational Linguistics in the Netherlands 2007: Selected Papers from the 18th CLING Meeting', Nijmegen, The Netherlands, pp. 21–36.

Stehouwer, H. & van Zaanen, M. (2009*a*), Language models for contextual error detection and correction, *in* 'Proceedings of the EACL 2009 Workshop on Computational Linguistic Aspects of Grammatical Inference', CLAGI '09, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 41–48.

Stehouwer, H. & van Zaanen, M. (2009*b*), Token merging in language model-based confusible disambiguation, *in* '21st Benelux Conference on Artificial Intelligence (BNAIC)'.

Stehouwer, H. & van Zaanen, M. (2010), Using suffix arrays as language models: Scaling the n-gram, *in* '22nd Benelux Conference on Artificial Intelligence (BNAIC)'.

Stolcke, A. (2002), Srilm – an extensible language modeling toolkit, *in* 'Proceedings of International Conference on Spoken Language Processing', Vol. 2, Denver, pp. 901–904.

Su, F., Wang, D. & Dai, G. (2004), 'A rule-statistic model based on tag and an algorithm to recognize unknown words', *Computer Engineering and Applications* 计算机工程与应用 **15**, 43–45.

Sun, H. (1998), Distributional property of collocations in the texts, *in* 'Proceeding of International Conference on Chinese Information Processing (Chinese)'.

Sun, M., Shen, D. & Tsou, B. K. (1998), Chinese word segmentation without using lexicon and hand-crafted training data, *in* 'Meeting of the Association for Computational Linguistics', pp. 1265–1271.

Tanaka-Ishii, K. & Jin, Z. (2006), From phoneme to morpheme: Another verification using a corpus, *in* 'Proceedings of the 21st International Conference Computer Processing of Oriental Languages (ICCPOL 2006)', pp. 234–244.

Tanaka-Ishii, K. & Jin, Z. (2008), 'From phoneme to morpheme – another verification in english and chinese using corpora', *Studia Linguistica* **62**, 224–248.

Tang, L.-X., Geva, S., Trotman, A. & Xu, Y. (2010), A boundary-oriented chinese segmentation method using n-gram mutual information, *in* 'Proceedings of the Joint Conference on Chinese Language Processing'.

Teahan, W., Wen, Y. & R. McNab, I. W. (2000), 'A compression-based algorithm for chinese word segmentation', *Computational Linguistics* **26**(3), 375–394.

Tsai, J.-L. (2006*a*), Bmm-based chinese word segmentor with word support model for the sighan bakeoff 2006, *in* 'Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing', Association for Computational Linguistics, Sydney, Australia, pp. 130–133.

Tsai, J.-L. (2006*b*), Using word support model to improve chinese input system, *in* 'Proceeding of the COLING/ACL 2006 Main Conference Poster Sessions', Sydney, pp. 842–849.

Tseng, C.-H. (2008), Chinese input method based on first mandarin phonetic alphabet for mobile devices and an approach in speaker diarization with divide-and-conquer, Master's thesis, National Sun Yat-Sen University.

Tseng, C.-H. & Chen, C.-P. (2006), Chinese input method based on reduced mandarin phonetic alphabet, *in* 'Proceedings of the Ninth International Conference on Spoken Language Processing'.

Tseng, H., Chang, P., Andrew, G., Jurafsky, D. & Manning, C. (2005), A conditional random field word segmenter, *in* 'Fourth SIGHAN Workshop on Chinese Language Processing'.

Tung, C. & Lee, H. (1994), 'Identification of unknown words from a corpus', *Computer Processing of Chinese and Oriental Languages* **8**(supplement).

Ukkonen, E. (1995), 'On-line construction of suffix tree', *Algorithmica* **14**(3), 249–260.

Van den Bosch, A. (2006*a*), All-word prediction as the ultimate confusable disambiguation, *in* 'Proceedings of the Workshop on Computationally Hard Problems and Joint Inference in Speech and Language Processing', CHSLP '06, pp. 25–32.

Van den Bosch, A. (2006*b*), 'Scalable classification-based word prediction and confusible correction', *Traitement Automatique des Langues* **46**(2), 39–63.

Van Rijsbergen, C. J. (1979), *Information retrieval*, London: Butterworths.

Voorhees, E. & Harman, D. (1996), Overview of the fifth text retrieval conference (trec 5), *in* 'Proceedings of the Sixth Text REtrieval Conference (TREC 5)'.

Voorhees, E. M. & Harman, D., eds (2005), *TREC: Experiment and Evaluation in Information Retrieval*, MIT Press.

Wang, W. M. (2005), 'The three principles of computer chinese character keyboard design', *Chinese Journal of Computers* **28**(5).

Wang, X. L. (1993), 'Chinese input system by pinyin sentense: Insun', *Journal of Chinese Information Processing* **7**(2), 45–54.

Weiner, P. (1973), Linear pattern matching algorithms, *in* 'IEEE Symposium on Foundations of Computer Science', pp. 1–11.

Wilcox-O'hearn, L. A., Hirst, G. & Budanitsky, A. (2008), Real-word spelling correction with trigrams: A reconsideration of the mays, damerau, and mercer model, *in* 'Proceedings of Conference on Intelligent Text Processing and Computational Linguistics', pp. 605–616.

Wilkinson, R. (1998), Chinese document retrieval at trec 6, *in* 'Proceedings of the Sixth Text REtrieval Conference (TREC 6)'.

Winston, P. (1993), *Artificial Intelligence*, Addison Wesley.

Witten, I. H. & Bell, T. C. (1991), 'The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression', *IEEE Transactions on Information Theory* **37**, 1085–1094.

Wu, A. & Jiang, Z. (2000), Statistically-enhanced new word identification in a rule-based chinese system, *in* 'Proceedings of the Second Workshop on Chinese Language Processing (ACL00)', pp. 46–51.

Wu, L., Huang, X.-j., Guo, Y., Liu, B. & Zhang, Y. (2000), Fdu at trec 9: Clir, filtering and qa tasks, *in* 'Proceedings of the Nineth Text REtrieval Conference (TREC 9)'.

Xia, F. (2000), The segmentation guidelines for the penn chinese treebank (3.0), Technical report, University of Pennsylvania.

Xiao, J., Liu, B. & Wang, X. (2007), 'Exploiting pinyin constraints in pinyin-to-character conversion task: a class-based maximum entropy markov model approach', *Computational Linguistics and Chinese Language Processing* **12**, 325–348.

Xiaolong, W., Qingcai, C. & Yeung, D. S. (2004), 'Mining pinyin-to-character conversion rules from large-scale corpus: a rough set approach', *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* **34**, 834–844.

Xu, J., Zens, R. & Ney, H. (2004), Do we need chinese word segmentation for statistical machine translation?, *in* O. Streiter & Q. Lu, eds, 'Proceedings of ACL SIGHAN Workshop 2004', Association for Computational Linguistics, Barcelona, Spain, pp. 122–128.

Xu, P. & Jelinek, F. (2007), 'Random forests and the data sparseness problem in language modeling', *Computer Speech and Language* **21**(1), 105 – 152.

Xu, X., Zhu, M., Fei, X. & Zhu, J. (2010), High oov-recall chinese word segmenter, *in* 'Proceedings of the Joint Conference on Chinese Language Processing'.

Xue, N. (2003), 'Chinese word segmentation as character tagging', *Computational Linguistics and Chinese Language Processing* **8**(1), 29–48.

Yamamoto, M. & Church, K. (2001), 'Using suffix arrays to compute term frequency and document frequency for all substrings in a corpus', *Computational Linguistics* **27**(1), 1–30.

Yang, D. & Powers, D. M. W. (2008), Automatic thesaurus construction, *in* 'Australasian Computer Science Conference', Vol. 74, pp. 147–156.

Yang, Y. (1995), Noise reduction in a statistical approach to text categorization, *in* 'Proceedings of 18th ACM International Conference on Research and Development in Information Retrieval (SIGIR '95)'.

Yarowsky, D. (1994), Decision lists for lexical ambiguity resolution: application to accent restoration in spanish and french, *in* 'Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics', Las Cruces, pp. 88–95.

Yeh, C. & Lee, H. (1991), 'Rule-based word identification for mandarin chinese sentences - a unification approach', *Computer Processing of Chinese and Oriental Languages* **5**(2).

Yu, S. (1999), Guidelines for the annotation of contemporary chinese texts: word segmentation and pos-tagging, Technical report, Institute of Computational Linguistics, Beijing University, Beijing.

Yu, S., Zhu, Y., Wang, f. & Zhang, Y. (1998), *The grammatical knowledge-base of contemporary Chinese (*现代汉语语法信息词典详解*)*, Hsinghua University and Guangxi Press.

Zhai, C. (2008), 'Statistical language models for information retrieval: A critical review', *Foundations and Trends in Information Retrieval* **2**, 137–213.

Zhang, H.-P., Liu, Q., Cheng, X.-Q., Zhang, H. & Yu, H.-K. (2003), Chinese lexical analysis using hierarchical hidden markov model, *in* 'Proceedings of the

Second SIGHAN Workshop on Chinese Language Processing', Association for Computational Linguistics, Sapporo, Japan, pp. 63–70.

Zhang, J., Gao, J. & Zhou, M. (2000), Extraction of chinese compound words - an experimental study on a very large corpus, *in* 'The second Chinese Language Processing Workshop attached to ACL2000', Hong Kong.

Zhang, Y. & Clark, S. (2007), Chinese segmentation with a word-based perceptron algorithm, *in* 'Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics', pp. 840–847.

Zhao, H., Huang, C.-N., Li, M. & Lu, B.-L. (2010), 'A unified character-based tagging framework for chinese word segmentation', *ACM Transactions on Asian Language Information Processing (TALIP)* **9**.

Zhao, H. & Kit, C. (2008*a*), An empirical comparison of goodness measures for unsupervised chinese word segmentation with a unified framework, *in* 'Proceedings of the Third International Joint Conference on Natural Language Processing (IJCNLP-2008)'.

Zhao, H. & Kit, C. (2008*b*), 'Exploiting unlabeled text with different unsupervised segmentation criteria for chinese word segmentation', *Research in Computing Science* **33**, 93–104.

Zhao, H. & Kit, C. (2011), 'Integrating unsupervised and supervised word segmentation: The role of goodness measures', *Information Sciences* **181**, 163–183.

Zhao, H. & Liu, Q. (2010), The cips-sighan clp2010 chinese word segmentation backoff, *in* 'Proceedings of the Joint Conference on Chinese Language Processing'.

Zhikov, V., Takamura, H. & Okumura, M. (2010), An efficient algorithm for unsupervised word segmentation with branching entropy and mdl, *in* 'Proceedings of the 2010 Conference on Empirical Methods on Natural Language Processing (EMNLP)', pp. 832–842.

Zhou, G. D. & Lua, K. T. (1999), 'Interpolation of n-gram and mutual-information based trigger pair language models for mandarin speech recognition', *Computer Speech and Language* **13**(2), 125–141.

Zhu, D. X. (1982), *The Lectures of Grammars*, The Commercial Press.

Zou, F., Wang, F. L., Deng, X. & Han, S. (2006), 'Stop word list construction and application in chinese language processing', *WSEAS Transaction on Information Science and Applications* **3**(6), 1036–1045.