Computational Linguistics and Chinese Language Processing
Vol. 12, No. 4, December 2007, pp. 349-370

Modeling Taiwanese Southern-Min Tone Sandhi Using Rule-Based Methods

Un-Gian Iunn[*], Kiat-Gak Lau[+], Hong-Giau Tan-Tenn[#], Sheng-An Lee[*], and Cheng-Yan Kao[*]

Abstract

A sizable corpus of Taiwanese text in Latin script has been accumulated over the past two hundred or so years. However, due to the special status of Taiwan, few people can read these materials at present. It is regrettable that the utilization of these plentiful materials is very low.

This paper addresses problems raised in the Taiwanese Southern-Min tone sandhi system by describing a set of computational rules to approximate this system, as well as the results obtained from its implementation. Using the romanized Taiwanese Southern-Min text as source, we take the sentence as the unit, translate every word into Chinese via an online Taiwanese-Chinese dictionary (OTCD), and obtain the part-of-speech (POS) information from the Chinese Electronic Dictionary (CED) made by the Chinese Knowledge and Information Processing (CKIP) group of Academia Sinica. By using the POS data and tone sandhi rules based on linguistics, we then tag each syllable with its post-sandhi tone marker. Finally, we implement a Taiwanese Southern-Min tone sandhi processing system which takes a romanized sentence as an input and then outputs the tone markers.

Our system achieves 97.39% and 88.98% accuracy rates with training and test data, respectively. Finally, we analyze the factors influencing error for the purpose of future improvement.

[*] Department of Computer Science and Information Engineering, National Taiwan University
E-mail: {d93001, d93005, cykao}@csie.ntu.edu.tw
[+] Phahng Taiwanese Workshop, http://www.phahng.idv.tw
E-mail: kiatgak@gmail.com
[#] Independent scholar
E-mail: chenchen@umdnj.edu

[Received October 31, 2006; Revised July 6, 2007; Accepted July 11, 2007]

Keywords: Taiwanese Southern-Min, Written Taiwanese, Tone Sandhi System, Taiwanese Romanization

1. Introduction

1.1 Background and Motivation

Taiwanese is often used in daily life in Taiwan, but written Taiwanese is less common by far. Even so, the history of written Taiwanese stands at well over a century [Tiunn 2001]. At present, there are several dozen if not more than a hundred proposed phonetic and writing systems for Taiwanese [Iunn and Tiunn 1999]. The orthography adopted by this article is Peh-oe-ji (*POJ*, 白話字, also known as *Latinized Taiwanese* or *Missionary Romanization System for Taiwanese*).

Under the auspices of the National Museum of Taiwanese Literature, the Department of Taiwanese Literature of Cheng Kung University carried out a project titled “The Collection and Cataloging of Taiwanese Peh-oe-ji Literature Data” (CCTPLD). Although many texts have already been lost due to the alternation of political status, this project nevertheless revealed nearly 2,000 POJ books and periodicals, with publication sites spread over Taiwan, Xiamen (Amoy), Shanghai, Guangzhou (Canton), Hong Kong, Singapore, the Philippines, London, Japan, and beyond. The amount of publishing peaked in the 1950’s and 60’s [Iunn and Tan-Tenn, unpublished]. The scope covers both formally published books and periodicals as well as non-published items such as personal letters and medical charts. Later on, the government, citing supposedly detrimental effects of POJ on Mandarin promotion, banned its use and thus caused the rapid decline of this practice.

We hope that the extant materials collected by the above-mentioned CCTPLD project can be accessed by more people, as well as contribute to both basic and applied Taiwanese research. As most people nowadays are not familiar with Latinized Taiwanese, use of state-of-the-art text-to-speech technology would enhance the value of these materials to the general public.

Tone sandhi represents a challenging problem to be solved before one can successfully transform the written Taiwanese text to its natural speech-like tonal contour. This is because the written form of Latinized Taiwanese represents the tones as "basic tones", the tones of syllables when they are pronounced in isolation. At the level of the word, all syllables except the last one are usually pronounced differently (that is, they manifest tone sandhi). At the level of a whole sentence, in most situations only the last syllables next to the boundary of the phrases or structural markers are read as basic tones, the others being read as sandhi tones. In fact, besides the "regular tone sandhi" mentioned above, there are still several other kinds of tone sandhi phenomena which will be discussed in detail later.

We will first formulate the sandhi rules, which are the key to correct pronunciation and the core issue of this paper. The input of our experiment mainly consists of the data collected by the CCTPLD project; these data are processed by our sandhi system to produce sandhi-marked final outputs. Due to the lack of tagged data, we adopt the rule-based model, not the statistical model in this experiment. Figure 1 describes the skeleton of our system, and the webpage http://iug.csie.dahan.edu.tw/nmtl/dadwt/ demonstrates the results. Three of the authors who are native Taiwanese speakers evaluated the outputs for their accuracy.

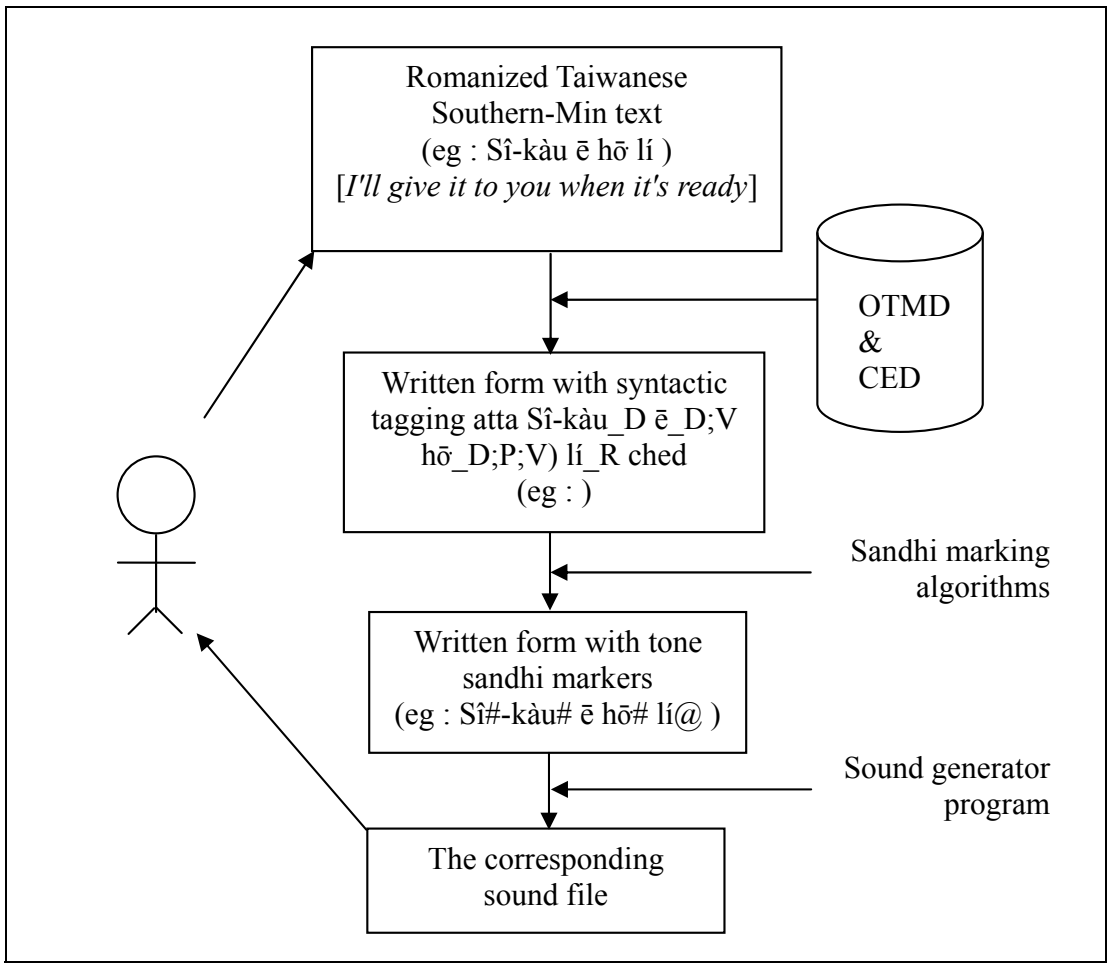


Figure 1. Taiwanese Southern-Min tone sandhi system diagram

1.2 The Tone Sandhi Problem

Tones in Taiwanese are traditionally analyzed as consisting of *piâⁿ* [平], *siáng* [上], *khì* [去], *ji̍p* [入], each having *im* [陰, "*yin*"] and *iâng* [陽, "*yang*"] except *siáng*. So there are a total of seven tones. Following the sequence of *im-piâⁿ*[陰平], *siáng*[上], *im-khì*[陰去], *im-ji̍p*[陰入], *iâng-piâⁿ*[陽平], *iâng-khì*[陽去], *iâng-ji̍p*[陽入], they are numbered 1 (high flat), 2 (high to

low), 3 (low), 4 (middle short), 5 (low rising), 7 (middle flat), and 8 (high short). The tone pitch is described within the parentheses. Please refer to the following examples for tone diacritics. In this paper, all examples are written in Taiwanese. For the sake of apprehensibility, we also add the Mandarin and English translations.

Tone sandhi is a very important characteristic of Taiwanese. At the word level, the last syllable is usually pronounced as basic tone and the others as sandhi tones. In example (1), the underlined syllables are pronounced as basic tones, the others as sandhi tones:

(1) tâi [台, "*platform*"]
Tâi-gí [台語, "*Taiwanese language*"]
Tâi-gí-bûn [台語文, "*written Taiwanese*"]
Tâi-gí bûn-hák [台語文學, "*Taiwanese literature*"]
Tâi-gí bûn-hák-sú [台語文學史, "*history of Taiwanese literature*"]

At the level of the syllable or the word, tone sandhi may manifest itself in at least the following several ways:

(a) Normal sandhi: using reduplicated syllables as examples (the numbers within parentheses are reading tones).

(2) (i) tone 1 → tone 7: "chheng-chheng" (7,1) [清清, "*clear*"]
(ii) tone 7 → tone 3: "chēng-chēng" (3,7) [靜靜, "*quiet*"]
(iii) tone 3 → tone 2: "chhiò-chhiò" (2,3) [笑笑, "*smiley*"]
(iv) tone 2 → tone 1: "léng-léng" (1,2) [冷冷, "*cold*"]
(v) tone 5 → tone 7 or 3 (northern Taiwan): "âng-âng" (7/3,5) [紅紅, "*red*"]
(vi) tone 4 → tone 8 (-p/t/k) or 2 (-h): like "sip-sip" (8,4); [濕濕, "*moist*"]
"khoeh-khoeh" (2,4) [擁擠, "*crowd*"]
(vii) tone 8 → tone 4 (-p/t/k) or 3 (-h): like "tit-tit" (4,8) [直直, "*straight*"];
"jo̍ah-jo̍ah" (3,8) [熱熱, "*hot*"]

(b) Following sandhi: this pattern generally occurs with pronouns or the suffix of names. The tone pitch depends on that of the preceding syllable and is either tone 1 (high), 3 (middle), or 7 (low).

(3) (i) "A-eng--a" (7,1,1) [阿英, *a personal name*] (the second "a" is a suffix)
(ii) "góa lâi khòaⁿ -- i" (1,7/3,3,3) [我來看他, "*I come to see him/her*"] (the basic tone of "i"[他, "*(s)he*"] is tone 1)
(iii) "hō --lí" (7,7) [給你, "*give you*"] (the basic tone of "lí"[你, "*you*"] is tone 2)

(c) Neutral sandhi: the syllable immediately preceding the neutral sandhi (marked orthographically with double hyphens same as (b)) is read as basic tone, and the tones of the neutral sandhi are pronounced softly as if they were tone 3 or tone 4.

(4) (i) "Tân--sian-siⁿ" (5,3,3) [陳先生, "*Mr. Tân*"] (the original tones of "sian-siⁿ "[先生, "*Mr.*"] are tone 7 and tone 1)
(ii) "kiâⁿ--chhut-lâi" (5,4,3) [走出來, "*walk out*"] (the original tones of "chhut-lâi" [出來, "*out*"] are tone 8 and tone 5)

(d) Double sandhi: this pattern mostly appears in syllables ending in the glottal stop (-h) and having tone 4. The normal sandhi rules are applied twice in sequence (*i.e.* tone 4 → tone 2 → tone 1):

(5) (i) "beh thȧk-chu" (1,4,1) [要讀書, "*want to read books*"] ("beh" [要, "*want*"] is tone 4, but rather than becoming tone 2, it becomes tone 1)
(ii) "khì gōa-kháu" (1,3,2) [去外面, "*go outside*"] ("khì"[去, "*go*"] is tone 3, but rather than becoming tone 2, it becomes tone 1)

(e) Pre-*á* sandhi: the syllables before á do not follow normal sandhi rules unless they are tone 1 or 2.

(6) (i) tone 1 → tone 7: "sun-á" (7,2) [姪子, "*nephew*"]
(ii) tone 2 → tone 1: "chháu-á" (1,2) [小草, "*grass*"]
(iii) tone 3 → tone 1: "tàⁿ-á" (1,2) [攤位, "*stall*"]
(iv) tone 4 → tone 8 (-p/t/k) or tone 1 (-h): "tek-á" (8,2) [竹子, "*bamboo*"]
"thih-á" (1,2) [鐵,"*iron*"]
(v) tone 5 → tone 7: "lô-á " (7,2) [爐子, "*oven*"]
(vi) tone 7 does not change: "phō-á" (7,2) [簿子, "*tablet*"]
(vii) tone 8 → tone 4 (-p/t/k) or tone 7 (-h): "chhȧt-á" (4,2) [賊, "*thief*"]
"hio̍h-á " (7,2) [葉, "*leaf*"]

(f) Triplicate sandhi: the first syllable of triplicated words does not follow normal sandhi rules unless it is of tone 2, 3, or 4:

(7) (i) tone 1 → tone 5: like "chheng-chheng-chheng" (5,7,1) [清清清, "*very clear*"]
(ii) tone 2 → tone 1: like "ún-ún-ún" (1,1,2) [穩穩穩, "*very stable*"]
(iii) tone 3 → tone 2: like "hèng-hèng-hèng" (2,2,3) [興興興, "*very interesting*"]
(iv) tone 4 → tone 8 (-p/t/k) or tone 2 (-h): like "sip-sip-sip" (8,8,4) [濕濕濕, "*very humid*"] "bah-bah-bah" (2,2,4) [肉肉肉, "*very fat*"]
(v) tone 5 → (similar to) tone 5: like "kôaⁿ-kôaⁿ-kôaⁿ" (5,7/3,5) [冷冷冷, "*very cold*"]
(vi) tone 7 → (similar to) tone 5: like "chēng-chēng-chēng" (5,3,7) [靜靜靜, "*very quiet*"]
(vii) tone 8 → (similar to) tone 5: like "tit-tit-tit" (5,4,8) [直直直, "*very straight*"] "pe̍h-pe̍h-pe̍h" (5,3,8) [白白白, "*very white*"]

(g) Rising sandhi: this pattern usually occurs on loanwords from Japanese; the sandhi tone is similar to tone 5.

(8) "ǒai-siak-chù" (5,8,3) [白襯衫, "*white shirt*"]
"khǎn-páng" (5,2) [看板, "*signboard*"]
"hǎn-tó͘-lù "(5,1,3) [方向盤, "*steering wheel*"]

We collate the above sandhi phenomena in *Table 1*.

Table 1. Taiwanese Southern-Min tone sandhi phenomena

Normal sandhi	Basic tone of syllable	1	2	3	4	5	7	8
	Sandhi tone	7	1	2	8/2	7/3	3	4/3
Following sandhi	Basic tone of preceding syllable	1	2	3	4	5	7	8
	Sandhi tone	1	3	3	3	7	7	1
Neutral sandhi	Basic tone of preceding syllable	1	2	3	4	5	7	8
	Sandhi tone	3	3	3	4/3	3	3	4/3
Double sandhi	Basic tone of syllable	-	-	3	4	-	-	-
	Sandhi tone			1	1			
Pre-*á* sandhi	Basic tone of preceding syllable of "*á*"	1	2	3	4	5	7	8
	Sandhi tone	7	1	1	8/1	7	7	4/7
Triplicate sandhi	Basic tone of the first syllable of three	1	2	3	4	5	7	8
	Sandhi tone	5	1	2	8/2	5	5	5

1.3 Historial Review

[Lin and Chen 1999] describes an early sandhi system. Users input Chinese texts, and the system outputs Taiwanese texts with pronunciation. The corpus is news reports in Chinese. They used the word segmentation and tagged data from the CKIP group and the Taiwanese-Chinese dictionary from Robert Cheng to map the Chinese news into Taiwanese (in both Han and Latin scripts). The sandhi rules applied were as follows: a) pronounce the last syllable at the end of a sentence as basic tone; b) pronounce the syllable before the particle *ê* as basic tone; c) pronounce the last syllable of a noun as basic tone; d) pronounce other syllables as normal sandhi tones. An accuracy rate of 82.53% was reported. However, as the system did not take Taiwanese as input, word order and semantic ambiguities were not taken into account when converting, the translation was not quite native-like.

[Liang *et al.* 2004] is a recent text-to-speech system for Taiwanese Southern-Min. Its input was a large corpus of Chinese news texts, but sentences longer than 20 syllables were removed. It utilized a dictionary to convert the Chinese text into Taiwanese Southern-Min, followed by word segmentation, phonetic marking, and rule-based sandhi processing to generate speech files. Due to the size of the corpus, only the first 200 sentences generated were evaluated by two Taiwanese-speaking experts. The accuracy rates were 97% for word segmentation, 89% for pronunciation marking, and 65% for rule-based sandhi processing.

Compared with the above systems, our approach has some major differences: a balanced Taiwanese corpus for both literary and non-literary sources (about 50% each) was prepared; no translation from Chinese to Taiwanese; and no limits for length of sentences. In addition, because the text is written in Latinized script, we do not need to manipulate word segmentation and phonetic marking. However, compared to text with Han character script, there is a more rigorous challenge to deal with homonymy, especially with monosyllabic words.

2. Method

2.1 Data

The input data of our system are from the CCTPLD project. Following POJ orthography, syllables of a word are joined by hyphens, and the words are separated with spaces.

We select parts of four sources as training data. The training data sources are shown in *Table 2*.

Table 2. Training data sources

Book or aticle	year	author	genre
"Sin-bûn ê chảp-liȯk" [新聞的雜錄, "*News Bulletin*"]	1913	unknown	journalism
"Chảp-hāng kóan-kiàn" [十項管見, "*Ten Humble Opinions*"]	1925	Chhòa Pôe-hóe [蔡培火]	discourse
"Chháu-tui téng ê bîn-bāng -- jî-tông chong-kàu kò-sū" [草堆上的夢─兒童宗教故事, "*Dreams on the Grass Stack -- Religious Stories for Children*"]	1955	Ńg Hôai-un [黃懷恩]	short stories
"Tang-pō thôan-tō kiàn-bûn kì" [東部傳道見聞記, "*Record of Preaching in Eastern Taiwan*"]	1961	Tân Kàng-hâng [陳降祥]	journalism

The published dates of the above sources range from Japan-ruled era (1895-1945) to postwar era (1945-). Two paragraphs are selected from each book, with a total of 614 syllables (438 word tokens).

In addition to data drawn from the same project, the test data also include some other sources we collected. Four sources are selected as well. The test data sources are shown in *Table 3*.

Table 3. Test data sources

Book or article	year	author	genre
"Pe̍h-ōe-jī ê lī-ek" [白話字的利益, "*The Benefits of Using Peh-oe-ji*"]	1885	Reverend Iáp [葉牧師]	discourse
"Kau-chiàn ê Siau-sit" [交戰的消息, "*News of the War*"]	1905	the editorial office of *Tâi-lâm Prefectural Church News*	report
"Thiàⁿ lí iâⁿ kè thong sè-kan" [疼愛你勝過全世界, "*Caring About You More Than the Whole World*"]	1955	Lōa Jîn-seng [賴仁聲]	novel
"Ài lí kap ài i pîⁿ-á chōe" [愛妳和她一樣多, "*Loving You as Much as Her*"]	1997	Lô Tàn-chhun [盧誕春]	prose

Two or three paragraphs are selected from each book or article. The test data total 962 syllables (656 word tokens) and also cover two eras but with a longer time span.

2.2 Part of Speech Tagging

As there is no standard on part of speech (POS) for Taiwanese at present, we use the standard of Chinese instead (see Results section). We obtain the corresponding Chinese translation for each Taiwanese word by looking up the Taiwanese-Chinese On-line Dictionary. [Iunn 2003] We, then, look up the POS of the Chinese in the 80,000-word CED. Ambiguity encountered includes:

(a) homonymy, especially monosyllabic homonyms;

(b) one-to-many mapping when mapping Taiwanese to Chinese;

(c) multiple possible POSs for each Chinese word.

To resolve homonymy, we choose the word with the highest querying frequency. We found out that this strategy works under most situations. Due to the fact that one Taiwanese word may map to multiple Chinese words, and one Chinese word could possibly have multiple POSs, there may be multiple POSs for one Taiwanese word. We initially retain all candidate POSs in tagging and only attempt to narrow down the list upon applying the sandhi algorithm. Of the 46 POSs in the Chinese Electronic Dictionary, we adopt the top level and adjust certain POSs known to affect tone sandhi. For example, Vh (state intransitive verb, etc.) is marked A, Nh (pronoun) marked R, Ng (postposition) marked G, and Nd (time) marked S. The POS classes we used are shown in *Table 4*.

Table 4. POS classes

POS	statement	POS	statement	POS	statement
A	adjective	I	interjection	R	pronoun
C	conjunction	M	special marker	S	time
D	adverb	N	noun	T	auxiliary
G	postposition	P	preposition	V	verb

As for unknown words, if they are of the form 'XX' or 'XXX' (duplicate or triplicate syllables), we mark them as A (adjective). Other words are marked as N (noun).

2.3 Tone Sandhi Marks

The marks representing tone sandhis are listed in *Table 5*. Words with normal sandhi are usually not marked.

Table 5*. *Sandhi marks

Symbol	Phenomenon	Symbol	Phenomenon
(none)	Normal sandhi	**$**	Double sandhi
#	Basic tone	**&**	Pre-*á* sandhi
@	Following sandhi	~	Triplicate sandhi
%	Neutral sandhi	^	Rising sandhi

2.4 Tone Sandhi Rules

Tone sandhi rules are the most important part of this study. The algorithm for sandhi marking is shown in *Table 6*.

Table 6. Tone sandhi marking algorithm

	Rule	Remark
1	Apply normal sandhi to all syllables	
2	Mark the last syllable as basic tone #	
3	*ê* [的, "*of*"] : Mark the syllable preceding *ê* as basic tone #	*ê* is a special marker
4	A/A Pair 4.1 A/A Pair: Mark the last syllable of the first word as basic tone #	POS level, with ambiguity
5	N/V, N/A, N/P, N/R, and N/D Pairs 5.1 N/V Pair: Mark the last syllable of the first word as basic tone # 5.2 N/A Pair: Mark the last syllable of the first word as basic tone # 5.3 N/P Pair: Mark the last syllable of the first word as basic tone # 5.4 N/R Pair: Mark the last syllable of the first word as basic tone # 5.5 N/D Pair: Mark the last syllable of the first word as basic tone #	POS level , with ambiguity
6	C: Mark the last syllable of the preceding word as basic tone #	POS level
7	G: Mark the last syllables of both the preceding word and the word itself as basic tones #'s	POS level, without ambiguity
8	S: Mark the last syllable of this word as basic tone #	
9	POS R 9.1 *i* / *in* [他(們), "*(s)he/they*"] : Mark them as normal sandhi even if they are the last syllables 9.2 *góa* / *lí* / *gún* / *góan* / *lán* / *lín* [我/你(們)(的), "*I/you/my/our/your*"]of POS R: Mark them as normal sandhi if they are not the last syllable	POS/Word level
10	Sentence-final *kóng* [講, "*say*"] : Mark this word as normal sandhi if the delimiter is among [, ： : "] and there is any word of POS R in front of this word (note: this rule needs to be refined in case there is a name in front of this word)	Word level, induced from training data

Table 6. Tone sandhi marking algorithm

11	pre-*á* [*á is suffixof a word*]: Mark any syllables just before *á* as pre-*á* sandhi &	Syllable level
12	Double sandhi	
	12.1 *beh* [要, "*want*"] : Mark any *beh* as double sandhi $ unless it appears at the end, including those within a word, such as *kioⁿ-beh, tih-beh* .	Syllable level
	12.2 *khì* [去, "*go*"] : Mark *khì* as double sandhi $ if the POS of the immediately following word is N or V, unless it appears at the end	Word level
	12.3 *koh* [再, "*again*"] : Mark any *koh* as double sandhi $, including those within a word, such as *chiah-koh*[再, "*and then*"] or *iáu-koh*[還是, "*still*"], unless it appears at the end	Syllable level, extended from training data
	12.4 *kah*[和, "*and*"] : Mark any *kah* as double sandhi $ unless it appears at the end	Word level
13	Neutral sandhi of --: Mark the syllable just before -- as basic tone, and mark each syllable after -- as neutral sandhi %	Word level
14	Triplicate sandhi: Mark the first syllable as triplicate sandhi if that word has 3 syllables of the same spelling	Word level
15	Special words	Word level, extend from training data beacuse of not yet standardized
	15.1 *sím-mih* / *sím-mn̂h*[什麼, "*what*"] : Change these words into sím-mí (sandhi marks not changed)	
	15.2 *án-ni* / *àn-ni* / *an-ni* / *an-nī* [這樣, "*thus*"] : Change these words into *án-ni* and to mark its sandhi marks as t#	
16	Markers	
	16.1 *iah-sī* / *ah-sī* / *iáh-sī* / *áh-sī* / *á-sī* [或是, "*or*"] : Mark the last syllable before these words as basic tone #	word level, extended from training data
	16.2 V *sī* [是, "*is*"] V: Mark the last syllable of the verb that just before *sī* as basic tone # if this verb appears again after *sī*	Sentence pattern level, induced from training data
	16.3 *che* / *he* / *chia* / *hia* [這/那(裡), "*this/that/(t)here*"] : Mark these words as basic tone #	word level
	16.4 *ū-sî* [有時, "*sometimes*"] / *put-sî* [不時, "*from time to time*"] / *kui-khì* [乾脆, "*just*"] / *óan-jiân* [宛然, "*like*"] / *gôan-lâi* [原來, "*originally*"] *chiong-lâi* [將來, "*future*"] / *chiông-lâi* [從來, "*always*"] / *sui-jiân*/ *sui-bóng* [雖然, "*though*"] / *sî-siông* [時常, "*often*"] / *hui-siông* [非常, "*very*"] / *sit-chāi* [實在, "*really*"] /	word level, extended from training data

Table 6. Tone sandhi marking algorithm

	sî-chūn [時候, "*(the duration of) time*"] : Mark the last syllables of these words as basic tone #	
	16.5 *chiū* / *tō* [就, "*as soon as*"] : Mark the syllable of the word just before as basic tone # if the POS of the word is A	word level, induced / extended from training data
	16.6 *sî-kàu* [到時候, "*at that time*"]: Mark both of the two syllables of this word as basic tones	word level, induced from training data
17	T: Mark the last syllable of a word as basic tone if the word is just before a word of POS T in the end	POS level
18	Other sandhi: 18.1 *teh* [在, "*at*"]: Mark *teh* or the *teh* in *tī-teh* as other sandhi ^	word level, our observations
19	Neutral sandhi	
	19.1 *chhut-lâi* [出來, "*come out*"] / *chhut-khì* [出去, "*go out*"] / *lo̍h-lâi* [下來, "*come down*"] *lo̍h-khì* [下去, "*go down*"] *kòe-lâi* [過來, "*come up*"] *kòe-khì* [過去, "*pass away*"]: Mark the last syllable of a verb just before these words as basic tone #, and mark these words as neutral sandhi %	word level
	19.2 *sian-si^{n}*/*sin-se^{n}*/*sian-se^{n}* [先生, "*Mr.*"]: Mark the word before these words as basic tone # and these words as neutral sandhi %, if the first letter of the preceding word is uppercase	word level
	19.3 *bô* [無, "*have nothing*"] at the end 19.3.1 *á* / *á-sī* / *iah* / *iah-sī* / *ah* /*ah-sī* [或是, "*or*"]: if the preceding word is among these words, do nothing 19.3.2 Otherwise: Mark the last syllable of the word just before *bô* as basic tone #, and mark *bô* as neutral sandhi %	word level, indecud / extended from training data
	19.4 *bē*/*bōe* [不會, "*will not*"] at the end 19.4.1 *ē*/*ōe* [會, "*be good at*"] *ē-hiáu*/*ōe-hiáu* [會, "*be good at*"]: Mark any final *bē*/*bōe* as neutral sandhi % 19.4.2 *á* / *á-sī* / *iah* / *iah-sī* / *ah* /*ah-sī* [或是, "*or*"]: Mark the *bē*/*bōe* as basic tone # if any of these words immediately precedes it 19.4.3 Otherwise: Do nothing as it could be ambiguous (*e.g. bē/bōe* [賣, "*sell*"])	word level, indecud / extended from training data
20	R at the end *góa* / *lí* / *i* / *gún* / *góan* / *lán* / *lín* / *in* [我/你/他(們)(的), "*I/we/my/our/ you(r)(s)/(s)he/they/their*"]: Mark the pronoun as following sandhi @ if it appears at the end and there is a verb before it	word level

These sandhi rules work on 4 different levels: the syllable, the word, the part of speech, and the sentence pattern.

The algorithm described above is mainly based on a) tone sandhi rules proposed by linguists; b) rules induced from the training data; and c) our intuition as native-speaking observers of sandhi phenomena. We also consulted d) the word segmentation results of the CKIP (examining its POS tagging output) and e) the Taiwanese concordancer system (to check the sandhi phenomena of certain words) when we met some questions.

It should be noted that some of the sandhi rules proposed by linguists deal with specific contexts and thus cannot be broadly applied; some others carry exceptions. There is, therefore, some difficulty in converting these rules into an algorithm. So, besides (a), we also formulated some rules from (b) and (c) by analyzing errors in the training data output. In principle sandhi rules are formulated to be applicable to "most situations" -- *i.e*. an accuracy rate of over 75% on corpus data. Once applied, the new rules may affect the original rules, so (d) and (e) are our important references in deciding whether or not to apply the new rules.

Some rules have priority. Subsequent rules can supersede previous ones. As an example, rule 9 (pronoun rule) can supersede rule 3 (*of* rule). At the level of sentence pattern, rule 19.4.2 can supersede 19.4.1 as in the following example:

(9) "*Lí ē khì kok-gōa bē*" [你會不會去國外? "*Will you go abroad or not*"]:
the last *bē* [不會, "*will not*"] is marked as neutral sandhi, whereas
"*Lí ē khì kok-gōa iah-sī bē*" [你會不會去國外? "*Will you go abroad or not*"]:
the last *bē* is marked as basic tone.

Moreover, because of the uncertainty in tagging POS, some rules are set to apply only when there is no ambiguity, while some other rules are applied to any matching POSs.

We currently employ 20 rules and expect to refine them or append new ones.

The following training data represents a pre-tagged source (Chinese and English translations added):

(10) Chhin-chhiūⁿ [像] án-ni[這樣] lâi[來] kóng[說], chāi[在] lán[我們] Tâi-ôan[台灣] kīn-kīn[近近] chit-tiap-á-kú[一下子] ê[的] kang-hu[工夫], ài[要] soaⁿ[山] chiū[就] ū[有] soaⁿ[山], ài[要] hái[海] chiū[就] ū[有] hái[海], beh[要] jo̍ah[熱] chiū[就] ū[有] jo̍ah[熱], kôaⁿ[冷] chiū[就] ū[有] kôaⁿ[冷]. Só-í[所以] thang[可以] kóng[說] Tâi-ôan[台灣] sī[是] chit-ê[一個] sió[小] Tang-iûⁿ[東洋]. Lán[我們] Tâi-ôan[台灣] ū[有] chit-khóan[這種] thian-jiân[天然] ê[的] hó-kéng[好景] ,hó[好] khì-hāu[氣候], chiong-lâi[將來] nā-sī[若是] ēng-sim[用心] ke[加] lâng[人] ê[的] kang-hu[工夫] tōa-tōa[大大] lâi[來] chéng-tùn[整頓], tek-khak[的確] ē[會] chiâⁿ-chò[成爲] Tang-iûⁿ[東洋] ê[的] tōa[大] kong-hn̂g [公園], hō [讓] Tang-iûⁿ[東洋] ê[的] lâng[人] chi̍p-óa[靠近] lâi[來] hióng-hok[享福] an- lo̍k [安樂].

--- "*Cha̍p-hāng kóan-kiàn*" [十項管見]
by Chhòa Pôe-hóe[蔡培火], 1925

Take this as an example. Here in Taiwan, reachable with a minimum of effort, you have mountains for those who like mountains, seas for those who like seas, hot weather for those who like heat, and cold weather for those who like cold. So you can say Taiwan is a miniature East. Given Taiwan's natural sceneries and fair climate, if you'd take care to rebuild it, it'd surely become the Great Park of the East, where Easterners go for rest or fun.

---"*Ten Humble Opinions*"
by Chhòa Pôe-hóe, 1925

After POS tagging and applying the sandhi rules:

(11) Chhin -chhiūⁿ(D) án-ni#(D;N) lâi(D;V) kóng#(V), chāi(D;A;P;V) lán(R) Tâi-ôan#(N) kīn-kīn(A) chit-tiap&-á-kú#(N) ê(M) kang-hu#(A;N), ài(D;V) soaⁿ# (N) chiū(D) ū(D;P;V) soaⁿ#(N), ài(D;V) hái#(N) chiū(D) ū(D;P;V) hái#(N), beh$(D) jo̍ah#(A) chiū(D) ū(D;P;V) jo̍ah#(A), kôaⁿ#(A) chiū(D) ū(D;P;V) kôaⁿ#(A). Só-í(C) thang(D) kóng(V) Tâi-ôan#(N) sī(D;V) chit-ê#(N) sió(D;A) Tang-iûⁿ#(N). Lán(R) Tâi-ôan#(N) ū(D;P;V) chit-khóan#(D;N) thian-jiân#(A) ê(M) hó-kéng#(N), hó(D;A;C;V) khì-hāu#(N), chiong-lâi#(S) nā-sī(C) ēng-sim#(N) ke(V) lâng#(N) ê(M) kang-hu#(A;N) tōa-tōa(A) lâi(D;V) chéng-tùn#(V), tek-khak(D) ē(D;V) chiâⁿ-chò(V) Tang-iûⁿ#(N) ê(M) tōa(A;N) kong-hn̂g#(N), hō(D;P;V) Tang-iûⁿ#(N) ê(M) lâng#(N) chi̍p-óa(V) lâi(D;V) hióng-hok#(A) an-lo̍k#(A).

The letters within the parentheses are the POSs. Incorrectly processed syllables are boxed.

3. Results

3.1 Evaluation

Three authors of this paper, who are skilled native speakers familiar with written Taiwanese, evaluated the correctness of the output. Note that in certain contexts more than one sandhi result is acceptable, and depending on discourse considerations some speakers may opt for one sandhi result over others. For example, "hō lí" [給你,"*give you*"] can be read as (3,2) (normal sandhi) or (7,7) (following sandhi). Telephone number is another example: the number may be divided into various groups, each group containing 2, 3 or 4 digits.

3.2 Preliminary Results

There are 614 syllables of training data, 16 errors, giving an accuracy rate of 97.39%. There are 962 syllables of test data with 106 errors, or an accuracy rate of 88.98%. *Table 7* shows the number of errors and accuracy rate for each paragraph.

Table 7. Number of errors and accuracy rate for each paragraph

training data					test data				
para. id.	no. of words	no. of syllables	no. of errors	accuracy rate	para. id.	no. of words	no. of syllables	no. of errors	accuracy rate
1	27	30	1	96.67%	1	130	184	16	91.30%
2	42	54	0	100.00%	2	56	85	12	85.88%
3	44	70	0	100.00%	3	53	84	13	84.52%
4	33	52	0	100.00%	4	96	143	16	88.81%
5	38	51	4	92.16%	5	66	97	10	89.69%
6	85	110	4	96.36%	6	63	86	9	89.53%
7	97	144	6	95.83%	7	32	43	3	93.02%
8	72	103	1	99.03%	8	38	58	2	96.55%
					9	122	182	25	86.26%
total	438	614	16	97.39%	total	656	962	106	88.98%

Table 8 shows the numbers of each rule applied in training data and test data respectively. We count the number of affected syllables, accurately affected syllables, and accuracy rate of each rule. Note that rules 5 & 6 don't seem work well because of POS ambiguities, rule 7 does not affect any syllables because the word whose POS is G (postposition) also has other POSs, rule 14 does not affect any syllables because there are no triplicated words in our training and

test data.

Table 8. Affected and accurately affected syllables of each rule

	training Data			test data		
rule id.	affected syllables	accurately affected	accuracy rate	affected syllables	accurately affected	accuracy rate
1	614	411	66.94%	962	662	68.81%
2	74	68	91.89%	112	105	93.75%
3	32	24	75.00%	38	26	68.42%
4	3	3	100.00%	13	7	53.85%
5	65	57	87.69%	129	90	69.77%
6	4	3	75.00%	4	3	75.00%
7	0	0	--	0	0	--
8	5	5	100.00%	3	3	100.00%
9	29	29	100.00%	25	25	100.00%
10	5	5	100.00%	0	0	--
11	3	3	100.00%	8	8	100.00%
12	8	8	100.00%	11	11	100.00%
13	2	2	100.00%	6	5	83.33%
14	0	0	--	0	0	--
15	8	8	100.00%	6	5	83.33%
16	13	13	100.00%	4	4	100.00%
17	3	3	100.00%	2	2	100.00%
18	9	9	100.00%	3	3	100.00%
19	0	0	--	6	6	100.00%
20	2	2	100.00%	0	0	--

Every syllable is affected by at least one rule, and is affected by four rules at most. *Table 9* shows the number of dominant rule, accurate dominant rule, and accuracy rate.

Table 9. Number of dominant rule, accurate dominate rule and accuracy rate

	training data			test data		
rule id.	no. of dominant rule	no. of accurate dominant rule	accuracy rate	no. of dominant rule	no. of accurate dominant rule	accuracy rate
1	381	371	97.38%	616	568	92.21%
2	62	62	100.00%	104	99	95.19%

Table 9. Number of dominant rule, accurate dominate rule and accuracy rate

3	24	23	95.83%	34	26	76.47%
4	2	2	100.00%	6	3	50.00%
5	62	57	91.94%	126	87	69.05%
6	2	2	100.00%	4	3	75.00%
7	0	0	--	0	0	--
8	4	4	100.00%	3	3	100.00%
9	27	27	100.00%	25	25	100.00%
10	5	5	100.00%	0	0	--
11	3	3	100.00%	8	8	100.00%
12	7	7	100.00%	11	11	100.00%
13	2	2	100.00%	6	5	83.33%
14	0	0	--	0	0	--
15	6	6	100.00%	5	4	80.00%
16	13	13	100.00%	3	3	100.00%
17	3	3	100.00%	2	2	100.00%
18	9	9	100.00%	3	3	100.00%
19	0	0	--	6	6	100.00%
20	2	2	100.00%	0	0	--

After examination, we find that we can add 7 additional rules without too much effort; in this way, we were able to fix 20 errors and achieve a 91.06% accuracy rate. *Table 10* shows the additional rules in order to fix 20 errors in test data.

Table 10. Additional rules to obtain higher accuracy rate

Rules	Number of corrections in test data
Word suffix "V-tit" (adverbialize the word whose POS is verb)	5
Double sandhi of "khah" [更, "*more*"]	4
Re-process the syllable preceding "ê" [個, *a numerary adjunct*] when the preceding word is a number or "chit/hit/pat̍" [這/那/別, "*this/that/other*"]	4
"V-jip-lâi" [V 進來, "*V-in*"]: mark as neutral sandhi when sentence-final	3
Word "hut-jiân" [忽然, "*suddenly*"]: mark the last syllable as basic tone in any case	2
Word "kīn-lâi" [近來, "*recently*"]: mark the last syllable as basic tone in any case	1
Word suffix "N-nih"[N 裡, "*inside N*"]: mark as neutral sandhi	1

4. Analysis of Errors and Relevant Issues

Some of the problems we encountered may be taken into account in the future.

4.1 POS

In our investigation, we use the POS set for Chinese. Whether this approach is suitable for Taiwanese is a debatable linguistic question requiring further investigation. Although a few studies of the POS of Taiwanese are available from as early as the 1930s, currently these data have yet to be digitized, and will need to be reviewed by linguists to ensure that they are suitable for dealing with the sandhi problem.

4.2 Word Segmentation Standard and Dictionary

[Tseng 1997] proposes a standard for Taiwanese word segmentation. Unfortunately discussion is lagging. Should a working word segmentation standard emerge, we would also need a dictionary conforming to that standard.

4.3 Standardization of Written Taiwanese

Historically, the use of Han script to represent Taiwanese has suffered from a high degree of idiosyncrasy in character choice. For documents written in Latin script, most of the differences attributed to dialects can be reconciled by referencing existing dictionaries. Orthographic inconsistency in the use of hyphen is more problematic, as it could affect the result of sandhi processing. Manual standardization of hyphen placement is hardly a solution.

4.4 Tone Sandhi Problems Not Solvable by POS Order

We have encountered certain sandhi problems that likely cannot be solved solely by inspecting the POS order. These include verb-verb (VV) and noun-noun (NN) patterns:

(12) a. "phah-pià^{n}(V) chò(V) khang-khòe(khè) (N)" (2,2,2,7,3)
[努力做工作, "*do work hard*"]
b. "kiảh-bák(V) khòa^{n}(V) hn̂g(N)" (3,8,2,5)
[舉目看園, "*lift eyes and see plowland*"]

(12) is an example of a VV pattern. The final syllable of the first verb in (a) should be marked as sandhi tone, while in (b) it should be marked as basic tone. Differences in the internal structure of these two initial verbs suggest some clues for handling this problem. However, its implementation awaits further research.

(13) a. "tiān-sī kóng-kò" [電視(的)廣告, "*TV advertisement*"]
b. "thâng-thōa chiáu-chiah" [昆蟲(、)小鳥, "*insects and birds*"]

(13) is an example of a NN pattern. Again, the final syllable of the first noun in (a) should be marked as sandhi tone, while in (b) it should be marked as basic tone. Currently, we see no solution to this.

4.5 Error Conditions

Error conditions, including those discussed in the previous sections, are listed below with possible in *Table 11* :

Table 11. Error conditions and possible solutions

Errors	Possible Solutions
(a) Due to dictionary limitation (not having the words)	Increase entries
(b) Due to lack of punctuation marks	Pre-process, but this is very difficult
(c) Due to wrong POS because of homonymy	Apply semantic knowledge
(d) Due to indeterminate POS or multiple candidates	Tagging disambiguity
(e) Caused by inconsistent orthography in hyphen segmentation	Pre-process the sources or deal with the procedures of adding or removing hyphens automatically
(f) Due to incomplete sandhi rule set	Refine the sandhi rules while avoiding side effects
(g) Associated with quantitative words;	Add DM rules
(h) Associated with proper nouns	Detect proper nouns
(i) Associated with sentence pattern	Add sandhi rules for sentence patterns
(j) Possibly other sources of error yet to be identified	

5. Future Work

A three-year-old child native speaker can process tone sandhi correctly and apparently without effort, yet it is rather more difficult for a computer system to do so. Clearly, a practical system for sandhi processing of Taiwanese remains out-of-reach and a cause for future research. Some suggestions for future work:

(a) Solicit assistance from linguists. It is hoped that linguistics will define a standard for part-of-speech analysis and word segmentation, and that a dictionary conforming to such a standard will be built.

(b) Improve word segmentation, especially the processing of morphology, quantitative words, and proper nouns.

(c) Improve the processing of POS tags to account for ambiguity.

(d) Improve the dictionary's part-of-speech data, such as making use of Embree's POS analysis [Embree 1984].

(e) Improve the sandhi rules.

(f) Find alternative ways of modeling sandhi processing, such as Cheng's grammar template model. [Cheng 2002]

Acknowledgements

This work is supported by National Museum of Taiwanese Literature in Taiwan. We also thank the anonymous reviewers for their constructive opinions.

References

Cheng, R., *Taiwanese and Mandarin Structures and Their Developmental Trends in Taiwan Book I : Taiwanese Phonology and Morphology*, Yuan-liou Publishing Co., 1997.

Cheng, R., "Tone Sandhi on the Grammar Template--Cognition and Testing," *Proceeding of 2002 International Conference on Teaching and Researching of Taiwanese Romanization*, 2002, pp. I1-I19.

Embree, B. L.M.A., *A Dictionary of Southern Min*. Taipei Language Institute, 1984.

Iunn, U.-G., "Taiwanese-Chinese On-line Dictionary -- Discussion of Building Technique and its Utilization," *Proceeding of 3rd International Conference on Internet Chinese Education*, 2003, pp. 132-141.

Iunn, U.-G. and H.-K. Tiunn, "Review and Analysis of Taiwan Ho-lo Language non-Han Character Spelling Symbols," *Proceedings of 1st Conference on the Regeneration and Rebuild of Taiwan Mother Tongue Culture*, 1999, pp. 62-76.

Iunn, U.-G. and H. H.Tan-Tenn, "A Survey of Media and Data Processing Development for Written Taiwanese," Accepted by *International Journal of the Sociology of Language, Special Issues on Taiwanese*.

Liang, M.-S., J.-C. Yang, Y.-C. Chiang, and R.-Y. Lyu, "A Taiwanese Text-to-Speech System with Applications to Language Learning," *Proceedings of the 4th IEEE International Conference on Advanced Learning Technologies*, 2004, pp. 91-95.

Lin, C.-J. and H.-H. Chen, "A Mandarin to Taiwanese Min Nan Machine Translation System with Speech Synthesis of Taiwanese Min Nan," *International Journal of Computational Linguistics and Chinese Language Processing*, 4(1), 1999, pp. 59-84.

Lu, G.-C., *The Study of Minnan Vocabulary in Taiwan*. SMC Publishing Inc., 1999.

Tiunn, J.-H. (Chang, Y.-H.). *Principles of POJ or the Taiwanese Orthography: An Introduction to Its Sound-Symbol Correspondences and Related Issues*, Crane Publishing Co., 2001.

Tseng, C.-C., "The Discussion of Taiwanese Word Segmentation Principles," *The Project Report for the Collecting, Cataloging and Select Editing of Taiwanese Liturature Publications*, pp. 47-73, Council for Culture Affairs, 1997.

Online Resources

Chinese On-line Word Segmentation System, http://ckipsvr.iis.sinica.edu.tw

Digital Archive Database for Written Taiwanese, http://iug.csie.dahan.edu.tw/nmtl/dadwt

Taiwanese Concordancer System, http://iug.csie.dahan.edu.tw/TG/concordance/form.asp

Taiwanese Package, http://www.phahng.idv.tw or http://taigu.fhl.net/TP/

Computational Linguistics and Chinese Language Processing
Vol. 12, No. 4, December 2007, pp. 371-390

A System Framework for Integrated Synthesis of Mandarin, Min-Nan, and Hakka Speech

Hung-Yan Gu*, Yan-Zuo Zhou*, and Huang-Liang Liau*

Abstract

In this paper, a framework for integrated synthesis of Mandarin, Min-nan, and Hakka speech is proposed. To show its feasibility, an initial integrated system has been built as well. Through integration, a model only trained with Min-nan sentences is used to generate pitch-contours for all three languages, same rules are used to generate syllable duration and amplitude values, and the same program module implementing the method, TIPW, is used to synthesize the three languages' speech waveforms. Also, in this system, each syllable of a language has just one recorded signal waveform, *i.e*. no chance of unit selection. Under such a restricted situation, the synthetic speech signals still have noticeable naturalness level and signal clarity.

Keywords: Speech Synthesis, Pitch Contour Model, TIPW, Time Axis Warping.

1. Introduction

There are many languages in Taiwan, including Mandarin, Min-nan, Hakka, and others spoken by smaller population groups. Mandarin has been more extensively studied than the other languages because it is the official language. However, the successful construction of a synthesis model or system for Mandarin does not imply that the same modeling method can be directly applied to another language. Developing speech synthesis systems for other languages is strongly desired because Mandarin is not the mother tongue of most people in Taiwan, and all languages except Mandarin face the crisis of disappearance.

If systems developed or speech data collected previously can only be used for Mandarin, then further resources (effort and money) are inevitably needed to study other languages. Such a situation will become more severe if a corpus-based approach [Chou 1999; Chu *et al.* 2003] is adopted. In addition, there will be inconsistency in prosody and timbre among

* Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, 43 Keelung Rd., Sec. 4, Taipei, Taiwan
E-mail: {guhy, M9315058, M9215001}@mail.ntust.edu.tw

[Received October 31, 2006; Revised April 17, 2007; Accepted April 25, 2007]

independently developed speech synthesis systems for different languages. Therefore, a better approach is to construct a more generalized system that can synthesize not only Mandarin but also Min-nan and Hakka speech. Such an approach, if successfully realized, can not only save resources but also obtain much higher consistency among the synthesized speech for different languages. Another advantage is that an improvement, when made to a system component, can immediately benefit all the languages supported.

Mandarin, Min-nan, and Hakka are all syllable prominent languages, and are all tonal languages. Hakka has many accents found in users in Taiwan, of which "four-country" and "sea-land" are the primary ones. If not specified, the sea-land accent is the default accent representing Hakka in this paper. This is because it has more lexical tone and more unique syllables than the four-country accent has, and the authors believe the speech signal of the sea-land accent is more difficult to synthesize. As to the number of different syllables (not distinguishing lexical tones), Mandarin has 405, Min-nan has 833, and Hakka has 783 [Yu 1999]. The languages also vary in the numbers of different lexical tones, being 5, 7, and 7, respectively, for Mandarin, Min-nan, and Hakka. Since the numbers, 405, 833, and 783, are not large, syllable is commonly chosen as the speech unit for synthesis processing. Actually, in this system, each syllable (tone not distinguished) of a language has only one recorded utterance. That is, no extra units are available to do unit selection, and each syllable's waveform must be manipulated to synthesize speech signals with different required prosodic characteristics.

Note that the focus of this study is in the system framework of an integrated speech synthesis system for the three languages. To show the feasibility of the proposed framework, a workable integrated synthesis system is built. This system is just in its initial phase. Therefore, there will be many unsolved problems in the details. Most of these problems belong to text analysis since signal synthesis is the major concern in this research while text analysis is just a minor concern. In general, a speech synthesis system can be divided into three subsystems, *i.e.*, (a) text analysis, (b) prosodic parameter generation, and (c) speech waveform synthesis [Shih *et al.* 1996; Wang 1998]. The framework for the integrated synthesis system is also divided into such subsystems. The main processing flow of this framework is shown in Figure 1. The first two processing blocks are for text analysis, the middle two blocks are for prosodic parameter generation, and the last two blocks are for signal waveform synthesis. The synthesis system built is an integrated system, not a bundle of three independent systems for the three languages. This is because the program modules in the three subsystems are all shared in synthesizing the three languages' speech. For example, the model for pitch-contour parameter generation is shared (or adapted) between Mandarin and Hakka, although it is originally trained with Min-nan sentences. The explanations for why the program modules can be shared are given in the following sections.

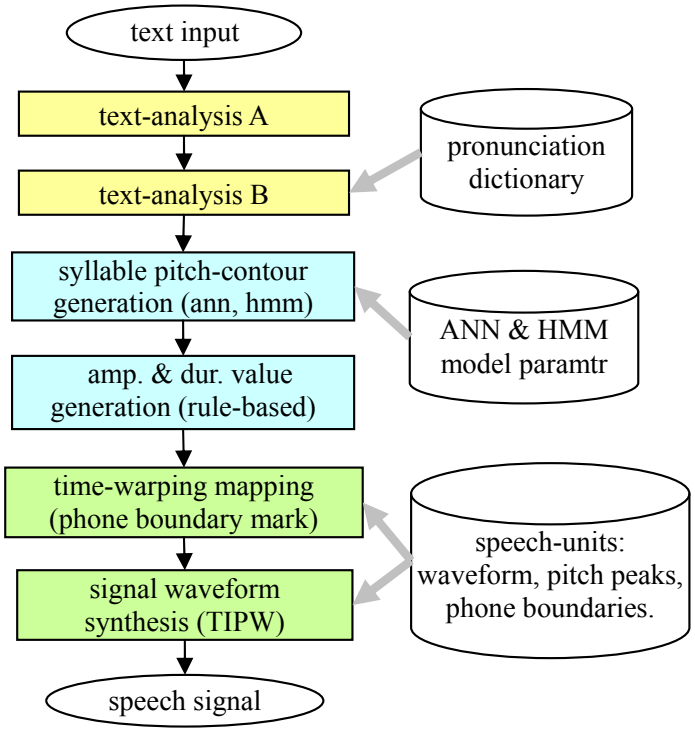


Figure 1. Main processing flow.

In the subsystem of text analysis, the first block in Figure 1, "Text Analysis A", parses the input text to recognize tags and slice the text string into a sequence of Chinese-character or alphanumeric-syllable tokens. Then, each Chinese-character token is tried in the second block (Text Analysis B), to check if it can be looked up in a pronunciation dictionary in order to determine the comprising character's pronunciation syllable. For the subsystem of prosodic parameter generation, the pitch-contour parameters of a syllable are determined by a mixed model of ANN (artificial neural network) [Chen *et al.* 1998; Lee *et al.* 1991] and SPC-HMM (syllable pitch-contour hidden Markov model) [Gu *et al.* 2000] in the third block of Figure 1. As to the parameters, amplitude and duration, their values are determined with a rule-based method [Chiou *et al.* 1991; Shiu 1996] in the fourth block of Figure 1. For the subsystem of signal waveform synthesis, a piece-wise linear time-warping function is first constructed in the fifth block of Figure 1. Then, the method of TIPW (time-proportioned interpolation of pitch waveform) [Gu *et al.* 1998] is used in the sixth block to synthesize speech waveforms. TIPW is an improved variant of PSOLA [Modulines 1990]. To show that the integrated system has noticeable performance in naturalness and signal clarity, the authors have set up a web page, http://guhy.csie.ntust.edu.tw/hmtts/, to demonstrate synthetic speech for the three languages. However, for the purpose of online testing, http://guhy.csie.ntust.edu.tw/hmtts/speak.html is preferable.

2. Text Analysis

In Min-nan and Hakka, there are still many spoken words whose corresponding ideographic words are not known. Therefore, the authors made a decision that, in the input text, Chinese characters may be interleaved with syllables spelled in alphanumeric symbols. For example, "cit-4 tou-5 人" is a Min-nan word whose first two syllables are spelled in alphanumeric symbols. This decision implies that the input text must first be parsed into a sequence of Chinese-character and alphanumeric-syllable tokens. For example, "今天 mai-3 ki-3" is parsed into "今天", "mai-3", and "ki-3". The number at the end of a syllable indicates the lexical tone of the syllable. This parsing processing is executed in the first block of Figure 1.

In addition, the authors have defined several kinds of tags to help carry some necessary controlling information. For example, the tag, "@>2", may be placed between two sentences to command that the sentences behind the tag will be synthesized to Hakka speech until another language-selection tag is encountered. Such a language-selection tag is needed because it is intended that the sentences of an article may be alternatively synthesized to different languages' speech. Another kind of tag is "@>dxxx". This tag may also be placed between two sentences to change the speaking rate of the sentences behind it. The part, "xxx", in the tag represents three decimal digits to specify how many milliseconds on average a syllable will be synthesized to. In addition to the two tags explained, several other kinds of tags are also defined. The details are listed in Table 1. The parsing of such tags is also executed in the first block of Figure 1.

Table 1. Tags and their meanings.

Tag symbol	Explanation
@>x	language selection, x may be 0, 1, or 2. 0: Mandarin, 1: Min-nan, 2: Hakka
@>dxxx	speaking rate, syllable average duration in xxx milliseconds.
@>txxx	average tone height, in xxx Hz.
@>vxxx	vocal track extended (or shrunken) to xxx percents of original length.
<, >	word-constructing tag, *e.g.*, <cit-4 tou-5>
*	breath-break tag

After an input sentence is parsed into a sequence of tokens, the pronunciation syllables for each Chinese character token are determined in the second block of Figure 1. According to the language-selection tag, the corresponding pronunciation dictionaries are consulted to check if the prefix part of a token can be found in the dictionaries. A dictionary consisting of longer words is tried before a dictionary consisting of shorter words. Currently, the authors have collected 55,000, 12,000, and 19,000 multi-syllabic words, respectively, for Mandarin,

Min-nan, and Hakka. Note that input text is usually composed in Mandarin written words. Therefore, use of the dictionary plays a role of word translation. For example, "今天" (today) in Mandarin is translated as "今仔日", in Min-nan, which is pronounced as, "gin-1 a-2 rit-8". Another example, "筷子" (chopstick) in Mandarin is translated to "箸" which is pronounced as, "di-7". These examples also show that the words obtained after translation may have longer or shorter lengths.

After a word is found in a dictionary or a block of syllables bounded with the tags, "<" and ">", is parsed out, one knows the boundaries of a word and its syllabic composition. Then, tone-sandhi rules for the currently selected language can be applied to the compositional syllables of the word. This is executed in the second block of Figure 1. Note that different languages have very different tone-sandhi rules. For example, in a word of Mandarin, if two adjacent syllables are both of the third tone, then the former one must have its tone changed to the second tone. As another example, consider the tone-sandhi rule applied to a word of Min-nan that every syllable of a word except the final one must have its tone changed to its inflected tone.

3. Prosodic Parameter Generation

The prosodic parameters of a syllable include pitch-contour, duration, amplitude, and leading pause. The generation of prosodic parameter values plays a very important role because it determines the level of naturalness of synthesized speech. Therefore, much effort has been devoted to investigate models (or methods) for generating prosodic parameter values [Chen *et al.* 1998; Gu *et al.* 2000; Lee *et al.* 1993; Wu *et al.* 2001; Yu *et al.* 2002].

Among these prosodic parameters, pitch-contour is the most important one for obtaining a higher naturalness level. Therefore, the authors have spent considerable effort in investigating different kinds of models, HMM [Gu *et al.* 2000], ANN, and a mixed model of both [Gu *et al.* 2005b]. In the third block of Figure 1, a mixed model of HMM and ANN is used to generate pitch-contours. Here, model mixing means taking a weighted sum of two pitch-contours generated respectively by HMM and ANN. Note that, in this study, pitch-contour models, HHM and ANN, are both trained with Min-nan spoken sentences. Then, through tone mapping, the Min-nan trained and mixed model is adapted to generate pitch-contours for Hakka and Mandarin. By such a sharing of pitch-contour model, the effort in training other languages' pitch-contour models can be saved.

In contrast to pitch-contour, duration and amplitude are thought to be minor factors for naturalness. Hence, only a rule-based method is used in the fourth block of Figure 1 to generate their values [Chiou *et al.* 1991; Shiu 1996]. The authors program three sets of rules for the three layers, syllable layer, word layer, and breath-group layer. In the syllable layer, a syllable containing different vowel phonemes, /a/, /i/, /u/, /e/, or /o/, is assigned different

amplitude values, 0dB, -4dB, -3dB, -2dB, or -1dB. In the word layer, the first syllable of a word is emphasized 0.5 dB in amplitude. Finally, in the breath-group layer, the first two syllables of a group are emphasized 1dB and 0.5dB respectively. In addition, the last two syllables of the last breath-group of a sentence are deemphasized 0.5dB and 1dB respectively. By interaction of these rules in the three layers, the generated amplitude and duration values appear to have some randomness, and can present a certain level of naturalness.

3.1 Syllable Pitch Contour HMM

A syllable at the beginning of a sentence is usually uttered with higher pitch than one at the end, *i.e*., the phenomenon of declining. With respect to this phenomenon, the authors imagine that there are three prosodic states corresponding to sentence-initial, sentence-middle, and sentence-final. However, how to assign a sentence's syllables to these states is not explicitly known. Therefore, the authors imagine these prosodic states are hidden and will simulate them by the hidden states of a left-to-right hidden Markov model [Rabiner *et al.* 1993]. Besides the influence of prosodic states, the lexical tones of a syllable and its adjacent syllables also have strong influences. Therefore, the authors take into account the lexical-tones of a syllable and its adjacent syllables, and call such an HMM as syllable pitch-contour HMM (SPC-HMM).

The height and shape of a syllable's pitch-contour are mainly influenced by the lexical tones of the syllable and its immediately adjacent syllables. Therefore, the authors decide to combine the t-th syllable's lexical tone and pitch-contour VQ (vector quantization) code with its left and right adjacent syllables' lexical tones to define the t-th observation symbol, O_t, as

$$\begin{aligned} O_t &= 392 \cdot X_{t-1} + 56 \cdot X_t + 8 \cdot X_{t+1} + V_t, \\ & 0 \le X_t \le 6,\ 0 \le V_t \le 7. \end{aligned} \tag{1}$$

where X_t is the lexical-tone number of the t-th syllable, and V_t is the pitch-contour VQ code of the t-th syllable in a training sentence. Actually, the number, X_t, is indirectly obtained, *i.e*. lexical-tone number eight is mapped to six beforehand, and then the lexical-tone number is decreased by one. In Equation (1), the number, eight, is multiplied because there are eight codewords in each tone's pitch-contour VQ codebook. The numbers, 56 and 392, may be viewed as 7×8 and 7×7×8, respectively, and 7 is the number of different lexical tones in Min-nan and 8 is the number of code-words in a VQ codebook. When t=1, *i.e*. staying at the first syllable of a training sentence, X_{t-1} is undefined. In this case, the definition of O_t is modified to $7\times7\times7\times8 + 56\ X_t + 8\ X_{t+1} + V_t$. Similarly, the definition of O_t for the last syllable of a sentence must also be modified [Gu *et al.* 2000].

Before VQ encoding, the pitch-contour of each syllable from a training sentences is first time normalized and then pitch-height normalized [Gu *et al.* 2000]. Time normalization means

placing 16 measuring points equally spaced in time. Then, a pitch-contour is represented as a vector of 16 frequency values (in log Hz scale), called a frequency vector. After time normalization, these frequency vectors must be normalized in pitch-height to eliminate the influence of the speaker's mood at the time of recording. Totally, the authors have recorded 643 Min-nan training sentences that are comprised of 3,696 syllables.

Next, consider the generating of pitch-contours by using SPC-HMM. When a sentence is input, it will be analyzed first by the textual analysis components. Hence, its pronunciation-syllable sequence is available. For example, for the short sentence, "我來啊" (I have come), of Min-nan, its corresponding syllable sequence is "qua-1 lai-5 a-7". Then, one can encode the three adjacent syllables' lexical tones partially (because VQ code, V_t, is not known yet) according to Equation (1). Since each lexical tone has 8 codewords in its pitch-contour VQ codebook, each syllable of the sentence has 8 possible encoded observation symbols corresponding to it. For example, for the second syllable "lai-5", its possible encoded observation symbols are 392(1-1)+56(5-1)+8(7-1)+V_t, *i.e.* the value range from 264 to 271 since the value of V_t is not determined yet. Therefore, in the synthesis phase (or testing phase), besides the time (syllable index within a sentence) and state (prosodic-state index) axes, a third axis to index the 8 possible observation-symbol candidates, must be added. Then, the conventional two-dimensional (time and state) DP (dynamic programming) algorithm for speech recognition is extended to a three-dimensional DP algorithm and used to search the most probable path [Gu *et al.* 2000]. The main part of the extended algorithm is shown in Equation (2),

$$\delta_t(n,k) = \left[\max_{n-1 \le i \le n} \max_{0 \le j \le 7} \delta_{t-1}(i,j) \cdot a_{i,n} \right] \cdot b_n(O_t^k), \quad 0 \le n \le 2,\ 0 \le k \le 7, \tag{2}$$

where O_t^k represents the k-th possibly encoded observation symbol at time t, n and i are state indices, $a_{i,n}$ is state-transition probability, $b_n(\bullet)$ is symbol-observing probability at state n, and $\delta_t(n,k)$ is the largest obtainable probability of a best path that stays at state n and selects the k-th observation symbol at time t. According to the best path found, the state value and k value of O_t^k at each time point, t, can then be determined. Accordingly, the pitch-contour VQ code, V_t, for the t-th syllable of the sentence is set to the value of k determined at time t.

3.2 Syllable Pitch Contour ANN

The architecture of the artificial neural network used here is shown in Figure 2. It is designed to be a recurrent type ANN in order to have the prosodic state kept internally. The input layer of the ANN has 8 ports to receive contextual parameters. For the hidden and recurrent hidden layers, the numbers of nodes are both set to be 30, according to experiment results. After a syllable's contextual parameters are input and processed, a pitch contour represented as a 16

dimensional frequency vector is output in the output layer. This frequency vector can be interpreted as a sequence of 16 frequency values along a pitch-contour.

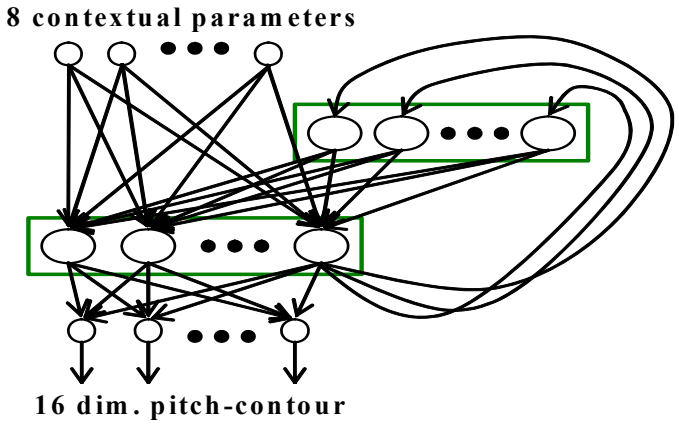


Figure 2. The architecture of the ANN studied here.

Here, the contextual parameters, *i.e*. the inputs to the ANN, are appropriately selected to provide essential contextual information and to lower the quantity of required training sentences. In detail, the contextual parameters are as listed in Table 2. As there are seven lexical tones in Min-nan, 3 bits are enough to represent them. The numbers of different syllable initials and finals are 18 and 61, respectively. Hence, 5 and 6 bits, respectively, are used to represent current syllable's initial and final types. As to the parameter of the previous syllable's final, the authors first group the 61 possible finals into 12 classes, using only 4 bits to represent the 12 final classes. Similarly, the authors first group the 18 possible initials into 6 classes, and use only 3 bits to represent the 6 initial classes for the next syllable's initial. Grouping is made here because the quantity of recorded training sentences is not large enough to let the ANN learn the influences of the detailed combinations of current syllable and previous (or next) syllable. Syllable initial and final classes grouped here are detailed in Table 3 and 4, respectively. The last item in Table 2, the time-progress index, is intended to carry timing information. If the current syllable is the k-th syllable of a sentence of N syllables in length, then the value of time-progress index is set to the floating-point number k/N.

Table 2. Contextual parameters.

Items	Tone of previous syllable	Final class of previous syllable	Tone of current syllable	Initial of current syllable	Final of current syllable	Tone of next syllable	Initial class of next syllable	Time progress index
Bits	3	4	3	5	6	3	3	void

Table 3. Syllable-initial classes. (General Phonetic Symbol System)

Classes	1	2	3	4	5	6
Initials	(null), m, n, l, r, ng, q, v	h, s	b, d, g	z	c	p, t, k

Table 4. Syllable-final classes. (General Phonetic Symbol System)

Classes	Finals	Classes	Finals
1	(null)	7	-i, -u, -ui, -iu
2	-a, -ia, -ua	8	-ing, -eng, -in, -un, -en
3	-o, -io, -ior	9	-ang, -iang, -uang, -ong, -iong, -an, -uan
4	-er, -ier	10	-am, -iam, -im, -om
5	-e, -ue	11	-ah, -eh, -ih, -oh, -uh, -auh, -erh, -iah, -ierh, -ioh, -uah, -ueh
6	-ai, -uai, -au, -iau	12	-ap, -iap, -ip, -op, -at, -et, -it, -uat, -ut, -ak, -iak, -ik, -iok, -ok

3.3 Adaptation of Pitch Contour Model

Here, the pitch-contour model trained with Min-nan sentences is used as the working model. This working model can be adapted in a way to generate pitch contours for a target (Mandarin or Hakka) language's sentences. In detail, a lexical-tone sequence, X_1X_2, …, X_n, extracted from a target language's sentence is first mapped to a lexical tone sequence, Y_1Y_2, …, Y_n, for the working language. Then, the mapped lexical tones are used instead as the input for the working model. The pitch-contours, R_1R_2, …, R_n, generated by the working model are treated as the output of the adapted model for the sequence, X_1X_2, …, X_n.

The reasons why this adaptation method may work are explained in detail in [Gu *et al.* 2005a]. In brief, slight differences in frequency-height or boundary-part shape between two pitch-contour curves need not be worried about for correct recognition of the carried lexical tone. The authors also think it is reasonable to approximate a pitch-contour curve in a target language with a curve-shape class trained in the working language that is of similar shape in the central part. Note that each lexical tone of the working language is usually trained to have several representative curve-shape classes, *e.g.* 8 code-words in each lexical tone's VQ codebook. Hence, for a pitch-contour curve from a target language, one can select from the curve-shape classes trained for the mapped lexical tone to pick out the curve that is most similar. Then, the possible decrease in naturalness due to differences in frequency-height or boundary-part shape can be minimized.

Note that some syllable initials and finals of Mandarin (*e.g.*, /yu/) and Hakka (*e.g.*, /oi, eu/) are not found in Min-nan. With respect to this, it may make one suspect if the adaptation method can indeed work. However, in SPC-HMM, the definition of observation symbol in equation (1) does not include syllable initials and finals. This indicates that pitch-contour is only insignificantly influenced by syllable initials and finals according to previous studies [Gu *et al.* 2000; Gu *et al.* 2005b]. Although the information of syllable initial and final is used in the pitch-contour ANN, the authors still think that the factors of syllable initial and final are insignificant according to previous experiments for evaluating classification methods of Min-nan initials and finals [Gu *et al.* 2005b]. Anyway, to solve the problem of mismatched initials and finals between the two languages, the authors let the program automatically select a similar one (more same letters in spelling) to replace a finial or initial not found in Min-nan. For example, /yu/ is replaced with /u/, and /oi, eu/ are replaced with /ai, au/ respectively. The authors think such replacements are acceptable.

The mappings from Hakka tones to Min-nan tones are listed in Table 5. The mapping from sea-land Hakka to Min-nan, Table 5(a), can be said to be a nice one-to-one mapping because both have the same number of lexical tones, and for each lexical tone of Hakka one can find a lexical tone in Min-nan that has almost same pitch-contour shape. The mapping from four-country Hakka to Min-nan, Table 5(b), is also straightforward. If tone number 7 is removed from Min-nan, then this mapping is still a nice one-to-one mapping.

Table 5. Tone mapping from Hakka to Min-nan.

(a) sea-land Hakka to Min-nan

Hakka tone number	1	2	3	4	5	7	8
Mapped Min-nan tone number	2	5	3	8	1	7	4
Example Chinese characters	衫	短	褲	寬	人	鼻	直

(b) four-country Hakka to Min-nan

Hakka tone number	1	2	3	4	5	8
Mapped Min-nan tone number	5	2	1	4	3	8
Example Chinese characters	夫	虎	富	福	湖	復

The mapping from Mandarin tones to Min-nan tones is listed in Table 6. Three of the Mandarin lexical tones, *i.e*. high-level, rising, and falling, also exist as Min-nan lexical tones. Besides these three tones, the low-level tone of Min-nan and the low-dipping tone of Mandarin are perceived to be almost identical. Therefore, the low-dipping tone of Mandarin

can be mapped to the low-level tone of Min-nan. The neutral tone of Mandarin has a shorter duration than the other tones. This contrast also exists in Min-nan, *i.e*. both abrupt tones have shorter durations. In addition, the low-abrupt tone of Min-nan has a low pitch-height, just as the neutral tone has. Hence, the neutral tone of Mandarin can be mapped to the low-abrupt tone of Min-nan.

Table 6. Tone mapping from Mandarin to Min-nan.

Mandarin tone number	1	2	3	4	5
Mapped Min-nan tone number	1	5	3	2	4
Example Chinese characters	加	油	打	氣	的

Here, to show the abilities of the adapted pitch-contour models, the authors take the Mandarin chunk, "花店的老闆" (boss of a flower shop), as an example. The sequence, X_1X_2, …, X_n, is hence, 1, 4, 5, 2, 3. And after lexical-tone mapping, the sequence, Y_1Y_2, …, Y_n, is obtained as 1, 2, 4, 5, 3. Then the mapped sequence and relevant contextual information are fed into the HMM and ANN models, respectively. The pitch-contours output by the two models are shown in Figure 3. The solid line is generated by the HMM model, the dotted line is generated by the ANN model, and the gray line is a mixture of the former two. Basically, these lines can all be recognized in their carried lexical tones. On the other hand, apparent differences in pitch heights can also be seen in the middle three syllables. Due to this phenomenon, the authors think the mixed pitch contour would be the better choice.

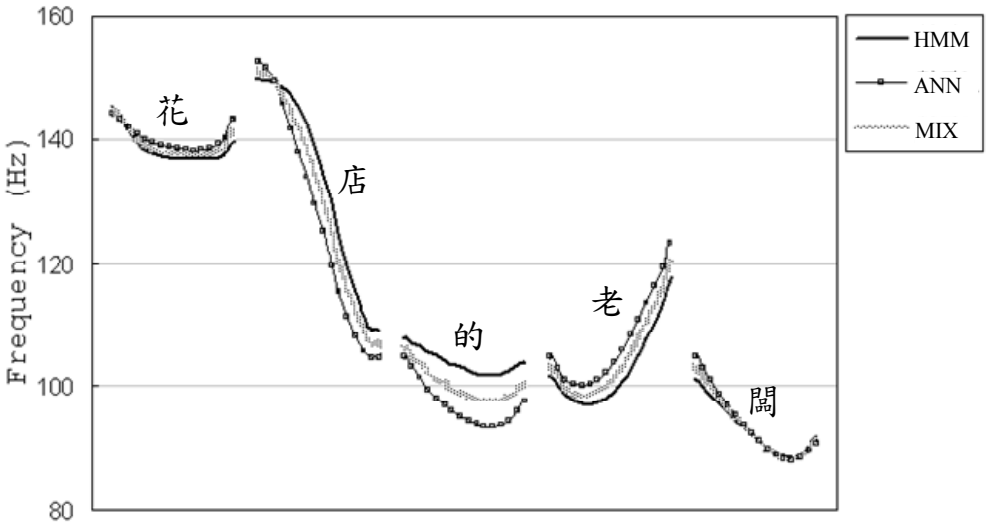


Figure 3. Pitch contours generated by adapted models.

4. Signal Waveform Synthesis

In this system, each syllable of a language has only one utterance recorded in a level tone (high or medium level). Therefore, the original syllable waveform must be manipulated to obtain synthesized waveforms with different prosodic characteristics. The synthesis method used here is TIPW [Gu *et al.* 1998]. TIPW is an improved variant of PSOLA, *i.e.* the effects of chorus and reverberation are largely reduced. Besides, TIPW is capable of adjusting vocal track length through re-sampling [Gu 2001].

Originally, TIPW was developed for synthesizing Mandarin speech. Thus, it does not support the synthesis of signal waveform with suddenly changed amplitude that is often found at the ending portion of an abrupt-tone syllable, *e.g.* /zit8/. Nevertheless, abrupt-tone syllables are very frequently used in Min-nan and Hakka. One method to overcome this difficulty is to treat the end portion of an abrupt-tone syllable as a stop consonant. Then, the same method used in synthesizing a stop consonant at the syllable initial portion can also be adopted to solve this problem.

4.1 A Fluency-Improving Method

In addition, the authors have made another improvement to TIPW. This improvement significantly increases the fluency of the synthesized speech. In an ordinary speech synthesis system, the subsystem of prosodic-parameter generating only determines the duration value, *Es*, of a syllable to be synthesized. The detailed dividing of syllable duration, *Es*, to its comprising phonemes is, however, not controlled by the prosody subsystem. Furthermore, the subsystem of signal waveform synthesis usually extends (or shrinks) the original speech waveform to an intended time length in a linear manner. According to this study, linear extending (or shrinking) of time length is a major cause of a decrease in much of the fluency of synthesized speech.

Consider an example syllable, /man/. Suppose that, in its original recorded waveform, the three phonemes, /m/, /a/, and /n/, occupy *Dm*, *Da*, and *Dn* milliseconds, respectively, and $Ds = Dm + Da + Dn$. A phenomenon that can be observed is that the ratio, $(Dm+Dn)/Ds$, will become smaller when /man/ is uttered within a sentence instead of being uttered in isolation. Currently, the authors are studying a simple method to simulate this phenomenon. In further research, the authors will study it with a more systematic method. The method used here is as depicted in Figure 4. That is, a piece-wise linear function is used to map the time-axis of a synthetic syllable waveform to the time-axis of its original waveform. In Figure 4, the symbols,

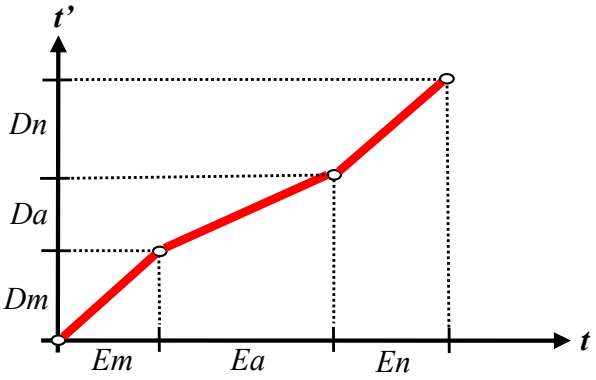


Figure 4. Piece-wise linear mapping function.

Em, *Ea*, and *En* represent the time lengths of the three phonemes, /m/, /a/, and /n/ in the synthesized waveform while *Dm*, *Da*, and *Dn* represent these three phonemes' time lengths in the original waveform. In this system, the values of *Em*, *Ea*, and *En* are determined by the following procedure:

```
r = 0.6;
while ( r >= 0.1 ) {
   Em = (Dm/Ds) * r * Es;
   En = (Dn/Ds) * r * Es;
   Ea = Es - Em - En;
   if (Ea > Es*0.4) break;
   r = r - 0.05;
}
Eb = Em + En;
if (Em > 0 && Em/Eb < 0.35) { Em = 0.35*Eb;  En=Eb-Em; }
if (En > 0 && En/Eb < 0.35) { En = 0.35*Eb;  Em=Eb-En; }
```

If the structure of a syllable is the same as /san/ or /an/, *i.e.* without voiced initial consonant, then the values of *Dm* and *Em* can be set to zero directly. Similarly, if the structure of a syllable is the same as /ma/, *i.e.* without a voiced ending consonant, then the values of *Dn* and *En* can be set to zero directly. Apparently, to apply the procedure given above, the boundary points between adjacent phonemes must be labeled beforehand in order to compute the values of *Dm*, *Da*, and *Dn*, and to construct the mapping function.

4.2 Example Waveforms

4.2.1 TIPW Synthesis Method

Since the synthesis method, TIPW, is not as popular as PSOLA, the authors will illustrate its processing steps with signal waveforms. To obtain a complete view of TIPW, including a detailed explanation of the method, see [Gu *et al.* 1998]. Here, let the two adjacent pitch periods in Figure 5(a) be around the mapped (using the piece-wise linear mapping function) time point, τ_m, in a recorded syllable. The first step of TIPW is to determine the weights, w_1 and w_2, for the left and right pitch periods. The value of w_2 is computed as $(\tau_m - \tau_1) / (\tau_2 - \tau_1)$ where τ_1 and τ_2 are time points of the left and right pitch periods' centers respectively. The value of w_1 is simply 1 - w_2. By weighting the two pitch periods with w_1 and w_2 respectively, one can obtain the two waveforms shown in Figure 5(b). Here, weighting a signal waveform with a weight, w, means that the value of each signal sample in the waveform is multiplied by the weight, w.

The second step is to window the pitch waveforms with two Hanning (or cosine) window halves. Here, windowing a signal waveform $x(n)$ with a window function $f(n)$ means that the result sample value at time n is $x(n) \cdot f(n)$, *i.e.* one-to-one multiplying. The two waveforms in Figure 6(a) are obtained by windowing the left pitch period in Figure 5(b) with two symmetric half Hanning windows. Here, the window length, L, is set to the smaller of L_1 and L_m where L_1 is the left pitch period's length and L_m is the length of the period to be synthesized. The detailed formula for the two window functions used in the left and right sides, respectively, of Figure 6(a) are:

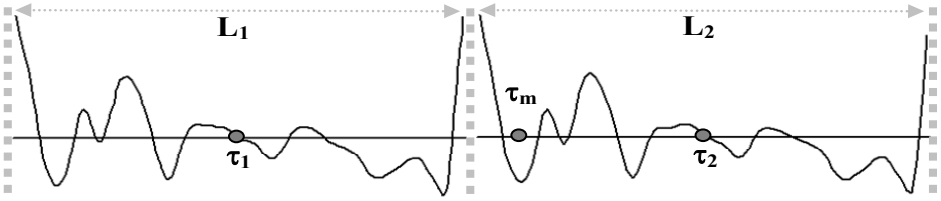
$$f_{left}(n) = 0.5 + 0.5\cos\left(\frac{n}{L}\cdot\pi\right), \quad n = 0,\ 1,\ 2,\ ...,\ L-1, \tag{3}$$

$$f_{right}(n) = 0.5 - 0.5\cos\left(\frac{n}{L}\cdot\pi\right), \quad n = 0,\ 1,\ 2,\ ...,\ L-1. \tag{4}$$

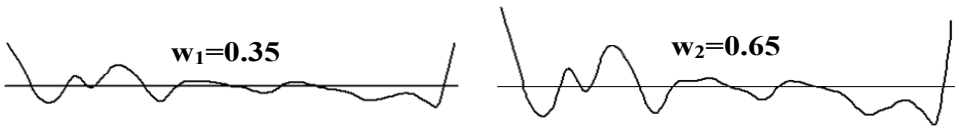
After windowing, the signal samples' values will be depressed in proportion to the window function's curve height. This can be seen from Figure 6(a). Similarly, the two waveforms in Figure 6(b) are obtained by windowing the right pitch period in Figure 5(b) with two symmetric half Hanning windows. However, the window length is now set to the smaller of L_2 and L_m. Note that the window length determination rules are important because they can prevent the effects of reverberation and dual-tones.

Then, as the last step, the four waveforms in Figure 6(a) and Figure 6(b) are overlapped and added to obtain a synthesized pitch period whose waveform is shown in Figure 7. Here, "overlapped" means that the four waveforms' locations on the time axis are left or right shifted in order that they have same starting or ending times. In detail, the two waveforms on

the left side of Figures 6(a) and 6(b) are left aligned to have same starting time, 0, in terms of the time axis of Figure 7. In contrast, the two waveforms on the right side of Figures 6(a) and 6(b) are right aligned to have same ending time, L_m-1. "Added", as used here, means that at each time point *n*, the four waveforms' four signal-sample values are added together to become the signal sample at time *n* for the resulted waveform.

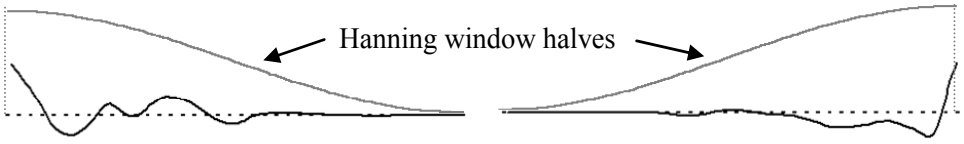


(a) original pitch periods



(b) weighted pitch periods

Figure 5. Original and weighted waveforms of two adjacent pitch periods.



(a) two half-windowed waveforms from the left pitch period

(b) two half-windowed waveforms from the right pitch period

Figure 6. Hanning windowed pitch waveforms.

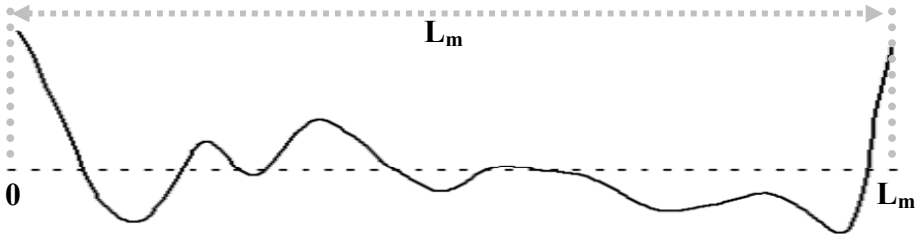


Figure 7. Synthesized pitch waveform.

4.2.2 Piece-wise Linear Mapping

To demonstrate the effect of the piece-wise linear mapping function, take the Mandarin word, "農田" (farmland) as an example and show its signal waveforms obtained from the original recording and the synthesis processing. The waveform in Figure 8 is a direct concatenation of the recorded waveforms of /nong-1/ and /tien-1/ while the waveform in Figure 9 is synthesized by this system. From Figure 8, it can be observed that the /ng/ part in /nong/ and the /n/ part in /tien/ both occupy a large portion of the syllable duration. However, through the remedying of the piece-wise linear mapping function, this phenomenon is largely reduced, and the fluency of the synthesized speech is improved greatly. It can be seen from Figure 9 that the duration ratios of /ng/ to /nong/ and /n/ to /tien/ now apparently become smaller.

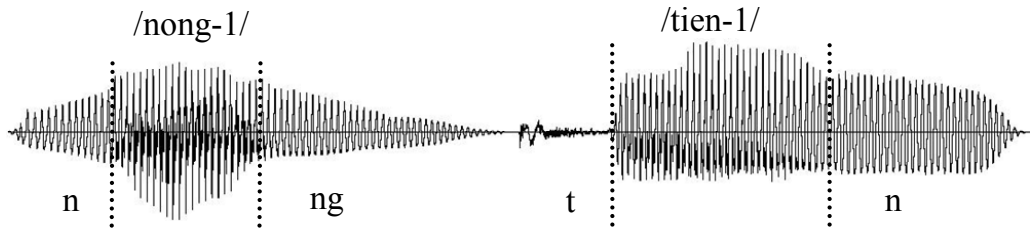


Figure 8. Direct concatenation of the recorded syllables, /nong-1/ and /tien-1/.

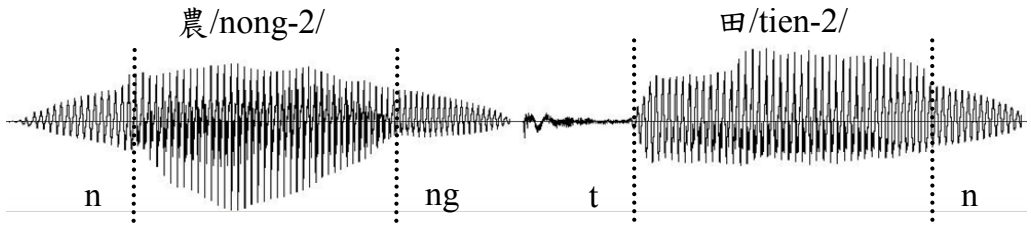


Figure 9. A synthetic waveform for /nong-2 tien-2/.

5. Experiments and Results

After the integrated system is implemented and ready to run, perception tests are conducted. The first issue of concern is the naturalness levels of Mandarin and Hakka pitch-contours generated by the Min-nan trained pitch contour models. Therefore, three short articles written in Mandarin, Hakka and Min-nan, respectively, are fed as inputs into the system. Then, the output speeches are played to each of the persons participating in the tests. Here, ten persons studying in university were invited to evaluate the synthetic speeches. For each person, two pairs of speech files were played, (*Sn*, *Sm*) and (*Sn*, *Sh*). *Sn*, *Sm*, and *Sh* represent synthetic Min-nan, Mandarin, and Hakka speeches, respectively. Then, the person was requested to give a score of -2, -1, 0, 1, or 2 for each pair. Here, 2 and -2 mean "better", 1 and -1 mean "slightly better", and 0 means "almost identical". As to the sign, positive sign means the latter is more

natural, and negative sign means the former is more natural. According to the scores given by the ten persons, the averaged scores are computed to be -0.6 for Mandarin and -0.2 for Hakka. That is, the synthetic Hakka speech was perceived to be almost as natural as the synthetic Min-nan speech. But the synthetic Mandarin speech was perceived to be less natural than the Min-nan speech. This is because the pitch-contours generated for Mandarin were perceived to have a slightly strange accent.

Another issue of concern is the fluency of the synthetic speech. Hence, two synthesis conditions are considered here for synthesizing a short Mandarin article. In the first condition, the mapping function, used in waveform synthesis, between the synthetic syllable's time axis and the recorded syllable's time axis is forced to be linear. In the other condition, the mapping function shown in Figure 4 is adopted. The two synthetic speeches obtained under the two conditions were played to each of the participating persons to compare their fluency. Again, each person was requested to give a score of -2, -1, 0, 1, or 2. The meanings of these numbers are as mentioned above. The average score was computed to be 0.4. That is, the fluency of the synthetic speech under the second condition was better than the one under the first condition.

6. Concluding Remarks

In this paper, the authors intend to promote the idea of synthesizing Mandarin, Min-nan, and Hakka speech with an integrated system. To show it is feasible, a possible system framework and some feasible implementation methods for the system components are proposed. According to the system framework and implementation methods presented, the authors have built an integrated speech synthesis system for the three languages. Then, speech files output from the system were used to perform perception tests. The initial results show that the lexical tone carried in the synthetic Mandarin and Hakka speeches can all be correctly recognized even though the pitch-contour models are trained with Min-nan sentences. As to naturalness level, the synthetic Hakka speech is perceived to be more natural than the synthetic Mandarin speech. How to interpret this phenomenon, though, is left to further studies.

Acknowledgments

This study is supported by National Science Council under the contract number, NSC 94-2218-E-011-007.

Reference

Chen, S. H., S. H. Hwang, and Y. R. Wang, "An RNN-based Prosodic Information Synthesizer for Mandarin Text-to-Speech," *IEEE Trans. on Speech and Audio Processing*, 6(3), 1998, pp. 226-239.

Chiou, H. B., H. C. Wang, and Y. C. Chang, "Synthesis of Mandarin Speech Based on Hybrid Concatenation," *Computer Processing of Chinese and Oriental Languages*, 5(1), 1991, pp. 217-231.

Chou, F. C., Corpus-based Technologies for Chinese Text-to-Speech Synthesis, PhD thesis, National Taiwan University, Taipei, Taiwan, 1999.

Chu, M., H. Peng, Y. Zhao, Z. Niu, and E. Chang, "Microsoft Mulan - a Bilingual TTS System," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2003, Hong Kong, China, vol. 1, pp. 264-267.

Gu, H. Y., and W. L. Shiu, "A Mandarin-Syllable Signal Synthesis Method with Increased Flexibility in Duration, Tone and Timbre Control," *Proceedings of the National Science Council ROC(A)*, 22(3), 1998, pp. 385-395.

Gu, H. Y., and C. C. Yang, "A Sentence-Pitch-Contour Generation Method Using VQ/HMM for Mandarin Text-to-speech," In *Proceedings of International Symposium on Chinese Spoken Language Processing*, 2000, Beijing, China, pp. 125-128.

Gu, H. Y., "Signal Resampling in Speech Synthesis," In *Proceedings of the 5th World Multi-conference on Systemics, Cybernetics and Informatics*, 2001, Orlando, USA, vol. vi, pp. 521-525.

Gu, H. Y., and H. C. Tsai, "A Pitch-Contour Model Adaptation Method for Integrated Synthesis of Mandarin, Min-Nan, and Hakka Speech," In *Proceedings of the 9th IEEE International Workshop on Cellular Neutral Networks and their Applications*, 2005, Hsin-Chu, Taiwan, pp. 190-193.

Gu, H. Y., and W. Huang, "Min-Nan Sentence Pitch-contour Generation: Mixing and Comparison of Two Kinds of Models," In *Proceedings of Conference on Computational Linguistics and Speech Processing (ROCLING)*, 2005, Tai-Nan, Taiwan, pp. 213-225. (in Chinese)

Lee, L. S., C. Y. Tseng, and C. J. Hsieh, "Improved Tone Concatenation Rules in a Formant-Based Chinese Text-to-Speech System," *IEEE Trans. Speech and Audio Processing*, 1(3), 1993, pp. 287-294.

Lee, S. J., K. C. Kim, H. Y. Jung, and W. Cho, "Application of Fully Recurrent Neural Networks for Speech Recognition," In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, 1991, Toronto, Canada, pp. 77-80.

Modulines, E., and F. Charpentier, "Pitch-synchronous Waveform Processing Techniques for Text-to-Speech Synthesis Using Diphones," *Speech Communication*, 9(5), 1990, pp. 453-467.

Rabiner, L., and B. H. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, New Jersey, 1993.

Shih, C., and R. Sproat, "Issues in Text-to-Speech Conversion for Mandarin," *International Journal of Computational Linguistics and Chinese Language Processing*, 1(1), 1996, pp. 37-86.

Shiu, W. L., A Mandarin Speech Synthesizer Using Time Proportioned Interpolation of Pitch Waveform, Master Thesis, National Taiwan University of Science and Technology, Taipei, Taiwan, 1996. (in Chinese)

Wang, R. H., "Overview of Chinese Text-to-Speech System," In *Proceedings of International Symposium on Chinese Spoken Language Processing*, 1998, Singapore.

Wu, C. H., and J. H. Chen, "Automatic Generation of Synthesis Units and Prosodic Information for Chinese Concatenative Synthesis," *Speech Communication*, 35(3). 2001, pp. 219-237.

Yu, B. C., Z. C. Syu, and C. N. Wu, *General Phonetic Symbol System for Languages in Taiwan*, Nan-Tien Book Company, Taipei, 1999.

Yu, M. S., N. H. Pan, and M. J. Wu, "A Statistical Model with Hierarchical Structure for Predicting Prosody in a Mandarin Text-to-Speech System," In *Proceedings of International Symposium on Chinese Spoken Language Processing*, 2002, Taipei, Taiwan, pp. 21-24.

Computational Linguistics and Chinese Language Processing
Vol. 12, No. 4, December 2007, pp. 391-410

Some Studies on Min-Nan Speech Processing

Wei-Chih Kuo*, Chen-Chung Ho*, Xiang-Rui Zhong*,

Zhen-Feng Liang*, Hsiu-Min Yu+, Yih-Ru Wang*, and Sin-Horng Chen*

Abstract

In this paper, three studies of Min-Nan speech processing are presented. The first study concerns the implementation of a high-performance Min-Nan TTS system. On the basis of the waveform templates of 877 base-syllables used as basic synthesis units and through the application of the RNN-based prosody generation method and the PSOLA algorithm for prosody modification, this Min-Nan TTS system can convert texts, represented in both Han-Luo (漢羅) and Chinese logographic writing systems, into natural Min-Nan speech. An informal, subjective listening test confirms that the system performs well and the synthetic speech sounds natural for well-tokenized Min-Nan texts and for automatically tokenized Chinese logographic texts. The second investigation concerns the realization of a Min-Nan speech recognizer. It adopts the *initial-final*-based HMM approach with a simple base-syllable bigram language model. A base-syllable recognition rate of 65.1% has been achieved. Finally, a model-based tone labeling method is presented. This method adopts a statistical model to eliminate the affections of all factors other than tone on the syllable pitch contour for automatic tone labeling. Experimental results confirm that this method outperforms the conventional VQ-based approach.

Keywords: Min-Nan Text-to-Speech System, Speech Recognition, Model-Based Tone Labeling

1. Introduction

Min-Nan is one of the subcategories of the Min dialect, which is one of the seven Chinese dialect families [Yuan *et al*. 1989]. Aside from some pockets of speakers scattered over

* Department of Communication Engineering, Chiao Tung University
Tel: +886-3-5731844, Fax: +886-3-5710116
E-mail: yrwang@mail.nctu.edu.tw
+ Department of Foreign Languages and Literature, Chung Hua University

[Received November 8, 2006; Revised June 29, 2007; Accepted July 5, 2007]

Southeast Asia, varieties of Min-Nan are spoken in southern Fujian, eastern and southeastern Guangdong, and are spread over much of the islands of Hainan and Taiwan, where it is spoken by approximately 73.3 percent of the inhabitants [Huang 1995]; hence, it is often called Taiwanese. In recent years, even though Min-Nan has captured much attention in Taiwan's academic community, research related to its speech processing still remains small due to (1) non-unified writing standards, (2) the various accents of Min-Nan used in Taiwan, and (3) lack of non-public Min-Nan speech and text corpora. These multiple factors may lead to hindering progress in Min-Nan speech processing technology.

However, in spite of the aforementioned deficiencies, which add a degree of difficulty to the automatic processing of this language, three achievements in the technology of Min-Nan speech processing have been made in our study, including the implementation of a high-performance Min-Nan TTS system, the realization of a Min-Nan speech recognizer, and a model-based tone labeling method.

The paper is organized as follows. Section 2 gives a brief introduction to the background of Min-Nan. Section 3 presents the proposed Min-Nan TTS system. Section 4 discusses the realization of a Min-Nan speech recognizer. Section 5 describes a new model-based tone labeling method for Min-Nan speech. Some conclusions are given in the last section.

2. A Brief Description of Min-Nan

Like Mandarin and most other Chinese dialects, Min-Nan is monosyllabic in nature, which means that, basically, every syllable is a free morpheme with a meaning value, and that syllable is the unit for pronunciation and every character in text reading is assigned one, but not the only, syllabic sound. The syllabic structures of both Min-Nan and Mandarin can be described in terms of traditional Chinese philology, where syllable is conventionally viewed to be formed by two constituents: the "initial", a consonantal onset, and the "final", made up from a prenucleus onglide, the nucleus – the only obligatory syllabic element, and a coda. Compared with Mandarin, which has 21 initials, 37 finals, and 408 base-syllables, which are legitimate syllables formed by rule-governed combinations of initials and finals, Min-Nan has 18 initials, 82 finals, and 877 base-syllables. In addition to the differences in the numbers of the above-mentioned syllabic constituents and base-syllables, Min-Nan and Mandarin also show differences in the types of syllables, which are often classified by Chinese linguists into "checked" or "entering" syllables, namely syllables ending in a plosive coda (-p,t,k, and a glottal stop), and "smooth" or "slack" syllables, namely syllables ending in a non-plosive. Of the two dialects, only Min-Nan has checked/entering syllables, which leads to different prosodic features associated with syllable types from those of Mandarin.

Min-Nan is a tonal language, where every syllable has an inherent tone, and tones of different pitch values function to distinguish different lexical meanings. [Yang 1999] Min-Nan

has 8 tones, including 7 lexical tones and one degenerated tone, each of which displays a distinct pitch contour. Moreover, based on the type of syllable, tones inherent in entering/checked syllables are termed entering/checked tones accordingly, and those in smooth/slack syllables are called non-entering/non-checked tones. If syllabic tones are under consideration, Min-Nan has approximately 2000 syllables. It is also worth a mention in passing that, despite the fact that in Min-Nan mono-syllable is held to be the basic pronunciation unit, in actual speech mono-syllabic morphemes are not uttered independently; instead, two or more mono-syllabic morphemes, to convey meaning relationship, are concatenated to form meaningful or syntactic poly-syllabic units, which generates changes in the inherent pitch contours of the concatenated syllables. This tonal variation is called "tone sandhi," a very well-known term used to describe the tonal changes depending on the tonal environment in which poly-syllabic words occur.

As for the writing system, although no consistent written forms have been standardized for Min-Nan, two sets of writing systems have been more widely accepted in Taiwan, namely "Romanization" or "Luo Ma Pin Yin" (羅馬拼音) and "Han Luo" system (漢羅系統). In the former, Roman letters are used to spell or transcribe Min-Nan speech, and numbers to specify its tones. This writing system has been widely used among churches to transcribe the Bible that has been translated into Min-Nan. With limited letters and numbers, Romanization provides an easy way to learn the pronunciation of Min-Nan. Therefore, it is not uncommon to see many functionally illiterate Min-Nan elderly churchgoers who cannot read Chinese characters but can recite in Min-Nan scriptures in the Bible written in Romanization. However, since most of the Min-Nan native speakers are literate, and possible ambiguity may be caused by homophones when Chinese characters are not shown, the other writing system, namely Han-Luo system (a hybrid from Chinese characters and Romanization) is used more often in written texts. Unfortunately, the problem still exits in the inconsistency of the Chinese characters selected to represent Min-Nan words or expressions. Except for some popular words, people often choose by preference a string of Chinese characters with similar pronunciations to Min-Nan to represent a Min-Nan word. This increases the degree of difficulty of text analysis for Min-Nan speech processing.

Another linguistic phenomenon worth noting is that, for many Min-Nan syllables, two pronunciation styles co-exist. The first one is called Bai Hua (白話) – the vernacular reading – which is widely used in daily conversation. The other, referred to as Wen Yan (文言) – literary reading – is restrictedly used in reading poetry, some numbers, or in terms used for naming people, buildings, festivals, and so forth.

3. An Implementation of Min-Nan TTS System

In this section, the implementation of a high-performance Min-Nan TTS system is presented. Figure 1 shows a block diagram of the proposed Min-Nan TTS system. It is worth noting that such an approach has been successfully applied to developing a high-performance Mandarin TTS system [Chen *et al*. 1998] [Chen *et al*. 2000] [Ho *et al*. 2000]. The system consists of four main functional blocks: a text analyzer, a recurrent neural network (RNN)-based prosody generator, an acoustic inventory, and a PSOLA speech synthesizer. Input text is first tokenized into word/syllable sequence by the text analyzer. The waveform sequence corresponding to the syllable sequence is then formed by the acoustic inventory. Meanwhile, some linguistic features are extracted from the syllable sequence and used in the RNN-based prosody generator to generate necessary prosodic parameters. Afterwards, the PSOLA speech synthesizer uses these prosodic parameters to modify the prosody of the waveform sequence and generate the output synthetic speech. In the following subsections, we will discuss these four main functional blocks in detail.

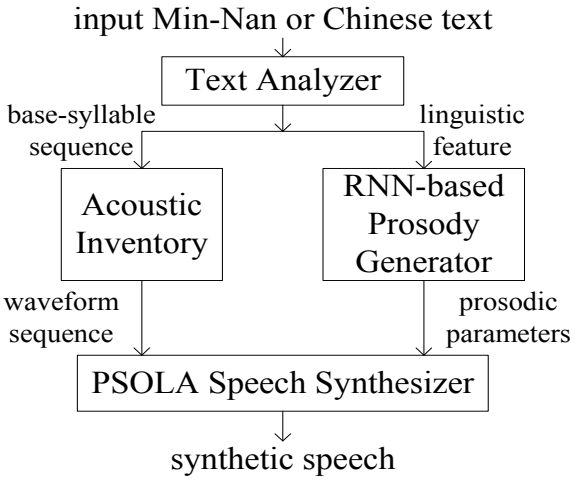


Figure 1. A schematic diagram of the proposed Min-Nan TTS system.

3.1 The Text Analyzer

The function of the text analyzer is first to tokenize the input text into word sequence and then extract relevant linguistic features from the sequence. Two kinds of input texts are processed. One kind is Min-Nan text represented in the hybrid written form of Han-Luo. Another kind of text is represented in Chinese characters only. Figure 2 displays the block diagram of the text analyzer. It first converts an input text into a Unicode sequence in preprocessing. Here, a look-up table is used to find all syllables represented in Romanized form. It then uses two lexica and a long-word-first criterion to convert the Unicode sequence into a word sequence.

The first lexicon is a Min-Nan lexicon. It contains about 120,000 entries represented in the Han-Luo system. Each entry is a word with a length in the range of 1-6 syllables. The second lexicon, with 110,000 entries, is a Chinese-to-Min-Nan lexicon. It is an extended version of our Chinese lexicon in which Chinese words are transferred to Min-Nan syllable sequence character by character. The use of the Chinese-to-Min-Nan lexicon helps us solve the out-of-vocabulary problem encountered in the text analysis. This also makes the system possess the capability of processing input Chinese text.

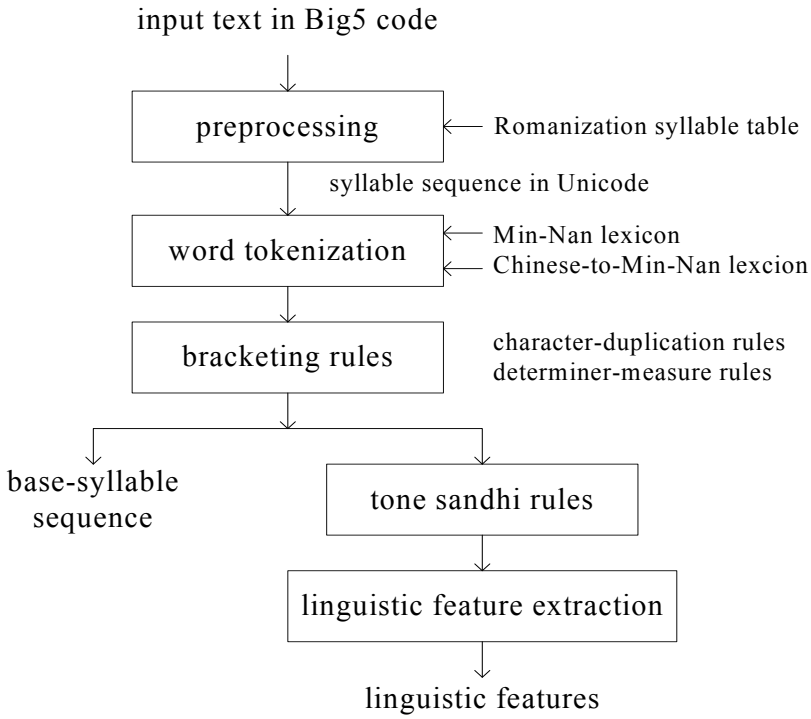


Figure 2. A functional block diagram of the text analyzer.

We then use two bracketing rules to construct two types of compound words which are not contained in the lexicon [Huang 2001]. One is for character-duplicated compound words and the other is for determiner-measured compound words. Here, we also decide whether to pronounce the number of a determiner-measured compound word in the style of vernacular reading or in literary reading. For instance, "1998" should be pronounced in the second style as "it kiu2 kiu2 bat", while "兩萬一千八百" (twenty one thousand eight hundred) is pronounced in the first style as "lng7 ban7 chit chheng peh pah".

After obtaining the word sequence, a set of tone *sandhi* rules is then explicitly applied to change the lexical tones of all syllables into the ones to be pronounced [Huang 2001]. Basically, all syllables except the final one of a word chunk (or pronunciation group) have to change their tones. These rules [Cheng 1993] are listed below:

$$
\begin{aligned}
&1 \rightarrow 7 \\
&7 \rightarrow 3 \\
&3 \rightarrow 2 \\
&2 \rightarrow 1 \\
&5 \rightarrow \begin{cases} 7 & \text{south} \\ 3 & \text{north} \end{cases} \\
&4\ (p, t, k) \leftrightarrow 8\ (p, t, k) \\
&4h \rightarrow 2 \\
&8h \rightarrow 3
\end{aligned}
\tag{1}
$$

Here, an arrow indicates the way a tone changes, *e.g.*, Tone 2 will change to Tone 1; "north" and "south" mean the northern and southern parts of Taiwan; and "*p*", "*t*", "*k*", and "*h*" represents entering tones. Besides, four additional rules [Cheng 1993] are used for special cases where a syllable preceding the special character "仔, a function word" (/a/) has been changed to Tone 2 or 3:

$$
\begin{aligned}
&7 \rightarrow 3 \rightarrow 7 \\
&8h \rightarrow 3 \rightarrow 7 \\
&3 \rightarrow 2 \rightarrow 1 \\
&4h \rightarrow 2 \rightarrow 1
\end{aligned}
\tag{2}
$$

For instances, 鋸(ki3→ki1)仔(saw) and 葉(hioh8→hioh7)仔(leaf). An advantage of the approach of using explicit tone *sandhi* rules is that it results in obtaining an RNN-based prosody generator with high efficiency on learning phonological rules of human's prosody generation.

Two sets of linguistic features are then extracted from the word sequence. One is the syllable sequence, which is extracted directly from the word sequence by referring to the lexicon. This will be used in the acoustic inventory to form the basic waveform template sequence. Another consists of two subsets of syllable-level and word-level linguistic features and is used in the RNN-based prosody generator to synthesize proper prosodic parameters. The subset of syllable-level linguistic features contains four parameters: the *initial* type, *final* type, and tone of the current syllable, and the position of the current syllable in the current word. The subset of word-level linguistic features includes two sequences of word length and PM.

3.2 The RNN-Based Prosody Generator

The RNN-based prosody generator uses four RNNs to separately generate four types of prosodic parameters for the current syllable: 4 pitch-contour parameters [Chen *et al*. 1990],

initial and *final* durations, log-energy level, and the following pause duration. All four RNNs have the same architecture shown in Figure 3. Each RNN is a four-layer network with outputs of the two hidden layers and the output layer being fed back to their own inputs. An RNN of this architecture has been proven in previous studies to be effective in exploring the contextual information of the input linguistic features for the generation of proper output prosodic information [Chen *et al*. 1998]. Table 1 shows the input linguistic features used in these four RNNs.

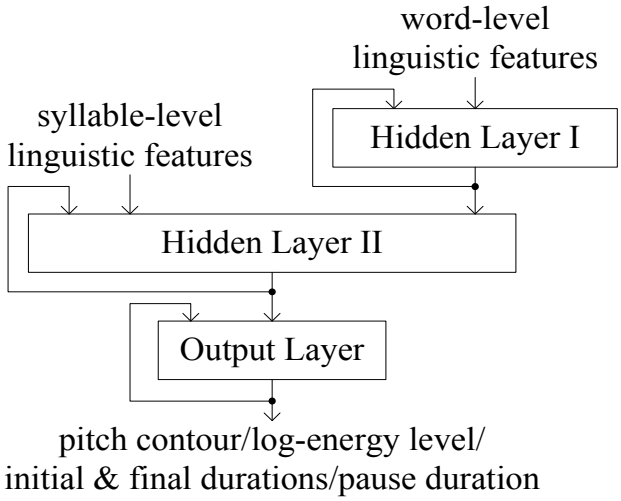


Figure 3. The architecture of the RNN used in the TTS system.

Table 1. The input linguistic features used in the four RNNs for generating syllable pitch contour, initial and final durations, syllable energy level, and pause duration. Here "common" means features commonly used for all four RNNs.

syllable-level linguistic features	common	1. tone of current syllable 2. position of current syllable in a word
	Pitch contour	1. tone of next syllable 2. *initial* types of current and next syllables
	Initial and *final* durations	1. *initial* and *final* types of current syllable 2. light pronunciation of current syllable
	energy level	1. *initial* and *final* types of current syllable 2. light pronunciation of current syllable
	pause duration	1. *initial* and *final* types of current syllable 2. light pronunciation of current and next syllables 3. tone of next syllable 4. existence a break following a long word?
word-level linguistic features (common)	1. lengths of current and next words 2. existence of special PM following the next word whose length equals to 1? 3. PM type following the current word 4. POSs of the current and next words	

These four RNNs can be trained using a large, single-speaker speech database following the back-propagation through time (BPTT) algorithm [Haykin 1994]. The BPTT algorithm is a supervised training algorithm used to learn the mapping from input linguistic features extracted from the input text to output prosodic parameters extracted from the associated utterance. For preparing those inputs and outputs, all texts of the database are manually processed to obtain the word and POS sequences, and the associated utterances are also manually segmented. A further processing of the database is also done for extracting some additional features to improve the efficiency of RNN training. The further processing includes: (1) all minor and major breaks occurring at inter-syllable locations without PMs are manually detected and labeled with special marks; (2) some special characters (referred to as "虛詞, function word") which are consistently pronounced lightly and short are marked, *e.g.*, "甲" in "互氣甲, be angered" and "仔" in "囝仔, child"; (3) all 5-syllable and 6-syllable words are classified respectively into {2-3, 3-2} and {2-2-2, 3-3} pronunciation patterns; and (4) pitch contours of all short syllables are manually refined. Finally, we modify the learning process of the RNN for inter-syllabic pause duration d. Instead of letting the RNN learn the real pause duration, we first classify the pause duration into four classes: short ($d \le 75$ ms), medium ($75 \text{ ms} \le d \le 175$ ms), long ($175 \text{ ms} \le d \le 475$ ms), and very long ($475 \text{ ms} \le d$). The pause duration of the "short" class was further normalized with respect to the mean and standard deviation of the final types (2 types: with and without entering tone) of the processing syllable and the initial types (4 types) of the preceding syllable. We then let the RNN learn (1) the class of the pause duration and (2) the pause duration when it belongs to the "short" class. This change can let the RNN take care of both the detail of short pause duration and rough classification of long pause duration.

3.3 The Acoustic Inventory

The function of the acoustic inventory is to generate a waveform template sequence for each base-syllable sequence given by the text analyzer. It is a look-up table containing waveforms templates of all 877 base-syllables which are the basic synthesis units used in our system. All of the waveform templates are obtained from isolated-syllable utterances pronounced clearly by a male speaker. All of the speech signals are directly recorded digitally, using a PC with a sound card. The sampling rate is 20 kHz. Each utterance is manually pre-processed to detect ending-points and to label pitch marks.

3.4 The PSOLA Speech Synthesizer

The function of the PSOLA speech synthesizer is to generate the output synthetic speech by modifying the waveform template sequence of the base-syllable sequence given by the acoustic inventory using the prosodic parameters given by the RNN-based prosody generator.

Modifications include changing the pitch contour for each syllable, adjusting *initial* and *final* durations for each syllable, scaling the energy level for each syllable, and setting the pause duration for each inter-syllable location. Finally, the output synthetic speech is generated by a 16-bit Sound Blaster card.

3.5 Experimental Results

Performance of the proposed Min-Nan TTS system was examined by simulation, using a male speaker database. The database contains 255 utterances including 130 sentential utterances with lengths in the range of 5-30 syllables and 125 paragraphic utterances with lengths in the range of 85-320 syllables. The total number of syllables is 23,633. In addition, a set of 877 isolated base-syllable utterances was recorded for constructing the acoustic inventory. Most of these 877 utterances are syllables with Tone 1. All speech signals were digitally recorded at a 20 kHz rate. All utterances and associated texts were manually pre-processed in order to extract the acoustic features and the linguistic features required to train and test the system.

We first examined the performance of the RNN-based prosody synthesizer. Table 2 lists the root mean square errors (RMSEs) of the synthesized prosodic parameters. RMSEs of 10.2 (12.4) ms, 26.2 (32.4) ms, 15.1 (21) ms, 0.79 (0.80) ms/frame and 2.28 (3.12) dB were achieved in the inside (outside) test for initial duration, final duration, pause duration, pitch contour and log-energy level, respectively. Here, in the calculation of RMSE for pause duration, we set the target pause duration of the three classes of "medium", "long" and "very long" to be 75ms. The classification errors for the four pause duration classes were 12.1% and 13.8% for the inside and outside tests, respectively. Actually, over 80% of classification errors were associated with Class 2. Figure 4 shows a typical example of these synthesized prosodic parameters. It can be seen from the figure that the synthesized prosodic parameters of most syllables matched well with their original counterparts.

Table 2. The experimental results of RNN prosody generation.

	inside	outside
initial duration (ms)	10.2	12.4
final duration (ms)	26.2	32.4
pause duration (ms)	15.1	21.0
pitch contour (ms/Frame)	0.79	0.80
energy level (dB)	2.28	3.12

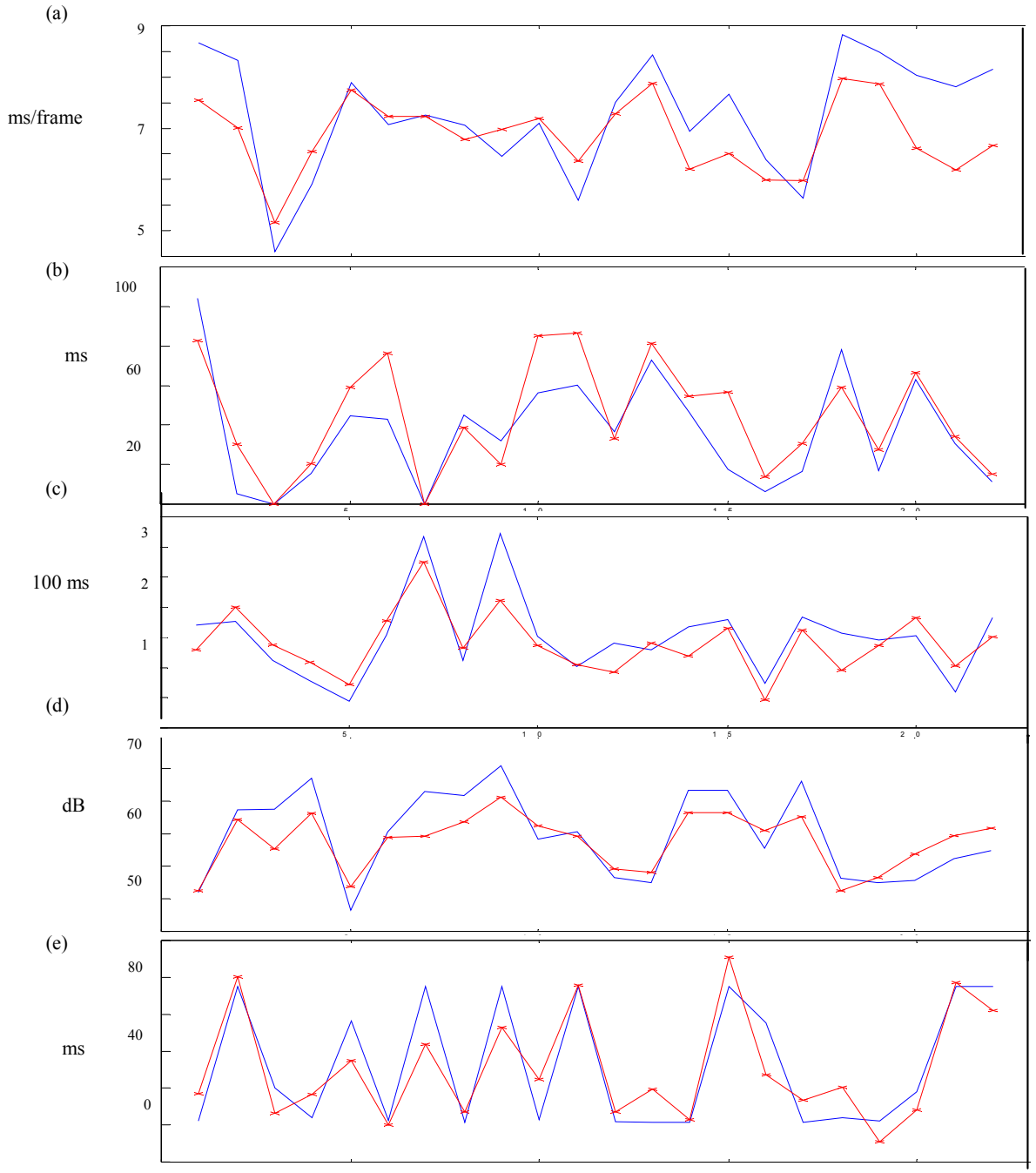


Figure 4. A typical example of synthesized prosodic parameters: (a) pitch mean, (b) initial duration, (c) final duration, and (d) log-energy level of syllable, and (e) inter-syllable pause duration. The text is "生活應該是鮮豔、開朗、充實，自信滿滿 the 享受人生才著。 seng-oah8 eng2-kai-si7 sian-iam7、khai-long2、chhiong-sit8，chu7-sin3-moa2-moa2 the hiang2-siu7 lin5-seng chai5 tioh8。".

The whole system has been implemented in software on a PC with a 16-bit Sound Blaster card. An informal subjective listening test using various texts not covered in the database was finally derived to examine the performance of the system. Many participants confirmed that all of the synthesized speeches sounded natural for well-tokenized Min-Nan (Han-Luo) texts and for automatically tokenized Chinese texts. However, the sound quality was only fair for automatically tokenized Min-Nan texts because of the lack of a standardized written form.

4. A Min-Nan Speech Recognizer

As can be expected, complicated linguistic properties would affect the performance of speech recognition. Compared with Mandarin, Min-Nan has an inventory of base-syllables double that of Mandarin, and contains syllables ending in a plosive coda, which are not found in Mandarin. These linguistic properties lead to a syllable recognition rate for Min-Nan significantly lower than for that of Mandarin. In [Lyu *et al*. 2000] [Lyu *et al*. 2003] , 825 basic syllables were used for Min-Nan speech recognition system, and a 58% syllable recognition rate was achieved when tri-phone HMM models were used. A Min-Nan speech recognizer is also implemented in this paper. Following the idea of using syllable *initial* and *final* as basic recognition units in Mandarin automatic speech recognition (ASR), the Min-Nan speech recognizer adopts 101 right-*final*-dependent *initial*s and 84 context-independent *final*s as basic acoustic modeling units. Each *initial* is modeled by a 3-state HMM and each *final* is modeled by a 5-state HMM. All 877 base-syllables, including 28 base-syllables with entering tone, can be represented by using these 185 sub-syllable units. Additionally, a 3-state silence model and a one-state short pause model are used to represent the background long silences and inter-syllabic short pauses, respectively. The recognition features consist of 12 MFCCs, 12 delta-MFCCs, 12 delta-delta-MFCCs, delta-log-energy and delta-delta-log-energy. They are extracted for each 30-ms frame with 10-ms frame shift. The cepstrum mean normalization (CMN) technique is also applied to remove the speaker bias.

We first examined the performance of the baseline recognizer (Scheme 1) using only acoustic models by simulation on a large Min-Nan speech database. The database was recorded in 16-kHz sampling rate. It consisted of many sentential and paragraphic utterances generated by 197 speakers, including 91 males and 106 females. We divided the database into two parts, one for training and the other for testing. The training set contained 105,687 syllables while the test set contained 12,211 syllables. The number of syllables in the database is only one-third of the TCC database, which is the most commonly used database for Mandarin ASR in Taiwan. The experimental result is displayed in the 2nd row of Table 3. A base-syllable recognition rate of 46.1% was achieved. The recognition result is relatively low as compared with a typical Mandarin base-syllable recognizer whose base-syllable recognition rate is usually over 60%. This could result, in part, from the fact that the number of

base-syllables in Min-Nan speech is almost twice as many as found in Mandarin speech, and in part from the high confusion of base-syllables of entering tone and their non-entering-tone counterparts. An error analysis showed that base-syllables of the same phonemic constituent with and without entering-tone are highly confusing pairs.

Table 3. The base-syllable recognition rates of the Min-Nan speech recognizer.

	Inclusion rate	deletion error	insertion error	recognition rate
Scheme 1	48.87%	2.90%	2.80%	46.1%
Scheme 2	52.73%	2.91%	2.56%	50.2%
Scheme 3	66.50%	3.14%	1.36%	65.1%

We then improved the baseline Min-Nan speech recognizer by considering the effect of tone *sandhi* rules. As shown in Equations (1) and (2), base-syllables with */h/* entering tone may change to their counterparts of non-entering tone. This tone *sandhi* will cause serious errors in both HMM model training and recognition test. The total number of *final*s with */h/* entering tone is 17 (out of 28 *final*s with entering tone). We, therefore, relabeled all syllables with */h/* entering tone in both the training and test data sets. Except when located before a long pause, 10-frame silence in our study, all syllables with */h/* entering tone were changed to their non-entering-tone counterparts. The performance of the modified recognizer (Scheme 2) is displayed in the 3rd row of Table 3. A base-syllable recognition rate of 50.2%, or a 4.1% improvement, was obtained.

The recognizer was further improved by incorporating it with a language model (LM). Due to the fact that it is very difficult to collect a large text database with proper tagging or parsing, we considered a simple base-syllable bigram LM instead of the conventional word bigram LM. A text database containing 325,267 syllables was used to train the LM. Texts in the database are news, articles, and stories. The performance of the improved recognizer (Scheme 3) is displayed in the 4th row of Table 3. A base-syllable recognition rate of 65.1%, which corresponded to a 30% error reduction rate, was achieved.

Last, the Min-Nan speech recognizer was applied to a domain-specific task, an in-car speaking assistant prototype for an intelligent transportation system (ITS). An in-car speaking assistant is a user-friendly spoken dialogue human-machine interface acting as an agent to allow the driver to easily control a variety of in-car equipment while keeping his hands and eyes on the road. To add the new module to the existing Mandarin-based in-car speaking assistant system, a Min-Nan grammar for ITS dialog management was needed. In this study, we simply implemented it by directly translated the Chinese grammar into a Min-Nan version. With some simple modifications, the Min-Nan speech recognizer with the ITS grammar was invoked in the ATK [Young 2007] as a real-time Min-Nan ASR module. It successfully expanded the function of the in-car speaking assistant to process Min-Nan input speech. Some

examples of on-line recognition results of the system are shown in Table 4.

Table 4. Some examples of on-line recognition results for Min-Nan input speec

User input	Recognition result (in Mandarin)
系統你好	系統你好
即馬我欲挃交大, 該按怎走?	然後我過去交大, 那會怎麼走
等下我要走科技路還是寶山路?	等一下我要走科技路還是寶山路
繼續直直走對不對?	繼續在馬路之後哪一個到

5. A Model-based Tone Labeling Method for Min-Nan Speech

The task of tone labeling is to determine the tone sequence pronounced in each utterance of a speech database [Li 2002] [Kuo *et al*. 2004]. A database with proper tone labeling should be good to be used in either TTS or ASR. Several approaches can be employed to tackle the task. First, a direct approach is to do the job manually by listening to and/or observing the pitch contour. However, as mentioned above, this approach will suffer from the difficulties of inconsistency and heavy workload. Another approach is to determine the tone sequence by applying the above tone *sandhi* rules to the associated text. As shown in [Liang *et al*. 2004], the tone sandhi rules have been applied to all syllables except for the ones word/sentence final. The results indicated that the tone labeling accuracy for the tonal variations was about 62-65%. The main problem of this approach is that it is not known exactly how to automatically form word chunks from the word sequence. Besides, determining tones only from texts may suffer from errors. The third approach is to regard it as a classification problem by classifying the pitch contours of all syllables with the same lexical tone using an unsupervised clustering technique such as vector quantization (VQ). A drawback of the third approach is that errors may occur because the pitch contour of a syllable in a continuous speech is influenced by many factors other than just the tone itself. The fourth approach is to tackle the task by an efficient pitch contour model which can separate all major affecting factors that control the variation of the pitch contour.

5.1 The Proposed Tone Labeling Method

In this study, we adopt the last approach by using a statistical pitch contour model [Wang *et al*. 2000] [Chen *et al*. 2005] [Yang 1999]. We first represent the pitch contour of each syllable by using a 3-rd order orthogonal polynomial expansion [Chen *et al*. 1990]. The basis polynomials used are normalized, in length, to [0,1] and can be expressed as:

$$\phi_0(\tfrac{i}{M}) = 1$$

$$\phi_1(\tfrac{i}{M}) = [\frac{12 \cdot M}{M+2}]^{1/2} \cdot [\frac{i}{M} - \frac{1}{2}]$$

$$\phi_2(\tfrac{i}{M}) = [\frac{180 \cdot M^3}{(M-1)(M+2)(M+3)}]^{1/2} \cdot [(\frac{i}{M})^2 - \frac{i}{M} + \frac{M-1}{6 \cdot M}] \tag{3}$$

$$\phi_3(\tfrac{i}{M}) = [\frac{2800 \cdot M^5}{(M-1)(M-2)(M+2)(M+3)(M+4)}]^{1/2}$$
$$\cdot [(\frac{i}{M})^3 - \frac{3}{2}(\frac{i}{M})^2 + \frac{6M^2 - 3M + 2}{10 \cdot M^2}(\frac{i}{M}) - \frac{(M-1)(M-2)}{20 \cdot M^2}]$$

for $0 \le i \le M$, where M+1 is the length of the current syllable log-pitch contour and $M \ge 3$. They are, in fact, discrete Legendre polynomials. A syllable pitch contour $f(\frac{i}{M})$ can then be approximated by:

$$\hat{f}(\tfrac{i}{M}) = \sum_{j=0}^{3} \alpha_j \cdot \phi_j(\tfrac{i}{M}) \qquad 0 \le i \le M\,, \tag{4}$$

where

$$\alpha_j = \frac{1}{M+1} \sum_{i=0}^{M} f(\tfrac{i}{M}) \cdot \phi_j(\tfrac{i}{M}) \tag{5}$$

The four coefficients are then divided into two parts: α_0 representing the mean and $[\alpha_1 \quad \alpha_2 \quad \alpha_3]$ representing the shape. They are separately modeled. The pitch mean model used can be expressed by:

$$Y_n = F_n + \beta_{pt_n} + \beta_{t_n} + \beta_{ft_n} + \beta_{p_n} \tag{6}$$

where Y_n is the observed pitch mean α_0 of the nth syllable; F_n is the normalized pitch mean and is modeled as a normal distribution with mean μ and variance ν; β_r is the compressing-expanding factor (CF) for affecting factor r; t_n, pt_n and ft_n represent, respectively, the lexical tones of the current, previous and following syllables; and p_n represents the prosodic state of the current syllable. Here, prosodic state roughly represents the state of the syllable in a prosodic phrase and is treated as hidden. Note that t_n ranges from 1 to 22 including 7 standard patterns of lexical tones and all their *sandhi* tones, while both pt_n and ft_n ranges from 0 to 22 with 0 denoting the cases of major punctuation marks {，, 。, ！, ；, ？, 、, ：} or the non-existence of the previous or following syllable. The CFs for $pt_n = 0$ and $ft_n = 0$ are set to zero because we do not want to count the effect of tone across a punctuation mark.

The pitch shape model used can be expressed by:

$$\mathbf{Z}_n = \mathbf{X}_n + \mathbf{b}_{pt_n} + \mathbf{b}_{t_n} + \mathbf{b}_{ft_n} + \mathbf{b}_{p_n} \tag{7}$$

where $\mathbf{Z}_n$ is the observed shape vector $\begin{bmatrix}\alpha_1 & \alpha_2 & \alpha_3\end{bmatrix}^T$ of the nth syllable's pitch contour; $\mathbf{X}_n$ is the normalized pitch shape vector and is modeled as a multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix R.

To estimate the parameters of these two models, an expectation-maximization (EM) algorithm is adopted. The EM algorithm is derived based on the maximum likelihood (ML) estimation from incomplete data with prosodic state and pronounced tone pattern being treated as hidden or unknown. To illustrate the EM algorithm, an auxiliary function is firstly defined in the expectation step (E-step) as:

$$Q(\bar{\lambda}, \lambda) = Q_1(\bar{\lambda}_1, \lambda_1) + Q_2(\bar{\lambda}_2, \lambda_2) \tag{8}$$

where

$$Q_1(\bar{\lambda}_1, \lambda_1) = \sum_{n=1}^{N} \sum_{p_n=1}^{P} \sum_{t_n} p(p_n, t_n \mid Y_n, \bar{\lambda}_1) \log p(Y_n, p_n, t_n \mid \lambda_1), \tag{9}$$

$$Q_2(\bar{\lambda}_2, \lambda_2) = \sum_{n=1}^{N} \sum_{t_n} p(p_n, t_n \mid \mathbf{Z}_n, \bar{\lambda}_2) \log p(\mathbf{Z}_n, p_n, t_n \mid \lambda_2), \tag{10}$$

N is the total number of training syllables, P is the total number of prosodic states, $p(p_n, t_n \mid Y_n, \bar{\lambda}_1)$, $p(Y_n, p_n, t_n \mid \lambda_1)$, $p(p_n, t_n \mid \mathbf{Z}_n, \bar{\lambda}_2)$ and $p(\mathbf{Z}_n, p_n, t_n \mid \lambda_2)$ are conditional probabilities, $\lambda = \lambda_1 \cup \lambda_2$, $\lambda_1 = \{\mu, \nu, \beta_t, \beta_{pt}, \beta_{ft}, \beta_p\}$ and $\lambda_2 = \{\boldsymbol{\mu}, \mathbf{R}, \mathbf{b}_{pt}, \mathbf{b}_t, \mathbf{b}_{ft}, \mathbf{b}_p\}$ are the sets of parameters to be estimated, and λ and $\bar{\lambda}$ are respectively the new and old parameter sets. Based on the assumption that the normalized pitch mean F_n and shape $\mathbf{X}_n$ are both normally distributed, $p(Y_n, p_n, t_n \mid \lambda_1)$ and $p(\mathbf{Z}_n, p_n, t_n \mid \lambda_2)$ can be derived from the assumed model given in Eqs.(6) and (7) and expressed by:

$$p(Y_n, p_n, t_n \mid \lambda_1) = N(Y_n; \mu + \beta_{pt_n} + \beta_{t_n} + \beta_{ft_n} + \beta_{p_n}, \nu), \tag{11}$$

and

$$p(\mathbf{Z}_n, p_n, t_n \mid \lambda_2) = N(\mathbf{Z}_n; \boldsymbol{\mu} + \mathbf{b}_{pt} + \mathbf{b}_t + \mathbf{b}_{ft} + \mathbf{b}_p, \mathbf{R}) \tag{12}$$

Similarly, $p(p_n, t_n \mid Y_n, \bar{\lambda}_1)$ and $p(p_n, t_n \mid \mathbf{Z}_n, \bar{\lambda}_2)$ can be expressed by:

$$p(p_n, t_n \mid Y_n, \bar{\lambda}_1) = \frac{p(Y_n, p_n, t_n \mid \bar{\lambda}_1)}{\sum_{p'_n=1}^{P} \sum_{t'_n} p(Y_n, p'_n, t'_n \mid \bar{\lambda}_1)}, \tag{13}$$

and

$$p(p_n, t_n \mid \mathbf{Z}_n, \bar{\lambda}_2) = \frac{p(\mathbf{Z}_n, p_n, t_n \mid \bar{\lambda}_2)}{\sum_{p'_n=1}^{P} \sum_{t'_n} p(\mathbf{Z}_n, p'_n, t'_n \mid \bar{\lambda}_2)} \tag{14}$$

Then, sequential optimizations of these parameters can be performed in the maximization step (M-step). At the end of each iteration, the pronounced tone pattern for each syllable is re-assigned to one of its possible patterns by:

$$t_n^* = \arg\max_{t_n} p(t_n \mid Y_n, \lambda_1) p(t_n \mid \mathbf{Z}_n, \lambda_2) \tag{15}$$

To execute the EM algorithm, initialization of the parameter set $\bar{\lambda}$ is needed. This can be done by estimating each individual parameter independently. Specifically, the initial CF for a specific value of an affecting factor is assigned to be the difference of the mean (mean vector) of $Y_n(\mathbf{Z}_n)$ with the affecting factor equaling the value of the mean of all $Y_n(\mathbf{Z}_n)$. Notice that, in the initialization of CFs for prosodic states, each syllable is pre-assigned a prosodic state by vector quantization. After initialization, all parameters are sequentially updated in each iteration. The iterative procedure is continued until a convergence is reached.

5.2 Experimental Results

Performance of the proposed model-based Min-Nan tone labeling method was examined by simulation on the same single-male speaker database used in the Min-Nan TTS system to train and test the RNN prosody generator. Four tone labeling methods were then realized and compared. The first one was the manual approach, which determined the tone sequence to be pronounced by examining the text. Although the results might contain some errors, we still took them as the reference target because of the lack of anything superior. It is referred to as MANUAL. Another two systems were the VQ-based methods which used 4 (mean + shape) and 3 (shape) orthogonal expansion coefficients of syllable pitch contour as classification features, respectively. They are referred to as VQ-4 and VQ-3. The last one was the proposed model-based method and referred to as MODEL. The RMSEs of the reconstructed pitch contour are 0.815 and 0.286 ms/frame for VQ-4 and MODEL, respectively. The superior results of MODEL show the effectiveness of the pitch mean and shape models. Table 5 shows the correct rates of tone labeling for the latter three methods by taking the results of MANUAL as reference target. Correct rates of 50.9, 52.4, and 61.9% were obtained by VQ-4, VQ-3, and MODEL, respectively. Obviously, MODEL outperformed both VQ-4 and VQ-3. It can also be found in Table 5 that Tone 1 and Tone 2, which share a single *sandhi* tone pattern, have better labeling results.

Table 5. The correct rates of the three tone labeling methods of VQ-4, VQ-3, and MODEL. (unit: %)

Tone (*sandhi* tones)	1 (7)	2 (1)	3 (2,1)	4 (2,1,8)	5 (7,3,7)	7 (3,7)	8 (3,7,4)	Ave.
VQ-4	61.9	82.9	55.4	40.9	28.1	34.0	33.9	50.9
VQ-3	58.7	84.8	44.1	28.7	43.7	47.2	35.8	52.4
MODEL	72.4	89.3	51.7	55.7	50.6	51.1	41.9	61.9

By examining all 22 tone patterns obtained in the pitch mean and shape models, we found that most *sandhi* tone patterns matched with those tone patterns suggested by the above-mentioned *sandhi* rules. Figure 5 displays the standard and *sandhi* tone patterns for lexical Tone 1 and Tone 2. Can be seen from Fig. 5(a) (Fig. 5(b)) that the shape of the *sandhi* tone pattern of Tone 1 (2) resembles the standard pattern of Tone 7 (1). Figure 6 displays pitch contour patterns of standard and *sandhi* tones for Tone 3 and Tone 2. It can be seen from Fig. 6(a) (Fig. 6(b)) that all three (two) *sandhi* Tone 3 (2) patterns resemble to the standard Tone 3 (2) pattern.

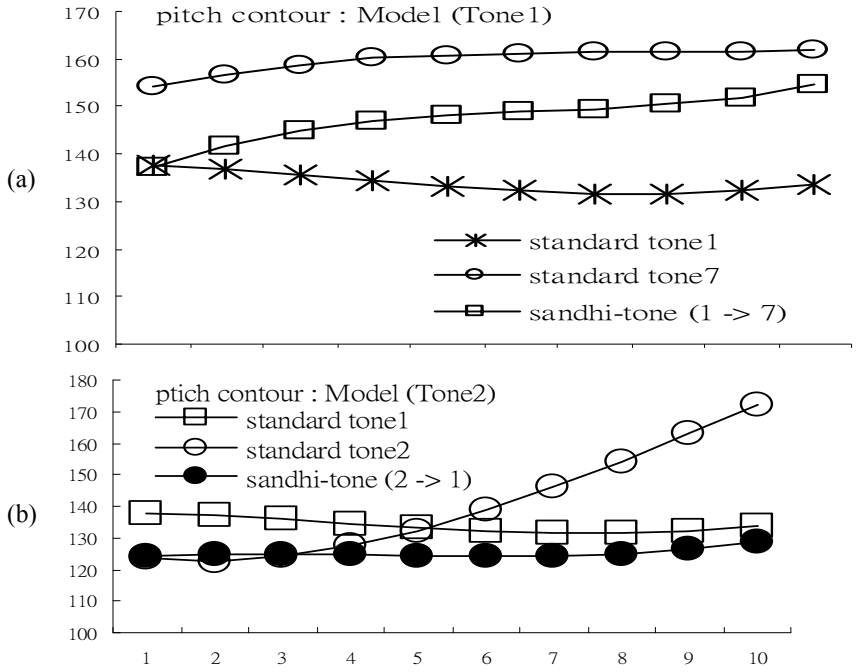


Figure 5. Comparison of standard and sandhi tone patterns for lexical (a) Tone 1 and (b) Tone2.

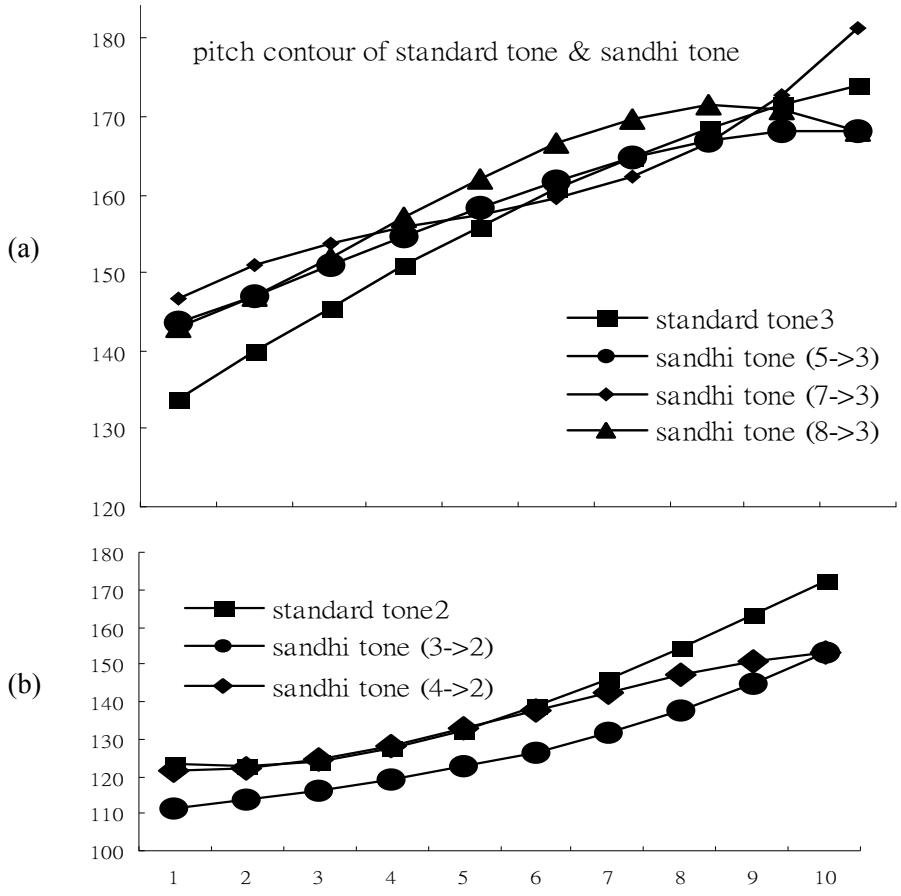


Figure 6. Comparison of pitch contour patterns of standard tone & sandhi tones for (a) Tone 3 and (b) Tone 2.

6. Conclusions

In this paper, three studies of Min-Nan speech processing have been discussed. They included the implementation of a high-performance Min-Nan TTS system, the realization of a Min-Nan speech recognizer, and a model-based tone labeling method. Experimental results confirmed that all proposed methods are promising.

From these studies, we find that the most important factor to affect the research results is the database. Basically, a large, phonetically-rich, high-quality speech database with text being properly annotated is needed. The two databases used in current three studies are still not perfect on their size and text annotation. To improve the quality of these two databases for achieving a good progress on our future Min-Nan speech processing studies are therefore worth doing.

ACKNOWLEDGEMENT

This work was supported in part by MOE under contract EX-94-E-FA06-4-4. The authors thank Prof. R. L. Cheng and Prof. Y. C. Chiang for supplying the lexicon and the text corpus.

REFERENCES

Chen, S. H., and C. C. Ho, "An Implementation of Taiwanese Text-to-Speech System," In *Proceedings of ISCSLP'2000*, 2000, Beijing, vol.1, pp. 613-616.

Chen, S. H., S. H. Hwang, and Y. R. Wang, "An RNN-Based Prosodic Information Synthesizer for Mandarin Text-to-Speech," *IEEE Trans. Speech and Audio Processing*, 6(3), 1998, pp. 226-239.

Chen, S. H., W.-H. Lai, and Y.-R. Wang, "A statistics-based pitch contour model for Mandarin speech," *J. Acoust. Soc. Am.*, 117(2), 2005, pp. 908-925.

Chen, S. H., and Y. R. Wang, "Vector Quantization of Pitch Information in Mandarin Speech," *IEEE Tarns. Communications*, 38(9), 1990, pp. 1317-1320.

Cheng, R. L., *Taiwanese pronunciation and Romanization – with rules and examples for teachers and students*, Wang Wen Publishing Company, Taipei, 1993.

Haykin, S., *Neural networks – A comprehensive foundation*, Macmillan College Publishing Company, 1994.

Ho, C. C., and S. H. Chen, "A Hybrid Statistical/RNN Approach to Prosody synthesis for Taiwanese TTS," In *Proceedings of ICSLP'2000*, 2000, Beijing.

Ho, C. C., and S. H. Chen, "A Maximum Likelihood Estimation of Duration Models for Taiwanese Speech," In *Proceedings of ISAS-SCI 2000*, Orlando, USA, vol. VI, pp. 395-399.

Huang, S.-F., *Language, Society and Ethnici*ty, 2nd ed.. Crane, Taipei, 1995

Huang, J. Y., "Implementation of Tone *Sandhi* Rules and Tagger for Taiwanese TTS," Master Thesis, Communication Eng. Dept., National Chiao Tung University, 2001.

Kuo, W.-C., Y.-R. Wang, and S.-H. Chen, "A Model-Based Tone Labeling Method for Min-Nan/Taiwanese Speech," In *Proceedings of ICASSP2004*, 2004, Montreal, Canada, Vol. 1, pp. 505-508.

Kuo, W.-C., X.-R. Zhong, Y.-R. Wang, and S.-H. Chen, "A High-Performance Min-Nan/Taiwanese TTS System," In *Proceedings of ICASSP2003*, 2003, Hong Kong, Vol. 1, pp. 512-515.

Li, A., "Chinese Prosody and Prosodic Labeling of Spontaneous Speech," In *Proceedings of Speech Prosody 2002*, Aix-en-Provence, France, 2002.

Liang, M.-S., R.-C. Yang, Y.-C. Chiang, D.-C. Lyu, and R.-Y. Lyu, **"**A Taiwanese text-to-speech system with applications to language learning," In *Proceedings of 2004 IEEE International Conference on Advanced Learning Technologies*, 2004, pp. 91-95.

Lyu, R.-Y., Y.-C. Chiang, W.-P. Hsieh, and R.-Z. Fang, "A Large-Vocabulary Speech Recognition System for Taiwanese (Min-nan)," *Journal of the Chinese Institute of Electrical Engineering*, 7(2), 2000, pp. 123-136.

Lyu , D.-C., B.-H. Yang, M.-S. Liang, R.-Y. Lyu, and C.-N. Hsu, "Speaker independent acoustic modeling for large vocabulary bi-lingual aiwanese/Mandarin continuous speech recognition," In *Proceedings of the ninth Australian international conference on Speech science and technology*, 2002, Melbourne, pp. 28-33.

Wang, W. J., Y. F. Liao, and S. H. Chen, "RNN-based Prosodic Modeling for Mandarin Speech and Its Application to Speech-to-Text Conversion," *Speech Communication,* 36, 2002, pp. 247-265.

Yang, Y. C., "An Implementation of Taiwanese Text-to-Speech System," Master Thesis, Communication Eng. Dept., National Chiao Tung University, Hsinchu, 1999.

Young, S., ATK: A Application Tool for HTK, http://mi.eng.cam.ac.uk/~sjy/software.htm, 2007.

Yuan, J. H., Hanyu Fangyan Gaiyao, *Outline of Chinese Dialects*, 2nd ed., Wenzi Gaige Chubanshe, Beijing, 1989.

Computational Linguistics and Chinese Language Processing
Vol. 12, No. 4, December 2007, pp. 411-442

Construction and Automatization of a Minnan Child Speech Corpus with some Research Findings

Jane S. Tsay*

Abstract

Taiwanese Child Language Corpus (TAICORP) is a corpus based on spontaneous conversations between young children and their adult caretakers in Minnan (Taiwan Southern Min) speaking families in Chiayi County, Taiwan. This corpus is special in several ways: (1) It is a Minnan corpus; (2) It is a speech-based corpus; (3) It is a corpus of a language that does not yet have a conventionalized orthography; (4) It is a collection of longitudinal child language data; (5) It is one of the largest child corpora in the world with about two million syllables in 497,426 lines (utterances) based on about 330 hours of recordings. Regarding the format, TAICORP adopted the Child Language Data Exchange System (CHILDES) [MacWhinney and Snow 1985; MacWhinney 1995] for transcribing and coding the recordings into machine-readable text. The goals of this paper are to introduce the construction of this speech-based corpus and at the same time to discuss some problems and challenges encountered. The development of an automatic word segmentation program with a spell-checker is also discussed. Finally, some findings in syllable distribution are reported.

Keywords: Minnan, Taiwan Southern Min, Taiwanese, Speech Corpus, Child Language, CHILDES, Automatic Word Segmentation

1. Introduction

Taiwanese Child Language Corpus is a corpus based on spontaneous conversations between young children and their adult caretakers in Minnan speaking families in Chiayi County, Taiwan. This corpus is special in several ways. First, it is a Minnan corpus. Minnan is Southern Min Chinese spoken in Taiwan (also known as Taiwanese in linguistic literature). It is less studied, especially when compared with Mandarin Chinese. Second, it is a speech-based corpus. The scripts in the corpus were transcribed from recordings of

* Institute of Linguistics, National Chung Cheng University, 168 University Road, Min-hsiung, Chiayi County, Taiwan 62102, ROC Phone: 886-5-2720411 ext. 31502 Fax: 886-5-2721654
E-mail: Lngtsay@ccu.edu.tw

[Received November 1, 2006; Revised August 14, 2007; Accepted August 16, 2007]

spontaneous speech. Third, it is a corpus of a language that does not yet have a conventionalized orthography. Fourth, it is a child corpus. It's a collection of longitudinal child language data. Fifth, it is currently one of the largest child corpora in the world. It contains about 2 million syllables/characters in 497,426 lines (utterances) based on about 330 hours of recordings. Finally, it is a corpus that uses an international platform. This platform is the Child Language Data Exchange System (CHILDES) [MacWhinney and Snow 1985; MacWhinney 1995] for transcribing and coding the recordings into machine-readable text.

The goals of this paper are:

(1) to introduce the construction of this speech-based child language corpus, TAICORP (Section 2);

(2) to introduce the automatization process of this corpus and discuss some issues encountered during the implementation of the system (Section 3);

(3) to present some research findings based on this corpus (Section 4).

2. Taiwanese Child Language Corpus

Taiwanese Child Language Corpus (TAICORP) contains scripts transcribed from about 330 hours of spontaneous speech from fourteen young children acquiring Taiwan Minnan as their first language. A brief introduction to this corpus was reported at the 5th Workshop on Asia Language Resources [Tsay 2005a]. In this extended paper, in addition to a more detailed description and more discussion about the corpus and related issues, findings in syllable type distribution and tone type distribution are also presented.

There are about 1.6 million words (over 2 million syllables/characters) in this corpus, as shown in Table 1.

Table 1. The size of TAICORP

	Lines (utterances)	Words	Syllables	
			Syllables (in words) 1,558,408	Syllables (in particles) 538,992
Total	497,426	1,646,503	2,097,400	

Since some words do not have corresponding Chinese characters and are presented in romanization notation (Minnan Pinyin) in this corpus, the syllable might be a more precise unit than the more traditional unit *zi* 字 (Chinese character).

Note that we divide the syllables into two categories: syllables in words (*e.g.*, chia 車) and syllables in particles (*e.g.*, la 啦). Among all the 2,097,400 syllables, 538,992 syllables (about 26%) are in particles. This is a very interesting fact and will be discussed in more detail in Section 4.

In this section, TAICORP is introduced in the following aspects:

2.1. Motivation

2.2. Data collection

2.3. Text files in CHILDES format

2.4. Transcribing sound files into text files

2.5. Annotations

2.1 Motivation

From the linguistics point of view, there is an urgent need to construct a Minnan child language corpus, partly because there has not been any such corpus available and partly because it may be getting more and more difficult to find young children learning Minnan as their first language, especially in the cities. On top of that, the significance of a large collection of longitudinal child language data for linguistic studies goes beyond saying.

Mandarin and Minnan are the two major Chinese languages in Taiwan. For over forty years, Mandarin was the only official language for instruction at school in spite of the fact that about 73% of the population belonged to the Minnan ethnic group [Huang 1993]. Young children in kindergartens and elementary schools were not allowed to speak Minnan even if Minnan was the language spoken at home. This policy caused a decrease in the number of young children learning Minnan as their first language.

Although the situation has changed in recent years and other local languages besides Mandarin, including Minnan, Hakka, and the aboriginal (Formosan) languages have been included in the curriculum of elementary schools, there is still a serious concern about the decrease of native Minnan speakers. This concern can be supported by a more recent survey. Tsay [2005] reports that in a survey of all 8th graders in Chiayi City in Southern Taiwan, an area where the population should be overwhelmingly Minnan, only about 26% of 14 year-olds used Minnan in their daily life, although over 80% of their grandparents and over 70% of their parents were native Minnan speakers.

Under this consideration, Minnan was chosen as the target language. The project was conducted in a rural area in Chiayi County in Southern Taiwan with the hope to find young children who were raised in a Minnan-speaking environment.

2.2 Data collection

Data collection took place over a period of around three years between August 1997 and July 2000 under the support of the National Science Council in Taiwan (NSC 87-2411-H-194-019, NSC 88-2411-H-194-019, NSC 88-2418-H-194-002).

Child participants

Young Children from Minnan-speaking families were recruited in Min-hsiung Village, Chiayi County, in Southern Taiwan. Nine boys and five girls from the following villages in Min-hsiung Xiang participated in this project: Fengshou（豐收村）, Sanxing（三興村）, Dongxing（東興村）, Xidibu（溪底部）, and Zhenbei（鎮北村）. They aged from one year and two months (1;2) to three years and eleven months (3;11) old at the beginning of the recording. More than half of the children were recorded over more than two years. The age range at the offset of the recordings is between 2;7 and 5;3.

Recording

Regular home visits were conducted every two weeks for younger children and every three weeks for children older than three years old. The recording setup was children at play at home interacting naturally with the adult(s), usually one of their caretakers (parents, grandparents, or, in very few cases, the nanny) and/or the investigator. The activities were children's daily life at home: playing with toys or games, reading picture books, or just talking without any specific topics. Since we hoped to have the most natural environment, Mini-disc recorders and microphones were used so that it was easier for the recorder (the investigator) to follow the child wherever she/he went. Usually, each recording session lasted from 40 to 60 minutes.

Information about the child participants and the recordings is given below.

Table 2. Recording Information of TAICORP

Name	Sex	Age range	Sessions	length (min.)
YDA	M	3;11.02 – 4;04.26	9	540
YCX	M	3;10.16 – 4;00.16	6	285
LJX	M	3;09.20 – 4;02.24	8	530
CQM	M	2;09.07 – 4;06.22	30	1584
LMC	F	2;08.07 – 5;03.21	50	2045
YJK	M	2;06.11 – 2;0626	2	105
CEY	F	2;01.27 – 3;10.00	37	1728
HBL	M	2;01.22 – 4;00.03	45	1889
LWJ	F	2;01.08 – 3;07.03	36	1777
WZX	M	2;01.17 – 4;03.15	44	1757
YSW	M	1;07.17 – 2;07.14	21	1210
TWX	F	1;05.12 – 3;06.15	44	1829
HYS	M	1;02.28 – 3;04.12	51	2280
LYC	F	1;02.13 – 3;03.29	48	2255
Total	M=9 F=5		431	about 330 hours

Sound file editing

There were a total of 431 recording sessions. Each session was saved as a separate sound file. The sound files were first edited so that the empty or noisy parts could be cleared. In order to have easier searching and locating the content of the recordings, each sound file was segmented into several tracks and the tracked marks were tagged.

2.3 Text Files in CHILDES Format

The sound files were transcribed into text files in CHILDES format. CHILDES (Child Language Data Exchange System) was originally set up by Elizabeth Bates, Brian MacWhinney, and Catherine Snow to transcribe and code recordings into machine-readable speech text [MacWhinney and Snow 1985; MacWhinney 1995].

CHILDES has been widely accepted as the standard system for child language data. TAICORP adopted the format of CHILDES so that it will be easy to exchange and share data with researchers around the world. CHILDES includes a transcription system, CHAT, and a set of programs, CLAN, for various analyses. In this section, we introduce a simplified version of the format of text files in CHAT. For details, please refer to MacWhinney [1995] or the official website of CHILDES at http://childes.psy.cmu.edu/.

The main components of the CHILDES format are headers and tiers.

Headers

There are three kinds of headers: obligatory headers, constant headers, and changeable headers.

> ***Obligatory headers***: Obligatory headers are necessary for every file. They mark the beginning, the end, and the participants of the file.
>
> ***Constant headers***: They mark the name of the file and the background information of the children.
>
> ***Changeable headers***: They contain information that may change across files, such as the recording date, duration, coders, and so on.

These headers all begin with @. Some examples are given below:

Obligatory headers:

@Begin	to mark the beginning of a file
@End	to mark the end of a file
@Participants	to list all the participants in a file

Constant headers:

@Age of XXX:	the age of speaker
@Birth of XXX:	the birthday of the speaker
@Coder:	the file coder's name
@Educ of XXX:	the highest education of the speaker
@Filename:	filename
@Language:	the main language used in the file
@Language of XXX:	the language used by the speaker
@Sex of XXX:	the sex of the speaker
@Warning:	the defects of the file

Changeable headers (optional):

@Activities:	Activities involved in the situation
@Bck:	background information of the utterance
@Comment:	the comment of the investigator
@Date:	the date of the interaction
@G:	gems
@Location:	the location of the interaction
@New Episode:	the new episode of the recording starts
@Room Layout:	room configuration and positioning of furniture
@Situation:	the situation of the interaction
@Tape Location:	the specific ID, side and footage
@Time Duration:	the length of recording time
@Time Start:	the starting time of recording

Tiers

The content of a file is presented in tiers in CHILDES. There is a main tier and several dependent tiers for each line (utterance).

The main tier, marked with *, is the speech of the speaker. Three capital letters indicate the status of the speaker, *e.g.*, *CHI is the child, *MOT the mother, and *INV the investigator.

Minnan Pinyin is used in the Main tier. Words are separated by a space. Therefore, an utterance "I want to water the vegetables" from a child would be:

*CHI:	gua2	beh4	ak4	chai3.
	I	want	water	vegetable

The additional information is given in dependent tiers that are marked with % at the beginning of a new line. The seven dependent tiers used in TAICORP are given below.

%ort:	the utterance in logographic orthography (*i.e*., Chinese characters)
%pro:	the actual target pronunciation of the utterance (dialectal variation)
%syl:	syllable type coded with C and V (*e.g*. CVV for /gua/)
%cod:	part-of-speech coding
%pho:	phonetic transcription in Unicode IPA (for child speech only)
%syc:	syllable type of the child's pronunciation
%ton:	tone value in 5-digit scale

For the adult speech, there are only four dependent tiers: %ort, %pro, %syl, and %cod because no phonetic transcription was done on the adult speech. For the child speech, there are up to seven dependent tiers as shown in the following example.

(main tier)	***CHI:**	**gua2**	**beh4^bueh4**	**ak4**	**chai3.**
(depnt tier)	%ort:	我	欲	沃	菜.
(depnt tier)	%pro:	gua2	beh4	ak4	chai3.
(depnt tier)	%syl :	CVV	CVK	VK	CVV
(depnt tier)	%cod:	Nh	D	VA	
(depnt tier)	%pho:	guaŋ	be	ak	t'ai
(depnt tier)	%syc:	CVVN	CV	VK	CVV
(depnt tier)	%ton:	55	55	5	21

2.4 From Sound Files to Text Files

All sound files were transcribed into text files. Transcriptions included (1) orthographic transcription; and (2) phonetic transcription (in IPA, International Phonetic Alphabet).

There were two kinds of systems used in orthographic transcription. One was the logographic orthography (*i.e.*, traditional Chinese writing system Hanzi 漢字), and the other was a spelling-based romanization system for Minnan (called Minnan Pinyin). Thus, each sound file was transcribed into a separate text file in both Chinese characters and Minnan Pinyin.

2.4.1 Orthographic Transcription in Chinese Characters

The reason that the sound files were first transcribed into Chinese characters was because this written form is closest to most native speakers' intuition. Therefore, by transcribing [tsetɕʰia] into "坐車", it makes it much easier for the user to read.

Although romanization notation (Minnan Pinyin) in the Main tier (*e.g.*, *CHI tier in the above example) makes it easier to run the analyzing programs in CHILDES and might also be easier for non-Chinese users of the corpus, having a tier with Chinese characters would be more convenient for those who know Chinese. Therefore, a dependent tier %ort was added to present the utterances in Chinese characters. This is a reasonable method because most Minnan words are cognates of Mandarin words. Still, there are quite a few words that either do not have their corresponding Chinese characters or their corresponding Chinese characters are so obsolete that they cannot be found in the software for typing Chinese characters.

Since Minnan does not have as conventionalized orthography as Mandarin, quite a few words in Minnan do not have a consistent way of writing them. In order to help build consensus in Minnan cognates（閩南語本字）, Minnan dictionaries were consulted. At least seven dictionaries were used as listed after the references.

There are several possibilities regarding Chinese characters used in Minnan:

First, they are exactly the same as those used in Mandarin, for example, 色筆/sik4pit4/ "color pens".

Second, they are synonyms of Mandarin words, but use different characters, for example, 挽 /ban2/ "pluck; pick up" is a synonym of Mandarin 摘 /zhai1/ or 採 /cai3/; 鼻芳 /phinn7phang1/ "smelling the fragrance" is a synonym of 聞香 /wen2xiang2/.

Third, although the Chinese characters in Minnan can be found in the dictionary, they might be so obsolete that one has to use special software to make the character forms, as in the first character of the following word meaning "good morning".

This is very inconvenient for users and is very hard to process, too. In such cases, Minnan Pinyin is used and the above word would be presented as *gau5 早*.

Fourth, when Chinese characters cannot be found at all for Minnan words, Minnan Pinyin is used, as in the first morpheme is the word *chua7 路* /chua7loo7/ "leading the way" or *chit4tho5* /chit4tho5/ "playing around".

For homonyms that share the same Chinese character, a number is added to the character to indicate different lemmas. For example:

蓋 1 /kah4/ "to cover with a blanket"

蓋 2 /kham3/ "to cover"

蓋 3 /kua3/ "a cover/lid"

2.4.2 Orthographic Transcription in Minnan Pinyin

The reason for transcribing the sound files into Minnan Pinyin was twofold: (1) to encode the sounds in a spelling system, and (2) to make it easier for the machine (computer program) to read and to do analyses such as syllable frequency counts.

The Minnan Pinyin system used in TAICORP is the Taiwan Southern Min Phonetic Alphabetic officially announced by the Ministry of Education in Taiwan in 1998.[1] Like most romanization systems, the Minnan Pinyin system labels sounds at the phonemic level.

The Minnan Pinyin notation system with examples is given in Table 3 (consonants) and Table 4 (vowels) below. Note that '-' before a symbol indicates the coda position, as in a checked (Rusheng) syllable. It is necessary to make such a distinction because of the asymmetry in the distribution of consonants. For example, [b] cannot occur in the coda position, although it can occur in the onset position. Following the IPA convention, a dot under a symbol is used to denote a syllabic consonant. Nasal vowels are denoted with "nn". Therefore, the word [tĩ] "sweet" is transcribed as /tinn/ in this system.

[1] The system released by the Ministry of education adopted the Taiwan Language Phonetic Alphabet (TLPA) originally proposed by Taiwan Languages Society in 1994. Revisions can easily be made if it becomes necessary.

Table 3. Minnan Pinyin System (Consonants)

Minnan Pinyin	IPA	Example	Glossary
p	p -p	pit4 筆 ciap4 汁	pen juice
ph	p^h	phue5 皮	skin
b	b	be2 馬	horse
m	m -m m̩	moo1 毛 sim1 心 a1m2 阿姆	fur heart aunt
t	t -t	to1 刀 that4 踢	knife kick
th	t^h	thau5 頭	head
l	l	lai5 來	come
n	n -n	ni5 年 sin1 新	year new
k	k -k	kau2 狗 kak4 角	dog horn
kh	k^h	kha1 跤	foot
g	g	gu5 牛	cow
ng	ŋ -ŋ ŋ̍	nge7 硬 sing1 升 ng5 黃	hard ascend yellow
h	h -ʔ	hue1 花 bah4 肉	flower meat
c	ts tɕ	cu2 煮 cit8 一	cook one
ch	ts $tɕ^h$	chai3 菜 chit4 七	vegetable seven
s	s ɕ	sai1 獅 si3 四	lion four
j	z	jit8 日	sun

Table 4. Minnan Pinyin System (Vowels)

Minnan Pinyin	IPA	Example	Glossary
i	i	ti1 豬	pig
e	e	be2 馬	horse
a	a	ka7 咬	bite
oo	ɔ	koo1 姑	aunt
o	o/ə	to1 刀	knife
u	u	gu5 牛	cow
inn	ĩ	tinn1 甜	sweet
enn	ẽ	chenn1 星	star
ann	ã	sann1 三	three
onn	ɔ̃	honn3ki5 好奇	curious
ia	ia	khia7 倚	stand
io	io/iə	kio5 橋	bridge
iu	iu	kiu5 球	ball
iann	ĩã	kiann5 行	walk
iunn	ĩũ	kiunn1 薑	ginger
ai	ai	lai5 來	come
au	au	chau2 草	grass
ainn	ãĩ	phainn2 歹	bad
ui	ui	cui2 水	water
ue	ue	hue2 火	fire
ua	ua	kua1 歌	song
uann	ũã	suann3 線	string
iau	iau	iau1 枵	hungry
uai	uai	kuai1 乖	submissive
uainn	ũãĩ	kuainn1 關	close
uinn	ũĩ	khuinn3uah8 快活	joyful

There are seven lexical tones in Minnan spoken in Chiayi, Taiwan. These tone categories are denoted by digits 1 to 8, except for Tone 6 Yangshang (陽上) which has been merged into other tone categories due to historical sound change. Morphemes (or syllables) without underlying tones are marked with '0'. Interjections and particles, which do not have an underlying tone and their surface tones might vary due to different contexts, are all marked with '0', for example, a0 啊, le0 咧. Loan words, for example, too0sang0 多桑, borrowed from the Japanese word for "father", are also marked with the '0' tone category. Tones deviating from the seven lexical tones are categorized into the '9' tone category, for example, tones derived by syllable concatenation, bo5iau3kin2 → bua9kin2 不要緊 "not matter".

Table 5. Minnan Tones

Tone Category		Example	Glossary
0	toneless	oo0 哦	(interjection)
1	Yinping 陰平	si1 詩	poem
2	Yinshang 陰上	si2 死	death
3	Yinqu 陰去	si3 四	four
4	Yinru 陰入	sik4 色	color
5	Yangping 陽平	si5 時	time
7	Yangqu 陽去	si7 寺	temple
8	Yangru 陽入	sik8 熟	ripe
9	others	bua9kin2 不要緊	not-matter

2.4.3 Phonetic Transcription in IPA

As mentioned above, Minnan Pinyin is a notation system at the phonemic level. The adult speech is considered the target as well as the input of the child language. We assume that the adult speech is "standard", and no phonetic transcription was done for the adult speech due to the limitation of manpower. In general, it is appropriate to represent the adult speech phonemically, unless one wants to know the allophonic variation or idiosyncratic characteristics of the adult speakers. In those cases, detailed phonetic transcription would be required.

However, we are most concerned with the child speech. The most important aspect in child language is its deviation from the ambient adult speech. Therefore, narrow phonetic transcription has to be available to understand the pattern and development of child language.

Narrow phonetic transcription was conducted for sound files of children under two and a half years old using Unicode IPA. The following are two sample utterances from the child

WZX at 2;1.17. The child's segmental pronunciation is shown in the %pho (phonetic) tier, and tonal pronunciation is shown in %ton (tone) tier using a 5-point scale. Note that the child's pronunciation was different from the standard speech of the adult. For example, /gua/ "I" was pronounced as [ua], /cing/ [tsiŋ] "plant" was pronounced as [t'iŋ], etc. This is to record truthfully what the child actually said. Such data are very important for studying children's phonological development.

Example 1

*CHI:	gua2	gua2	koh4	peh4
	我…	我	擱	擘
	I	I	again	split
	"I want to split it again."			
%pho:	ua	ua	kəʔ	pe
%ton:	55	55	4	32

Example 2

*CHI:	he1	a1po5	cing3	e0
	彼	阿婆	種	e
	that	grandma	plant	(Relative clause marker)
	"That was planted by grandma."			
%pho:	he	aʔ pə	t'iŋ	ẽ
%ton:	44	3 55	21	21

2.5 Annotations

Two kinds of annotations are described in this section: part of speech (POS) annotations and discourse annotations.

2.5.1 Part of Speech Annotation

Minnan and Mandarin are both Sinitic languages and are very similar in their morphology and syntactic structures. Therefore, the POS coding system of a Minnan corpus should be very similar to that of the Sinica Corpus of Mandarin (see various technical reports by the Chinese Knowledge Information Processing Group (CKIP) [CKIP 1993, 1998; Chen *et al*. 1996]. There are a total of 46 codes listed as simplified codes and 115 corresponding codes for Mandarin in Sinica Corpus [CKIP 1998].

Table 6. Tagset in Sinica Corpus [CKIP 1998]

Simplified codes (total 46 codes)	Corresponding CKIP codes (total 115 codes)
A	A
Caa	Caa
Cab	Cab
Cba	Cbab
Cbb	Cbaa, Cbba, Cbbb, Cbca, Cbcb
Da	Daa
Dfa	Dfa
Dfb	Dfb
Di	Di
Dk	Dk
D	Dab, Dbaa, Dbab, Dbb, Dbc, Dc, Dd, Dg, Dh, Dj
Na	Naa, Nab, Nac, Nad, Naea, Naeb
Nb	Nba, Nbc
Nc	Nca, Ncb, Ncc, Nce
Ncd	Ncda, Ncdb
Nd	Ndaa, Ndab, Ndc, Ndd
Neu	Neu
Nes	Nes
Nep	Nep
Neqa	Neqa
Neqb	Neqb
Nf	Nfa, Nfb, Nfc, Nfd, Nfe, Nfg, Nfh, Nfi
Ng	Ng
Nh	Nhaa, Nhab, Nhac, Nhb, Nhc
I	I
P	P*
T	Ta, Tb, Tc, Td
VA	VA11, 12, 13, VA3, VA4
VAC	VA2
VB	VB11, 12, VB2
VC	VC2, VC31, 32,33
VCL	VC1
VD	VD1, VD2
VE	VE11, VE12, VE2
VF	VF1, VF2
VG	VG1, VG2
VH	VH11, 12, 13, 14, 15, 17, VH21
VHC	VH16, VH22
VI	VI1, 2, 3
VJ	VJ1, 2, 3
VK	VK1, 2
VL	VL1, 2, 3, 4
V_2	V_2
DE	/的, 之, 得, 地/
SHI	/是/
FW	/外文標記/

To avoid arbitrary classification of words into the morpho-syntactic categories, we adopted the simplified version with 46 morph-syntactic codes, instead of the finer 115 categories used in the Sinica Corpus. In other words, categorization in TAICORP is broader. These codes (tagset) are listed in the table below.

Table 7. Tagset of TAICORP

Tagging	POS	POS (Chinese terms)
A	non-predicative adjective	非謂形容詞
Caa	coordinate conjunction	對等連接詞
Cab	listing conjunction	連接詞
Cba	conjunction occurring at the end of a sentence	連接詞
Cbb	following a subject	關聯連接詞
Da	possibly preceding a noun	數量副詞
Dfa	preceding VH through VL	動詞前程度副詞
Dfb	following adverb	動詞後程度副詞
Di	post-verbal	時態標記
Dk	sentence initial	句副詞
D	adverbial	副詞
Na	common noun	普通名詞
Nb	proper noun	專有名稱
Nc	location noun	地方詞
Ncd	localizer	位置詞
Nd	time noun	時間詞
Neu	numeral determiner	數詞定詞
Nes	specific determiner	特指定詞
Nep	anaphoric determiner	指代定詞
Neqa	classifier determiner	數量定詞
Neqb	postposed classifier determiner	後置數量定詞
Nf	classifier	量詞
Ng	postposition	後置詞
Nh	pronoun	代名詞
I	interjection	感嘆詞
P	preposition	介詞

Tagging	POS	POS (Chinese terms)
T	particle	語助詞
VA	active intransitive verb	動作不及物動詞
VAC		動作使動動詞
VB	active pseudo-transitive verb	動作類及物動詞
VC	active transitive verb	動作及物動詞
VCL	transitive verb taking a locative argument	動作接地方賓語動詞
VD	ditransitive verb	雙賓動詞
VE	active transitive verb with sentential object	動作句賓動詞
VF	active transitive verb with VP object	動作謂賓動詞
VG	classificatory verb	分類動詞
VH	stative intransitive verb	狀態不及物動詞
VHC	stative causative verb	狀態使動動詞
VI	stative pseudo-transitive verb	狀態類及物動詞
VJ	stative transitive verb	狀態及物動詞
VK	stative transitive verb with sentential object	狀態句賓動詞
VL	stative transitive verb with VP object	狀態謂賓動詞
V_2		有
DE	*special tag for the word "的"	的
SHI	special tag for the word "是"	是
FW	foreign words	外文標記
*Di/T	*marker following pseudo-transitive active verb	*le01
*CIT	*special tag for the word "得 2"	*得 2
*Comp	*complementizer	*補語連詞

2.5.2 Discourse Annotations

The texts in TAICORP are based on spontaneous conversations. Therefore, it is necessary to have discourse annotations. As a speech-based corpus, it is full of incomplete, repeated, repaired, and interrupted utterances. We tried to code these in the scripts. Since discourse analysis is not the primary focus of this paper, we only list some the discourse codes that were used in TAICORP.

(1) Codes for unidentifiable material

(a) xxx/xx: unintelligible speech (utterance/word).

(b) yyy/yy: unintelligible speech at the phonetic level.

(c) www/ww: untranscribed speech to be used in conjunction with a note to explain the situation

(2) Repetition

[/]: repetition of either one or more words

(3) Basic utterance terminators

The basic utterance terminators are the period, the question mark, and the exclamation mark. Each utterance must end with one of these three utterance terminators.

(4) Special utterance terminators: these terminators all begin with the + symbol and end with one of the three basic utterance terminators. For example,

(a) +… Incomplete but not interrupted utterance

(b) +/. Incomplete utterance due to interruption

(c) +//. Self-interruption: breaking off an utterance and starting up another by the same speaker

(d) +?. Interruption of a question: the utterance being interrupted is a question

(e) +, Self-completion: to mark the completion of an utterance after an interruption

(5) Scoped symbols

(a) [=! text] Paralinguistic material: marking paralinguistic events or actions, such as coughing, laughing, telling, crying, singing, and whispering.

(b) [>] Overlap follows

(c) [<] Overlap precedes

(d) [/] Retracing without correction

(e) [//] Retracting with correction

The following is a sample of discourse coding in TAICORP

@Begin

@Participants: CHI Lin Target_Child, INV Rose Investigator

@Age of CHI: 2;9.22

@Birth of CHI: 28-AUG-1995

@Coder: Rose, Kay, Joyce
@Filename: HBL17ipa
@Language: Taiwanese
@Sex of CHI: Male
@Date: 19-JUN-1998
@Tape Location: Lin D4-30-41
@Comment: Time Duration: 35 minutes
@Location: Chiayi, Taiwan
@Transcriber: Rose
@Comment: Track number is D4-30

*INV: a1lin5 [/] a1lin5, li2 koh4 kong2 cit8 kai2.
%ort: 阿林 [/] 阿林, 你 擱 講 一 1 改.
%cod: Nb Nb Nh D VE Neu Nf

*INV: <li2 kong2> [//] li2 thau5tu2a2 kong2 a1ma2 khi3 toh4?
%ort: <你 講> [//] 你 頭拄仔 講 阿媽 去 陀?
%cod: Nh VE Nh Nd VE Na VCL Ncd

*CHI: khi3 sio1kim1 la0.
%ort: 去 燒金 la0.
%cod: VCL VA T
%pho: i t,j i o t,j i,n ng ng a,n
%ton: 55 33 55 21

*INV: hann0/hannh0?
%ort: hann0?
%cod: I

*CHI: khi3 sio1kim1.
%ort: 去 燒金.
%cod: VCL VA
%pho: kh i t,c\ i o t,c\ i ng

%ton: 55 33 55
*INV: khi3 sio1hiunn1 oo0?
%ort: 去 燒香 oo02?
%cod: VCL VA

*CHI: li2 bo5 +/.
%ort: 你 無 1 +/.
%cod: Nh D
%pho: i b o
%ton: 55 33

*INV: a0 i1 u7 cah4 <sann2mih8hue3 khi3> [>]?
%ort: a01 伊 有 紮 <啥物貨 去> [>]?
%cod: Dk Nh V_2 VC Nep VCL

3. Automatization

Constructing a speech-based corpus requires a lot more steps than constructing a corpus based on written texts. The most labor-intensive and time-consuming work is devoted to transcribing the sound files into text files. In the first stage of the construction of TAICORP, every step was done manually. These steps are shown in Figure 1 below.

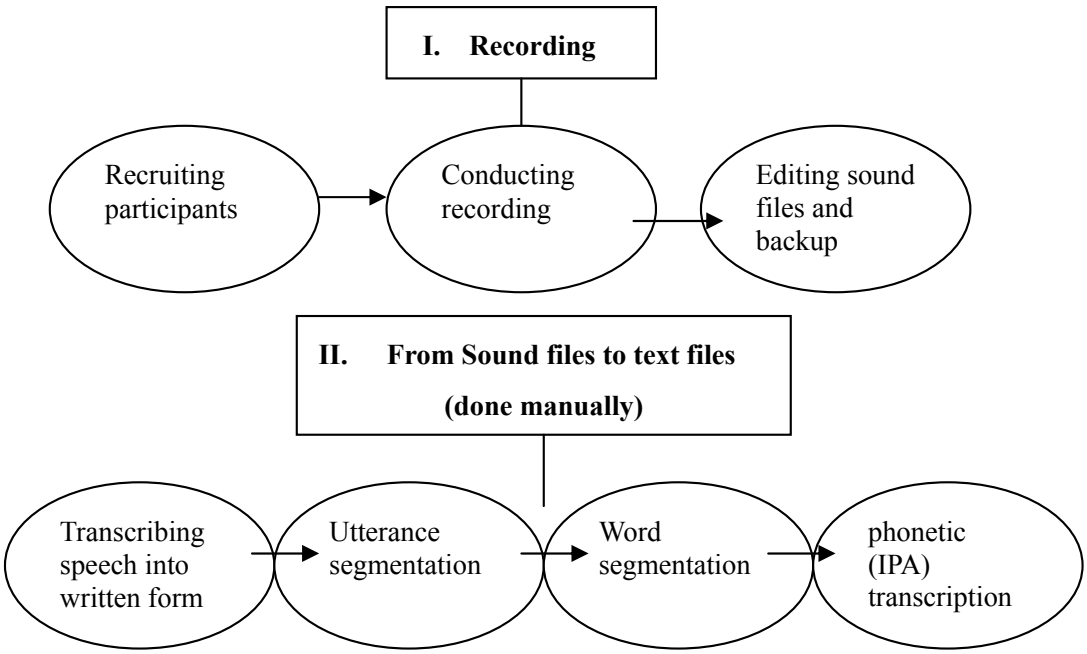


Figure 1. Steps in manual construction

3.1 Automatic Word Segmentation

After all the hard work, it was hoped that the corpus could contribute to the automatization of the procedure. Under this consideration, an automatic word segmentation program has been developed.[2] As the basis of the automatic word segmentation program, a corpus-based lexicon has been constructed manually, which includes the lexical item (both in Minnan Pinyin and in Chinese characters), alternative forms, synonyms, and part-of-speech labels.

Thus, the lexical bank contains the following information for each lexical item:

Logographic orthography: the word in Chinese characters

Spelling-based orthography: the word in Minnan Pinyin

Part-of-speech: the POS coding of the word

Alternative forms/synonyms: alternative written forms of the word

Since the orthography convention has not reached consensus in the Minnan-speaking community, the transcribers might not be consistent in their uses of the written form. Their non-standard uses of the written form are also listed as "alternative forms" so that they can be used in searching for such mistakes by the transcribers and thus can be corrected.

A sample of the lexical bank is given in Table 8 below.

Table 8. A sample of the lexical bank

Chinese characters	Minnan Pinyin	POS	Meaning/synonym (or alternative forms)
未記	be7ki3	VK	
未見笑	be7kian3siau3/ bue7kian3siau3	VH	不要臉
賣了	be7liau2/ bue7liau2	VB	賣完
賣了了	be7liau2liau2/ bue7liau2liau2	VB	賣光光
賣了了去	be7liau2liau2khi3/ bue7liau2liau2khi3	VB	賣光光去

[2] Programmers who helped out with the development of this program at different stages were Ming-Chung Chang and Charles Jie.

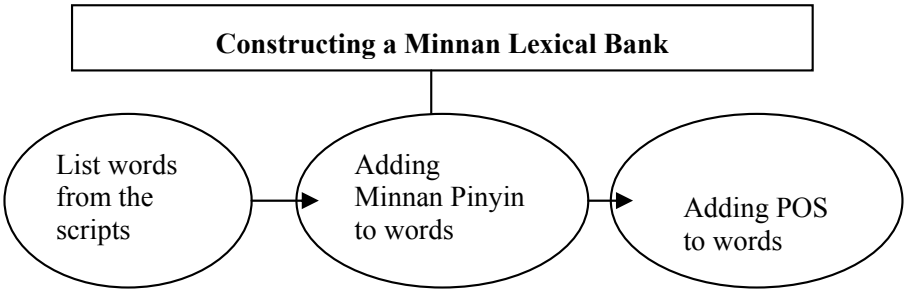


Figure 2. Constructing a Minnan lexical bank

After the lexical bank was established, an automatic word segmentation program was developed. This program also converts the word into Minnan Pinyin after segmentation. The way the program works is to identify a string of sounds that match the word in the column "Chinese characters" in the lexical bank. It then segments the word from the text and codes it in Minnan Pinyin. The word segmentation standard mostly follows that of the Sinica Corpus [Huang *et al*. 1997].

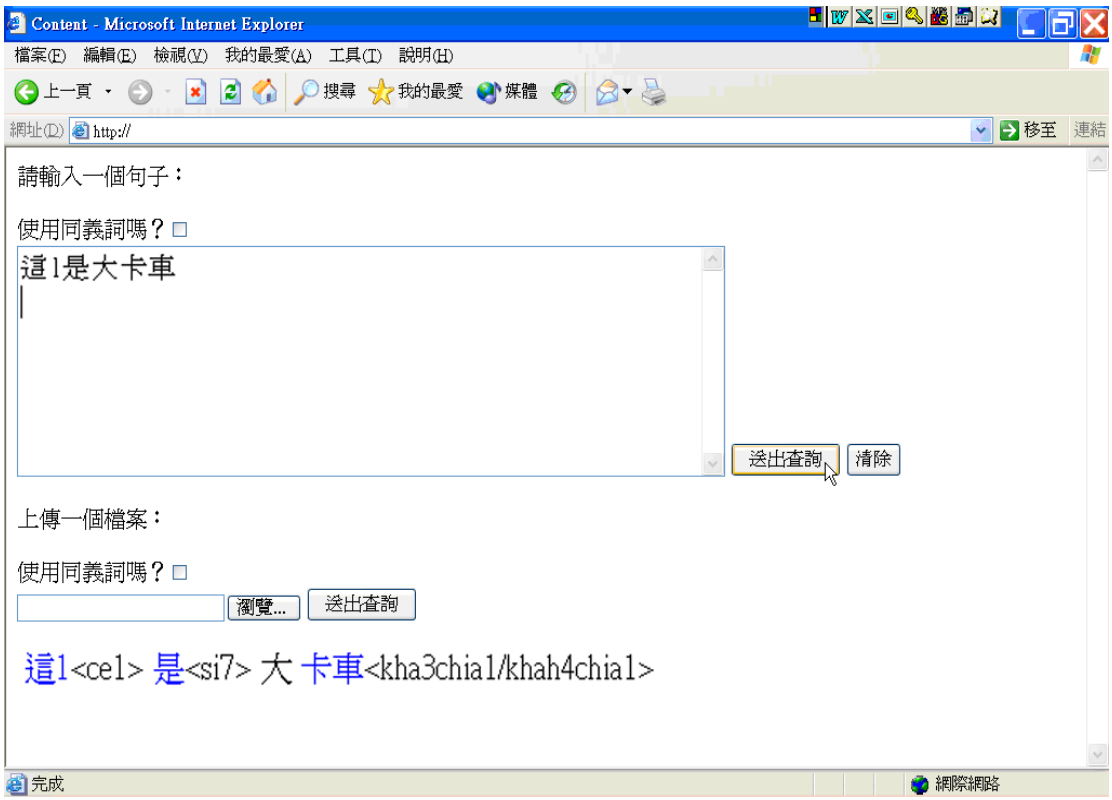


Figure 3. Automatic Word Segmentation

After word segmentation and Minnan Pinyin conversion at the %ort tier, POS codes are tagged to the word at the %cod tier.

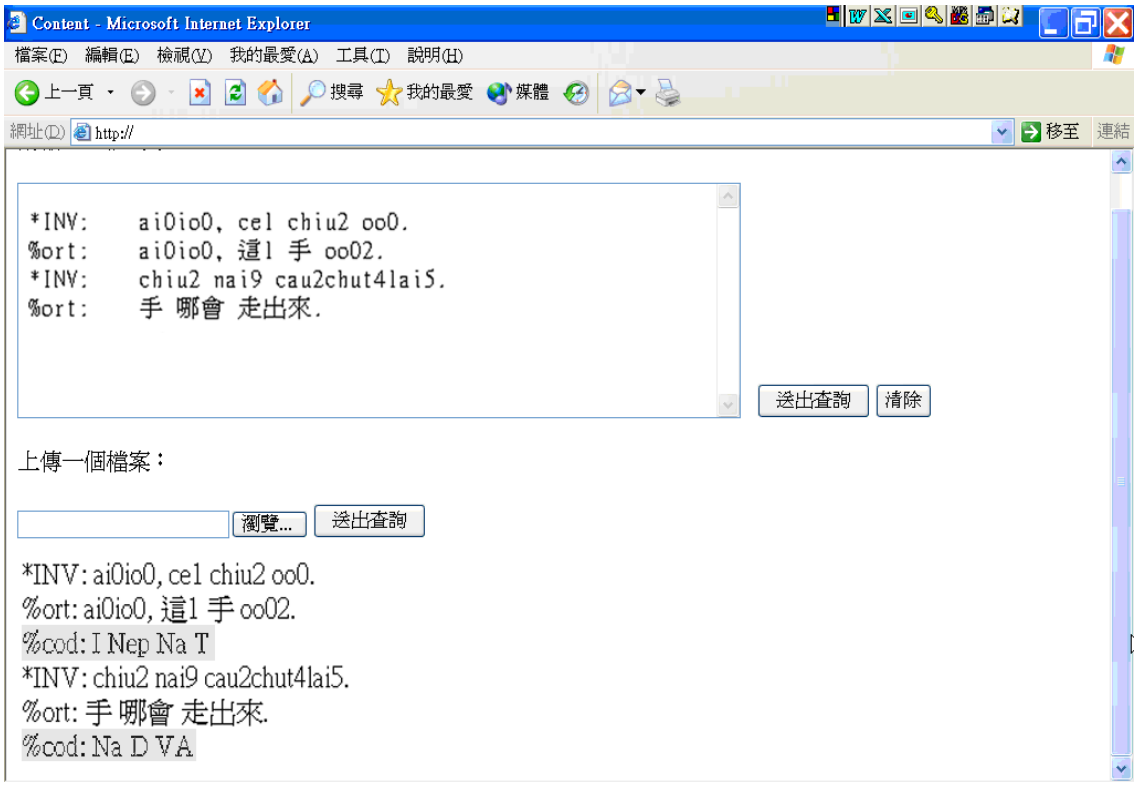


Figure 4. POS tagging after autosegmentation

During the process of constructing this corpus, we found some issues that only occur in speech-based corpora and not in corpora that are based on written texts. The first was the issue of multiple pronunciations of the same word due to dialectal variation. For example, the word for "want to" 欲 is pronounced /beh/ for some people, but is pronounced /bueh/ for others. Since they belong to the same word (same lexical entry), they have to be listed under the same lexical entry in the lexicon as /beh^bueh/. This does not happen in corpora based on written texts.

This phenomenon is especially common in Minnan because Taiwan Minnan speakers originally came from different areas in Fujian Province, China, including Zhangzhou 漳州, Quanzhou 泉州, and Xiamen 廈門. Therefore, dialect variations are very common. Words with multiple pronunciations (mostly dialectal variations) are all listed but connected by ^.

This is not a problem when the speech is transcribed into Chinese characters because there is only one orthographic form for each word. This problem is also not too serious when the speech is transcribed in a romanization notation, like Minnan Pinyin, manually by researchers. Transcribing speech into Minnan Pinyin is slow and an automatic converter is preferred. However, when an automatic word segmentation program uses the lexicon for word

segmentation, it will automatically retrieve a multiple pronunciation form like /beh^bueh/.

Take the following utterance as an example.

*CHI: gua2 beh4^bueh4 ak4chai3.
%ort: 我 欲 沃菜.

The word meaning "want" 欲 has two pronunciations *beh4* and *bueh4* and they show up as *beh4^bueh4* as in the main tier.

When counting words, they are counted as one word. That is, they are the same word in the lexicon and do not cause trouble in word frequency counts. However, when counting syllable token frequencies, they will be double counted. Besides, these two pronunciations have different syllable types, CVC and CVVC, respectively. Moreover, it is necessary to know the real target pronunciation of the specific speaker. Therefore, we need to have another tier %pro to show the actual pronunciation of the specific speaker based on the recording. Unfortunately, this can only be done manually.

*CHI: gua2 beh4^bueh4 ak4chai3.
%ort: 我 欲 沃菜.
%pro: gua2 beh4 ak4chai3.

3.2 The Inconsistency Issue and the Spell-Checker

Minnan speech recognition systems are still being developed. Hence, transcription can only be done manually. As mentioned above, Minnan does not have a conventionalized orthography, so transcribers might be inconsistent in choosing the written form. For example, /an3cuann2/ "how" can be transcribed as 怎樣, 怎麼樣, 按怎, 怎麼, 什麼, and so on. As shown by Minnan dictionaries, 按怎 is listed in the lexicon as the standard form in Minnan. Therefore, it is very important to design a program that can check for inconsistency in the written form.

A spell-checker for Minnan was thus developed.[3] This spell-checker works together with the automatic word segmentation program. When the program is segmenting the text, it searches for words in the columns of "Chinese character" and "alternative forms" in the lexical bank. It then segments the word and adds Minnan Pinyin to the word.

[3] This program was designed by the author and James Myers, and was implemented by Ming-Chung Chang.

The most challenging situation for the spell-checker is probably a case where the transcriber uses a form translated from Mandarin. That is, the form is not a standard Minnan written form. For example, the form 早上 "morning" is not a standard Minnan written form. However, the spell-checker finds that the form 早上 matches an alternative form (in the fourth column) in the lexical bank. In other words, it is very likely a Mandarin form being borrowed by the transcriber. The spell-checker then finds all the Minnan words that have listed 早上 as an alternative form. These are 早起 /ca2khi2^cai2khi2/, 早時 /ca2si5/, 兮早仔 /e1cai2a2^e7ca2a2/,兮早起/e1cai2khi2^e1ca2khi2/, and 透早/thau3ca2/, as shown in Table 9 below.

Table 9. Inconsistency in orthographic transcriptions

Chinese characters	Minnan Pinyin	POS	Synonym/ Alternative forms
早起	ca2khi2^cai2khi2	Nd	*早上*
早時	ca2si5	Nd	*早上*
兮早仔	e1cai2a2^e7ca2a2	Nd	e5早仔、今早、*早上*、下早仔
兮早起	e1cai2khi2^e1ca2khi2	Nd	e5早仔、今早、*早上*、下早仔
透早	thau3ca2	Nd	*早上*

The user can then decide which of the forms matches the pronunciation presented in Minnan Pinyin in the second column.

In summary, the automatic word segmentation program is able to do four things at the same time:

(1) segment words in the text

(2) code Minnan Pinyin for the words already transcribed in Chinese characters

(3) correct inconsistent written forms

(4) expand the lexical bank by adding new words

4. Some Findings from Research based on TAICORP

In this section, preliminary findings based on this corpus are reported. Since the syllable is a fundamental phonological unit, we will focus on findings on syllable distributions, including token and type frequencies.

As mentioned in the introduction, there are about two million syllables in TAICORP. The frequencies of syllables in words and syllables in particles are given in Table 10.

Table 10. Syllable Frequency Counts in TAICORP

	Syllables	
	syllables (in words) 1,558,408	syllables (particles) 538,992
Total	2,097,400	

One interesting finding about syllable distribution is that about 26% of the syllables are particles. There are 26 different syllables found in the corpus, as shown in Table 11 below. Note that, although it is possible to write these particles in Chinese characters, most of them still do not have conventionalized written forms. Also note that some syllables might represent more than one particle. In this case, a digit is added to distinguish among them in the text, *e.g.*, "a1 (啊 1)", "a2 (啊 2)", "a3 (啊 3)." The ones with very low frequencies could be considered idiosyncratic of the speakers.

Table 11. Particles and their token frequencies in TAICORP

Particle	Token frequencies
a 啊	240,103
oo 哦	118,450
la 啦	48,398
le 咧	41,137
hoonn	22,811
ne 呢	19,932
hoo	17,773
u	8,436
hann	7,588
m	6,760
ma 嘛	2,232
hannh	2,144
ue 喂	712
o	667
pa	508
io	466
lioo	440
noo	244
ng	204

Particle	Token frequencies
onn	150
loo	126
oonn	122
me	80
oi	24
na 哪	20
ni	10

These particles mainly serve pragmatic functions [Li 1999; Hung 2003; Hung *et al.* 2004]. Since particles do not have underlying tones, they play a more crucial role in prosody and pose more challenges for speech recognition. This is an area that deserves more attention.

As to syllables in words, the syllable token frequencies of adults and children are given below.

Table 12. Syllable Token Frequencies in TAICORP

	Adults	%	Rank	Children	%	Rank
CV	382760	33.2	1	140028	34.5	1
CVC	260358	22.6	2	79976	19.7	2
CVV	209672	18.2	3	79763	19.7	3
V	122111	10.6	4	47426	11.7	4
CVVC	71852	6.2	5	20092	5.0	5
Subtotal		90.8			90.6	
VC	28341	2.4	6	8438	2.1	6
VV	26126	2.3	7	8389	2.1	7
CN	21392	1.9	8	7661	1.9	8
N	12293	1.1	9	5812	1.4	9
CVVV	8723	0.8	10	3563	0.9	11
VVC	8655	0.8	11	4278	1.1	10
VVV	490	0.0	12	209	0.1	12
Total	1152773			405635		

Note that the top-five most frequent syllable types are the same for both adults and children. They are CV, CVC, CVV, V, and CVVC.

Regarding type frequencies, there are totally 624 different syllables found in both adults' and children's speech. However, different syllables might belong to the same syllable type. For example, the ten words listed in the following table are different syllables with the same

syllable type CV.

Table 13. Examples of CV syllables

example	IPA	Minnan Pinyin	Coding
抱 "hold"	p^ho	pho	CV
馬 "horse"	be	be	CV
霧 "fog"	bu	bu	CV
坐 "sit"	tse	ce	CV
飼 "feed"	$tɕ^h$i	chi	CV
好 "good"	ho	ho	CV
虎 "tiger"	hɔ	hoo	CV
雞 "chicken"	ke	ke	CV
去 "go"	k^hi	khi	CV
三 "three"	sã	sann	CV

In order to obtain syllable type frequencies, it is necessary to code the syllable types first. After coding the syllables, we found that there were a total of 12 different syllable types in Minnan. The syllable type frequencies are as follows.

Table 14. Syllable Type Frequencies in Minnan

Syllable Type	Total
CVC	218
CVV	131
CV	109
CVVC	83
VC	19
CVVV	17
VV	12
CN	12
V	11
VVC	8
VVV	2
N	2
Total	624

To summarize the findings:

(1) The most frequent syllable type is CV. This is consistent with theories in the phonology literature where CV has been considered the core syllable. This is also consistent with the findings in infant vocalization. In another words, this is a very unmarked pattern and might also be a cross-linguistic universal pattern.

(2) The second most frequent syllable type is CVC. This result is not surprising because in speech perception, a CVC syllable might be the easiest to perceive with the acoustic cues from formant transitions of the preceding as well as the following consonant of the nucleus vowel.

(3) Both adults and children have the same top five syllable types, *i.e*., CV > CVC > CVV > V > CVVC. Also note that, CV and CVC syllables count more than half of the total syllables. Even more strikingly, the five most frequent syllable types account for more than 90% of the syllables.

(4) Since the adults show the same patterns as the children, there is a possibility that the children were influenced by the adults (*i.e*., the input lexicon), although this needs to be confirmed by further research.

(5) Compared with data from Dutch children, there is a great similarity between in syllable types. Boersma and Levelt [2000] and Levelt, Schiller, and Levelt [1999] found that the order of acquisition in Dutch children was CV > CVC > V > VC. (These two languages differ in that Dutch does not allow VV syllables and that Minnan does not allow CC consonant clusters.）

(6) We have collapsed the sonorant coda with the obstruent coda (so-called Rusheng or checked syllables), *i.e*., collapsing CVN and CVK into CVC. Since the obstruent codas seem to behave differently [Tsay and Huang 1998], it might be interesting to have an alternative analysis. As Zamuner *et al*. [2005] point out, there seems to be a difference between syllables with sonorant coda and syllables with obstruent coda in English. Some cross-linguistic studies might be worth pursuing.

5. Concluding Remarks

We have introduced the construction of TAICORP, a speech-based corpus of Taiwan Minnan. We have also addressed some issues related to transcribing sound files into text files in Minnan, including multiple pronunciations and the orthographic problems. The automatization process using the corpus has also been illustrated.

This corpus has been used for studies on various aspects of child language acquisition, including tone acquisition [Tsay and Huang 1998; Tsay *et al*. 2000; Tsay 2001], consonant acquisition [Liu and Tsay 2000], vowel development [Lee 2007], classifier acquisition [Myers and Tsay 2000, 2002], final particle acquisition [Hung *et al*. 2004], verb acquisition [Lee and Tsay 2001; Huang 2005; Lin and Tsay, to apper], noun acquisition [Kuo *et al*. 2005],

vocabulary acquisition [Lin 2004; Tsay and Cheng, in preparation]. The corpus is being coded with more phonological annotations such as syllable boundary and tone groups for studying prosodic acquisition. A potential proposal on the WordNet of child language is also being explored. As this corpus is based on spontaneous speech, it also has applications for speech research, for example, analyzing phonetic characteristics of disfluency in child speech.

Acknowledgements

This research was supported by research grants from the National Science Council (NSC 92-2411-H-194-015, NSC 93-2411-H-194-025, NSC 94-2411-H194-002). The author would like to thank the research assistants who participated at different stages of this research, especially Ting-yu Huang, Hui-chuan Liu, Xiao-jun Chen, Peiyu Hsieh, and Yunwei Li. Comments and suggestions from the three anonymous reviewers are also highly appreciated.

References

Boersma, P., and C. Levelt, "Gradual Constraint-Ranking Learning Algorithm Predicts Acquisition Order," *The Proceedings of the Thirtieth Annual Child Language Research Forum*, 2000, pp, 229-237.

Chen, K.-J., C.-R. Huang, L.-P. Chang, and H.-L. Hsu, "SINICA CORPUS: Design Methodology for Balanced Corpora," *Language, Information, and Computation*, 11, 1996, pp. 167-176.

CKIP, "Chinese Part-of-Speech Analysis," Technical Report No. 93-05, Institute of Information Science Academia Sinica, Taipei, 1993.

CKIP, "Content and Instruction of the Sinica Balanced corpus (revised version)," Technical Report No. 95-02, Institute of Information Science Academia Sinica, Taipei, 1998.

Huang, C.-R., K.-J. Chen, F.-Y. Chen, and L.-L. Chang, "Segmentation Standard for Chinese Natural Language Processing," *Computational Linguistics and Chinese Language Processing*, 2 (2), 1997, pp. 47-62.

Huang, S., *Yuyan, Shehui yu Zuqun Yishi [Language, Society, and Ethnic Awareness]* Crane Publishing, Taipei, 1993. 黃宣範，語言、社會與族群意識 -- 台灣社會語言學的研究，文鶴出版公司，台北，1993。

Huang, Y.-C., The Child's Acquisition of Verbs in Taiwanese, MA thesis, National Chung Cheng University, Taiwan, 2005.

Hung, C. C.-F., The Child's Utterance Final Particles in Taiwanese: A Case Study, MA thesis, National Chung Cheng University, Taiwan, 2003.

Hung, J.-F., C. Li, and J. Tsay, "The Child's Utterance Final Particles in Taiwanese: A Case Study," In *Proceedings of the 9th International Symposium of Chinese Languages and Linguistics*, 2004, National Taiwan University, Taipei, Taiwan, pp. 477-498.

Kuo, J., J. Tsay, and J. Peng, "Basic Level Effects in Taiwanese Noun Acquisition." In *Proceedings of 2005 Conference on Taiwan Culture: Linguistics, Literature, Culture and Education,* 2005, Chia-yi: National Chiayi University. pp. 43-54.

Lee, T. H.-T., and J. Tsay, "Argument structure in the early speech of Cantonese-speaking and Taiwanese-speaking children," *The Joint Meeting of the 10th IACL and the 13th NACCL*, June 22-24, 2001, UC Irvine.

Lee, Y.-W., Vowel Development of a Child Acquiring Taiwan Southern Min, MA thesis, National Chung Cheng University, Taiwan, 2007.

Levelt, C. C., N. O. Schiller, and W. J. M. Levelt, "A developmental grammar for syllable structure in the production of child language," *Brain and Language*, 68, 1999, pp. 291-299.

Li, Y. C., *Utterance-final Particles in Taiwanese: a Discourse-pragmatic Analysis*, Crane Publishing, Taipei, 1999.

Lin, P.-C., The Acquisition of Nouns and Verbs in Taiwanese, MA thesis, National Chung Cheng University, Taiwan, 2004.

Lin, H.-L., and J. Tsay, "Acquiring Causatives in Taiwan Southern Min," *Journal of Child Language*, to appear.

Liu, J. H. C., and J. Tsay, "An Optimality-Theoretic Analysis of Taiwanese Consonant Acquisition," In *Proceedings of the 7th International Symposium of Chinese Languages and Linguistics*, 2000, National Chung Cheng University, Chiayi, Taiwan, 2000, pp. 107-126.

MacWhinney, B., *The CHILDES Project: Tools for Analyzing Talk*, 2nd ed. Lawrence Erlbaum Associates, NJ., 1995.

MacWhinney, B. and C. Snow, "The Child Language Data Exchange System," *Journal of Child Language*, 12, 1985, pp. 271-296.

Myers, J., and J. Tsay, "The Acquisition of the Default Classifier in Taiwanese," In *Proceedings of the 7th International Symposium of Chinese Languages and Linguistics*, 2000, National Chung Cheng University, Chiayi, Taiwan, pp. 87-106.

Myers, J. and J. Tsay. "Grammar and Cognition in Sinitic Noun Classifier Systems," In *Proceedings of the First Cognitive Linguistic Conference*, National Chengchi University, Taipei, 2002, pp. 199-216.

Tsay, J., "Phonetic Parameters of Tone Acquisition in Taiwanese," *Issues in East Asian Language Acquisition*, ed. by Minehru Nakayama, Kuroshio Publishers, Tokyo, 2001, pp. 205-226.

Tsay, J., "Taiwan Child Language Corpus: Data Collection and Annotation," In *Proceedings of 5th Workshop on Asia Language Resources*, Jeju Island, Republic of Korea, 2005a, pp. 56-61.

Tsay, J., "The Language Issue ," Documentation of Chaiyi City, the Language and Literature Volume, Vol. 8, Chiayi City Hall, Chiayi, Taiwan, 2005b, pp. 1-66. (蔡素娟撰，嘉義市政府編印，嘉義市志語言文學志語言篇，嘉義，2005b，pp. 1-66)

Tsay, J. and C. C. Cheng, "Productivity in Young Children's Vocabulary," in preparation. Manuscript, National Chung Cheng University, Taiwan.

Tsay, J. and T.-Y. Huang, "Phonetic Parameters in the Acquisition of Entering Tones in Taiwanese," In *The Proceedings of the Conference on Phonetics of the Languages in China,* City University of Hong Kong, China, 1998, pp. 109-112.

Tsay, J., J. Myers, and X.-J. Chen, "Tone Sandhi as Evidence for Segmentation in Taiwanese," In *Proceedings of the 30th Child Language Research Forum,* Center for the Study of Language and Information, Stanford, California, 2000, pp. 211-218.

Zamuner, T. S., L. Gerken, and M. Hammond, "The Acquisition of Phonology Based on Input: A closer look at the relation of cross-linguistic and child language data," *Lingua*, 115(10), 2005, pp. 1403-1426.

Minnan Dictionaries

Chen, X., Taiwanhua Dacidian [Taiwanese Dictionary], Yuanliu Publishing, Taipei, 1998. 陳修，台灣話大辭典：閩南話漳泉二腔系部分，遠流出版事業股份有限公司，台北，1998。

Dong, Z., Taiwan Minnanyu Cidian [Taiwan Southern Min Dictionary], Wunan Publisher, Taipei, 2001. 董忠司編纂，國立編譯館主編，台灣閩南語辭典，五南圖書出版有限公司，台北市，2001。

Li, R., Xiamen Fangyan Cidian [Xiamen Dialect Dictionary], Education Publisher, Jiangsu, 1998. 李榮，廈門方言詞典，江蘇教育出版社，江蘇省，1998。

Wu, S., Guotaiyu Duizhao Huoyong Cidian [Mandarin-Taiwanese Comparative Dictionary], Yuanliu Publising, Taipei, 2000. 吳守禮主編，國台語對照活用辭典—詞性分析、詳註廈漳泉音（上冊, 下冊），遠流出版有限公司，台北，2000。

Xu, J., Changyong Hanzi Taiyu Cidian [Taiwanese Dictionary of Frequently Used Chinese Characters]. Culture Department, Zili Evening News, Taipei, 1992.許極燉編著，常用漢字台語辭典，自立晚報社文化出版部，台北市，1992。

Yang, Q., Guotai Shuangyu Cidian [Mandarin-Taiwanese Bilingual Dictionary], Duli Publishing, Kaohsiung, 1993. 楊青矗主編，國台雙語辭典，敦理出版社，高雄，1993。

Yang, X., Minnanyu Cihui [Southern Min Vocabulary], Ministry of Education, Taipei, 2001. 楊秀芳撰稿，教育部國語推行委員會編輯，閩南語字彙（一, 二）修訂版，教育部，台北市，2001。

Computational Linguistics and Chinese Language Processing
Vol. 12, No. 4, December 2007, pp. 443-458

Automatic Pronunciation Assessment for Mandarin Chinese: Approaches and System Overview

Jiang-Chun Chen*, Jyh-Shing Roger Jang*, and Te-Lu Tsai+

Abstract

This paper presents the algorithms used in a prototypical software system for automatic pronunciation assessment of Mandarin Chinese. The system uses forced alignment of HMM (Hidden Markov Models) to identify each syllable and the corresponding log probability for phoneme assessment, through a ranking-based confidence measure. The pitch vector of each syllable is then sent to a GMM (Gaussian Mixture Model) for tone recognition and assessment. We also compute the similarity of scores for intensity and rhythm between the target and test utterances. All four scores for phoneme, tone, intensity, and rhythm are parametric functions with certain free parameters. The overall scoring function was then formulated as a linear combination of these four scoring functions of phoneme, tone, intensity, and rhythm. Since there are both linear and nonlinear parameters involved in the overall scoring function, we employ the downhill Simplex search to fine-tune these parameters in order to approximate the scoring results obtained from a human expert. The experimental results demonstrate that the system can give consistent scores that are close to those of a human's subjective evaluation.

Keywords: CAPT, CALL, Speech Recognition, Tone Recognition, Speech Assessment, GMM, Mandarin Chinese, Downhill Simplex Method, Phoneme, Intensity, Rhythm, Forced Alignment.

1. Introduction

With the fast-growing power of personal computers and the advances in speech and language processing technologies, software systems for CALL (Computer Assisted Language Learning) now allow a person to learn a language by interacting solely with computers, especially for second language (L2) learning. In general, a CALL system involves testing procedures for both the receptive and productive skills of a given subject. To evaluate receptive skills such as

* Department of Computer Science, National Tsing Hua University, Taiwan
E-mail: {jtchen; jang}@cs.nthu.edu.tw
+ Innovative DigiTech-Enabled Applications & Services Institute, Institute for Information Industry

[Received December 7, 2006; Revised October 16, 2007; Accepted October 17, 2007]

reading and listening, the procedure is relative simple, since the evaluation is usually based on exams containing questions of single or multiple choices. On the other hand, to evaluate the productive skills of speaking or writing, the procedure is relatively difficult and time-consuming, since a human expert is usually required to evaluate the speech or writing in a subjective and time-consuming manner. With advances in automatic speech recognition, a Computer-Assisted Pronunciation Training (CAPT) system can evaluate the pronunciation quality using various speech features, and provides high-level feedback (hints for further improvement, *etc.*) to the user. Successful applications of CAPT have been reported in the literature [Neumeyer *et al.* 2000; Kim and Sung 2002; Neri *et al.* 2003].

In this paper, we propose several algorithms for constructing a CAPT system that can assess a test utterance in Mandarin Chinese with respect to a target utterance by a native speaker. Basically, there are four evaluation criteria based on different acoustic features, as explained in the following.

1. **Phoneme:** This is based on the log probabilities of the test utterance with respect to the acoustic models derived from a large speech corpus for speaker-independent speech recognition. Note that the target utterance is not required for this evaluation.
2. **Tone:** Each syllable is associated with a tone in Mandarin Chinese. The pronounced tone of a syllable can be identified by a tone classifier, and the result is then compared with the correct tone for evaluation. Note that we can obtain the correct tones from the text of the utterance; hence, the target utterance is not used directly for this evaluation.
3. **Intensity:** Each syllable has an intensity vector, which is compared to that of the corresponding syllable in the target utterance to ensure it has a similar score.
4. **Rhythm:** The duration of each syllable and the silence in between are compared to those of the target utterance to ensure they have a similar score.

Each of the scoring functions for the above four criteria involves several nonlinear parameters. These four scoring functions are then linearly combined to give a score between 0 and 100. We then employ a search method that can find optimum values of these parameters such that the computed scores can approximate those of a human expert. The experimental results demonstrate the feasibility of the proposed approach, which can give consistent results when compared with human evaluation.

The rest of this paper is organized as follows. Section 2 gives a quick overview of related work on automatic pronunciation assessment. Section 3 explains the speech-related techniques used in our approach, including Viterbi decoding and tone recognition. The method of combining weighted scores based on derivative-free optimization is also explained. Section 4

describes the GUI of our system. Section 5 demonstrates the experimental results and Section 6 gives concluding remarks.

2. Related Work

In general, a CAPT system can evaluate pronunciation quality using various speech features. Moreover, the system is expected to have optimum performance by minimizing the discrepancies between the scores from computers and those from a human expert. However, most of the reported systems do not take the characteristics of tonal languages into consideration. In particular, Mandarin Chinese is a tonal language, and each character is associated with one out of five possible tones. The tone of a given character is also context-dependent according to tone sandhi [Lee 1997]. Hence, the correct pronunciation of the tone of each character in a sentence is the most challenging problem for a Mandarin-learning non-native speaker. The proposed system takes this specific problem into consideration and tries to create a comprehensive Mandarin Chinese pronunciation assessment system.

3. The Proposed Approach

The proposed pronunciation assessment system uses four criteria to evaluate a test utterance with respect to a target utterance. The algorithms of these four criteria are explained in this section.

3.1 Syllable/Phone Segmentation using HMM-based Forced Alignment

HMM (Hidden Markov Models) has been used for speech recognition with satisfactory performance over the past few decades [Rabiner and Juang 1993; Huang *et al*. 2001]. Our system employs a speaker-independent HMM-based recognition engine, which was trained on a balanced corpus of Mandarin Chinese recorded by 70 subjects in Taiwan. Each speech feature vector contains 39 dimensions, including 12 MFCC (Mel-Frequency Cepstral Coefficients) and 1 log energy, along with their delta and double delta values. 174 right-context-dependent (RCD) biphone models are derived from the speech corpus. In other words, two phones are regarded as different models if their right phones are different. For example, the right phone of “a” in the syllable “dan” is “n”, while the right phone of “a” in the syllable ‘jiang’ is “ng”. As a result, the phone “a” in the syllable ‘dan’ is defined as “a+n” in the RCD context, to distinguish it from, say, “a+ng” in the syllable “jiang”. The number of RCD biphone models is much larger than that of the context-independent monophone models, thus a large corpus is required for the reliable training of RCD biphone models. Furthermore, we have also designed an efficient pruning method for our speech recognizer [Jang and Lin 2002].

For pronunciation assessment, we need to build a lexicon net consisting of the models of the uttered text. Then, Viterbi decoding is used to do forced alignment between the speech signals and the models in the lexicon net. The final results include frame indices of isolated syllables/phones and the corresponding log probability. The log probability is an absolute measure of how closely the utterance matches the acoustic models identified from the speech corpus. Consequently, the log probability varies considerably among different models due to their different phonetic characteristics, and thus cannot be used directly for phoneme assessment. Instead, we use a ranking-based confidence measure to be explained later.

To deal with characters having multiple pronunciations in Mandarin, we use a sausage-like lexicon net. Take the sentence "朝辭白帝彩雲間" in Tang Poetry for example, where the third character can be pronounced as "bai" or "bo". Both pronunciations are commonly used. Therefore, our lexicon net has two branches for this character to accommodate both pronunciations, as shown in Figure 1 where Hanyu Pinyin is used for phonetic transcription.

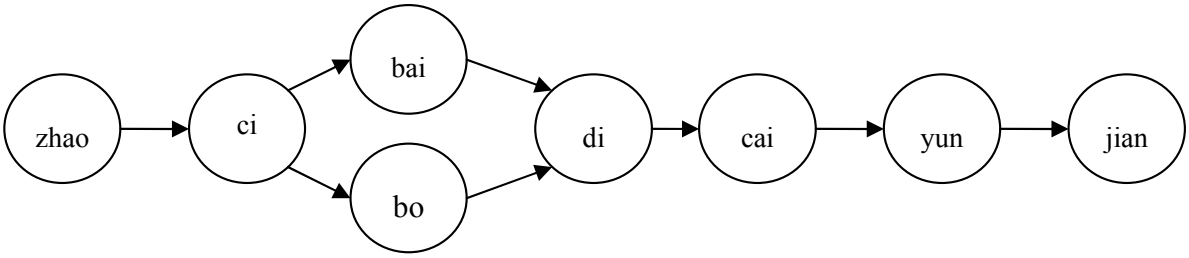


Figure 1. The lexicon net for "朝辭白帝彩雲間".

Figure 2 shows a typical result of forced alignment for the sentence "但使龍城飛將在" in Tang Poetry. The solid lines in the waveform plot indicate the boundaries of phones. The score for each syllable is labeled under each Chinese character, while the score for each phone is labeled under the phone name. The scores depend on the quality of the pronunciation as well as the correctness of forced alignment. In particular, we can see that the fricative phone "sh" in the second character is correctly segmented with a score of 100, while the phone "ch" in the forth character is badly segmented (due to its mispronunciation as a non-retroflexed consonant) with a score of 29. The details of phoneme assessment will be described in the next section.

Figure 2 also shows the pitch vector of this utterance. The dotted lines represent the pitches of the voiced parts, while the union of the dotted and the solid lines represents the pitch curve of the whole utterance derived by dynamic programming. Details of the pitch tracking method can be found in Chen and Jang [2007].

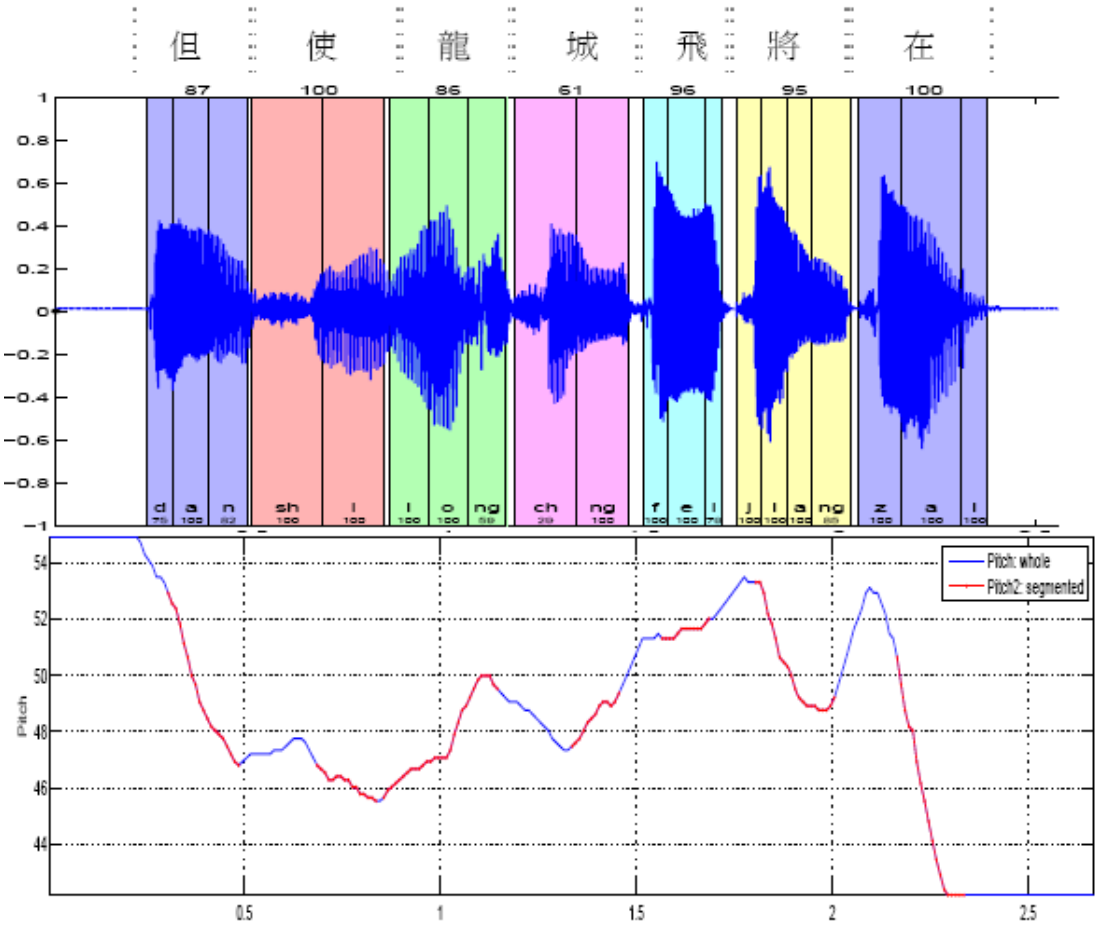


Figure 2. An example of forced alignment for the test utterance "但使龍城飛將在". The upper panel shows the phone segmentation results and the corresponding phoneme scores. The bottom panel plots the pitch vector where the dotted lines correspond to the voiced parts in the utterance.

3.2 Ranking-Based Confidence Measure for Phoneme Assessment

The log probability represents an absolute measure of how closely a pronunciation approximates a given phone model, which does not take into consideration the effect of other competing models. As a result, the log probability varies considerably among different phone models due to their different phonetic characteristics. To deal with this problem, we used a relative measure based on the ranking among all competing biphone models. This is an improved version of our previous approach to confidence measure, based on the ranking among 411 syllables in Mandarin [Chen *et al.* 2004]. By using phone-based ranking, our

system is able to track down phone-level pronunciation errors for detailed and better assessment.

The phone-based phoneme assessment proceeds as follows.

1. For a given biphone model of "x+y", we define the set of competing models as "*+y" where * is a wildcard representing all the possible phones that form a legal biphone with y.
2. After forced alignment, we can obtain the speech signals corresponding to the biphone "x+y". We then send the speech signals to the competing models for a log probability evaluation and find the rank (zero-based) of "x+y" in the competing models.
3. Since each biphone has a different set of competing models, we divide the rank of "x+y" by the size of its competing models to obtain a rank ratio between 0 and 1. Once the rank ratio is obtained, the phoneme score of the *i*-th phone in an utterance is then determined by the following formula:

$$s_{\text{phoneme},i} = \frac{100}{1+\left|\frac{rr_i}{a}\right|^b}, \quad (1)$$

where rr_i is the rank ratio of the i th biphone, and a and b are the tunable parameters of this scoring function. In particular, when rr_i is equal to 0, a perfect score of 100 is obtained. On the other hand, if rr_i is larger, the score is lower. The values of parameters a and b are empirically set to 0.1 and 2 respectively. A typical plot of function $s_{\text{phoneme},i}$ is shown in Figure 3. (An on-going research focus is to make the parameters a and b model-dependent and to determine their values automatically, which will be covered in our future publication.)

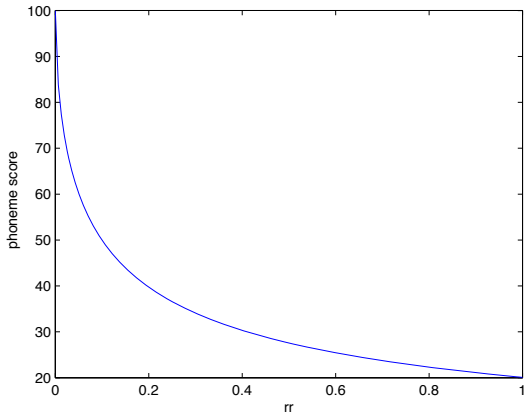


Figure 3. A typical plot of s_{phoneme} with respect to rr_i, where a and b are set to 0.1 and 2 respectively (For simplicity, we have removed the subscript i from the x-axis label of the plot).

The overall score of a test utterance with m phones can be expressed as a duration-weighted average of all of the phones' scores, as follows:

$$score_{\text{phoneme}} = \sum_{i=1}^{m} \frac{dur_i^p}{\sum_{j=1}^{m} dur_j^p} s_{\text{phoneme},i} \,, \tag{2}$$

where dur_i is the duration of the i th phone in test utterance and the weighting factor $\frac{dur_i^p}{\sum_{j=1}^{m} dur_j^p}$ is parameterized by p. In other words, in the test utterance, a phone with a longer duration will have a larger weighting factor. (For a clearer notation definition, we shall use p_{phoneme} to denote the free parameter p in the subsequent discussion.) Similar usage of log-probability is commonly adopted in the research of utterance verification [Sukkar and Lee 1996].

3.3 Tone Recognition Using GMM

Mandarin Chinese is a tonal language and each character is associated with a syllable (out of 411 possible syllables) and a tone (out of 5 possible tones). The tone/pitch information plays an important part in assessing a given utterance. As a result, we need to use isolated syllables for tone recognition.

Previous work of tone recognition for Mandarin Chinese is briefly described. [Chen and Wang 1993] classifies tones based on neural networks. More recently, Lin and Lee [2003] introduces a new set of inter-syllabic features to identify tones. To determine the tones of a given utterance, we use the pitch vector of each syllable segmented via the aforementioned forced alignment. Each pitch vector of a syllable is identified via the autocorrelation method [Huang *et al*. 2001], and then expanded into the Chebyshev polynomials, as explained in Li [2002].

In our experiment, a pitch vector is normalized to the interval [-1, 1] at the time axis and then approximated by the Chebyshev polynomials of degree 6. Moreover, before using the Chebyshev approximation, we have subtracted the mean from the pitch vector of a syllable. In other words, each pitch vector has a mean value of zero before the Chebyshev approximation is applied. As a result, we do not need to use $coef_0$ of the Chebyshev polynomials for tone recognition. Hence, the dimension of the feature vector for the tone recognizer is 5. In fact, more features can be employed for tone recognition, such as the volume of the syllable, the slope of the pitch vector, the duration of a syllable *etc*., as suggested in Huang [2006].

The polynomial coefficients are then used as the feature vectors for the GMM (Gaussian Mixture Model) tone classifier. We used the Tang Poetry microphone corpus [Tang Poetry 2002] of 3211 utterances recorded in our lab for the experiment. The corpus was recorded by eight males and two females, with a sample rate of 16 kHz and a bit resolution of 16 bits. To train the GMM, 2500 utterances (15144 syllables) are used as the training data and the other 711 utterances (4422 syllables) are used as the test set. All the utterances were segmented into syllables via forced alignment. We discarded syllables with durations of less than four frames or more than sixty frames, since the alignment might have been performed incorrectly on these syllables. The result demonstrates that when the number of Gaussian density functions for each tone is 128, we can obtain a recognition rate of 80.25%. For tone recognition, we only consider the most common four tones, the high flat (tone 1), the low rising (tone 2), the high low rising (tone 3), and the high falling (tone 4). Unlike these four lexical tones, the neutral tone does not have a specific pitch pattern; therefore, it is easily confused with the other four tones. As syllables with the neutral tone usually are shorter in duration and lower in energy, short-time energy is found useful in addition to the apparent key features derived from pitch-frequency contours [Lee 1997]. Considering the similarity in the lower energy between tone 1 and tone 5, we put tone 5 in the same class as tone 1 in our experiment. Table 1 lists the confusion matrix of the tone recognition result, where we can observe that tone 3 is mostly likely to be misclassified as tone 4. This is because the rising end of tone 3 is usually missing when the speech rate is high, causing tone 3 to be confused with tone 4.

Table 1. Confusion matrix of the test set for tone recognition

Recognized / Answer	Tone 1	Tone 2	Tone 3	Tone 4	Recognition Rate
Tone 1	1079	87	14	61	86.95%
Tone 2	105	961	89	54	79.49%
Tone 3	39	108	334	163	51.86%
Tone 4	65	27	32	1055	89.48%

Once the pitch vector of syllable *i* is classified into one of the four possible tones, the rank of the correct tone is converted into a score by the following equation:

$$s_{tone,i} = \frac{1-\frac{r_i}{3}}{1+k\cdot\frac{r_i}{3}}\cdot 100, \qquad (3)$$

where k is a tunable parameter and r_i is the rank of the desired tone for syllable *i*. When r_i=0 (where the desired tone appears at the top of the output ranking list), we have a perfect

score of 100. On the other hand, if r_i=3 (where the desired tone appears at the bottom), we have a score of 0.

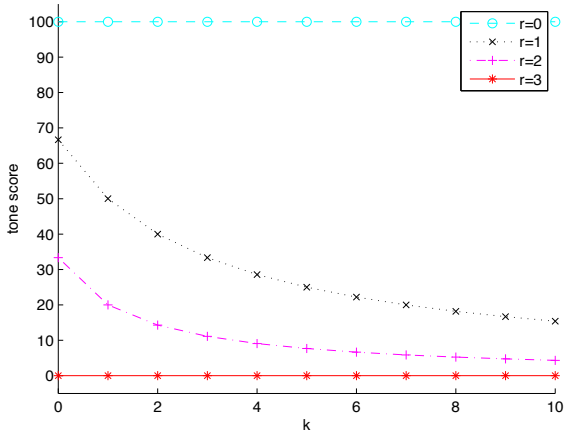


Figure 4. Tone score curves for r_i=0, r_i=1, r_i=2 and r_i=3, respectively, with the value of k varying from 0 to 10. (We have removed the subscript i from the x-axis label for simplicity.)

Figure 4 shows the tone score curves for r_i=0, r_i=1, r_i=2 and r_i=3, respectively, with the value of *k* varying from 0 to 10. The overall score of an utterance with c syllables is once again computed as the average of each syllable's score weighted by its duration:

$$score_{\text{tone}} = \sum_{i=1}^{c} \frac{dur_i^p}{\sum_{j=1}^{c} dur_j^p} s_{\text{tone},i}\,, \tag{4}$$

where dur_i is the duration of the i th syllable in the test utterance and the weighting factor $\frac{dur_i^p}{\sum_{j=1}^{c} dur_j^p}$ is parameterized by *p* (For a clearer notation definition, we shall use k_{tone} and p_{tone} to denote the free parameters *k* and *p*, respectively, in the subsequent sections).

3.4 Intensity

Intensity and rhythm are two other important factors in forming the prosody of an utterance. We shall describe how to determine the score of the intensity curves for measuring similarity in this subsection. The treatment of rhythm is covered in the next subsection.

Intensity is also referred to as the magnitude or the volume of a given utterance, which is an important cue for pronunciation and its assessment [Chen *et al*. 2004]. Since the intensity

of a given text can only be found in the target utterance, the similarity score is defined between the intensity curves of the test and the target utterances. In comparison, the phoneme score is obtained from the acoustic model and the tone score is obtained from the tone classification; both scores do not require the use of a target utterance.

In order to account for the variation in microphone gain, we need to normalize the signal amplitude of the test utterance before computing the intensity score. This is achieved by the least-squares estimate [chapter 5 of Jang *et al.* 1997] to find the best scaling factor on the test utterance, such that the squared error between the test and the target utterances is minimized. More formally, we define $\mathbf{r} = [r_1, r_2, ..., r_N]$ and $\mathbf{t} = [t_1, t_2, ..., t_N]$ as two intensity vectors of the reference (or target) and test utterances, respectively, after length normalization via interpolation. Then, the best scaling factor is computed, such that the error between the intensity vector of the reference utterance and the scaled version of the test utterance is minimized. The error can be expressed as $\|e\|^2 = \|\mathbf{r} - \mathbf{t}\theta\|^2$, which is minimized when $\theta = \hat{\theta} = \left(\mathbf{t}^T\mathbf{t}\right)^{-1}\mathbf{t}^T\mathbf{r}$. We then use the optimum value $\hat{\theta}$ to compute the scaled intensity vector of syllable i in the test utterance. Specifically, the dissimilarity (distance) of syllable i is $\|e_i\|^2 = \|\mathbf{r}_i - \mathbf{t}_i\hat{\theta}\|^2$, where $\mathbf{r}_i$ is the intensity vector of syllable i in the reference utterance, and $\mathbf{t}_i$ is the intensity vector (after interpolation to have the same length as $\mathbf{r}_i$) of syllable i in the test utterance. The intensity score of syllable i is then computed via the following equation:

$$s_{\text{intersity,i}} = \frac{100}{1 + k\|e_i\|^2}, \tag{5}$$

where k is a tunable parameter. When $\|e_i\|^2 = 0$, we have a perfect score of 100. On the other hand, if $\|e_i\|^2$ is large, the score will be small. Then, the overall intensity score of an utterance with c characters is computed as a weighted average:

$$score_{\text{intensity}} = \sum_{i=1}^{c} \frac{dur_i^p}{\sum_{j=1}^{c} dur_j^p} s_{\text{intersity},i}, \tag{6}$$

where dur_i is the duration of syllable i in the test utterance and the weighting factor $\frac{dur_i^p}{\sum_{j=1}^{c} dur_j^p}$ is parameterized by p. (For a clearer notation definition, we shall use $k_{\text{intensity}}$ and $p_{\text{intensity}}$ to denote the free parameters k and p, respectively, in the subsequent sections.)

3.5 Rhythm

We can define the rhythm as the duration vector (obtained from the forced alignment) of all syllables, including the short pause between any two syllables. For an utterance with c syllables, we can obtain a duration vector of size 2c-1, including the durations of c syllables and c-1 short-pauses. We define $\mathbf{p} = [p_1, p_2, ..., p_{2c-1}]$ and $\mathbf{q} = [q_1, q_2, ..., q_{2c-1}]$ as two duration vectors for the reference (target) and test utterances, respectively. The distance between these two duration vectors can be defined as the normalized sum of the absolute difference:

$$dist(\mathbf{p}, \mathbf{q}) = \frac{1}{2c-1} \sum_{i=1}^{2c-1} \left| p_i - q_i \right| / \, p_i \ . \tag{7}$$

The score for measuring rhythm is computed by the following equation:

$$score_{\text{rhythm}} = \frac{100}{1 + k \cdot dist(\mathbf{p}, \mathbf{q})} \tag{8}$$

where k is a tunable parameter. We denote the free parameter k as k_{rhythm} for clarity in later discussions.

A summary of the proposed four evaluation criteria and their corresponding speech/acoustic features, target of comparisons, and dissimilarity/distance measure is shown in Table 2.

Table 2. Summary of evaluation criteria and their corresponding speech/acoustic features, target of comparisons, and dissimilarity/distance measure.

Criteria	Speech/Acoustic Features	Target of Comparisons	Dissimilarity/ Distance Measurement
Phoneme	MFCC	Acoustic models	Ranking of the desired phoneme
Tone	Pitch	GMM-based tone classifier	Ranking of the desired tone
Intensity	Energy	Intensity vector of the reference utterance	Euclidean distance of scaled normalized intensity vectors
Rhythm	Duration	Durations of each syllable and short-pause of the reference utterance	Normalized absolute sum of difference in duration

3.6 Parametric Scoring Function

As mentioned in the previous subsections, we have obtained four scores based on phoneme, tone, intensity and rhythm. The overall scoring function is defined as the weighted average of four scores:

$$score = w_1 \cdot score_{\text{phoneme}} + w_2 \cdot score_{\text{tone}} + w_3 \cdot score_{\text{intensity}} + w_4 \cdot score_{\text{rhythm}}, \quad (9)$$

where $w_1 + w_2 + w_3 + w_4 = 1$. Apparently, the overall scoring function is formed within several parameters, including p_{phoneme}, k_{tone}, p_{tone}, $k_{\text{intensity}}$, $p_{\text{intensity}}$ and k_{rhythm}, and w_1, w_2, w_3, w_4. To fine-tune these parameters to approximate the scores by the human expert, we employ the downhill Simplex method [chapter 7 of Jang *et al*. 1997] to find the optimal values of these parameters. Basically, the downhill Simplex method is a derivative-free optimization method, which is less efficient than some methods, but simple and flexible in implementation. The flowchart of each of the evaluation processes is shown in Figure 5. The experimental results are covered in the next section.

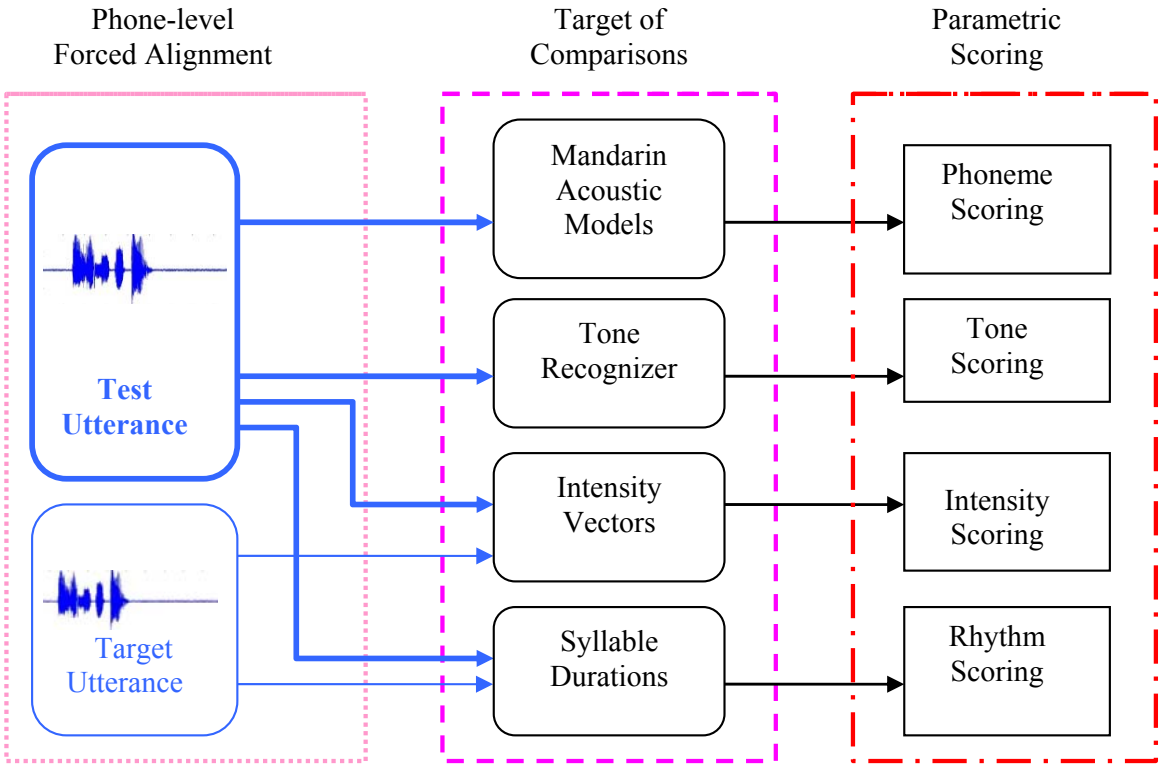


Figure 5. The flowchart of the proposed approach.

4. Experimental Results

To construct the scoring function, we used a dataset containing 400 utterances from 20 speakers, including 10 males and 10 females, each with various levels of proficiency in Mandarin Chinese. Each speaker was asked to utter 20 sentences chosen from the 300 most famous poems of the Tang dynasty. Some of the utterances are purposely pronounced

incorrectly in either content or tone to give "low-score" examples of the training data. These utterances were evaluated by a human expert who subjectively gave a score between 0 and 100 to each utterance, according to the "correctness". We then used the downhill Simplex method to fine tune the parameters of w_1, w_2, w_3, w_4, p_{phoneme}, k_{tone}, p_{tone}, $k_{\text{intensity}}$, $p_{\text{intensity}}$ and k_{rhythm}. The resulting value of w_1 for *phoneme* was 0.52, w_2 for *tone* was 0.22 and w_3 for *intensity* was 0.1 and w_4 for *rhythm* was 0.16, indicating that phoneme and tone were more important factors for speech assessment. This is also consistent with the observation that an utterance with wrong phonemes or wrong tones is much more easily recognized than an utterance with wrong intensity or rhythm.

Table 3. Confusion matrix in terms of three categories.

Machine / Human	Unit: Number of sentences		
	Good	Medium	Bad
Good	121	6	5
Medium	44	67	7
Bad	28	10	112

To evaluate the performance of the system, another set of 400 utterances recorded from 10 subjects was used for an outside test. Each utterance was assigned a category out of three candidates: good (above 80), medium (between 60 and 80), and bad (below 60). Table 3 lists the test results in the form of a confusion matrix in which each column corresponds to a category assigned by our system, and each row corresponds to a category assigned by the human expert. The median category has a smaller score range of [60, 80], therefore the data count is also lower.

In the above table, it is obvious that our system can match the categories assigned by a human expert in a satisfactory manner. The overall recognition rate in terms of these three categories is (121+67+112)/400 = 75%. In addition, the average of the absolute difference between scores from the computer and the human is 5.42, where the standard deviation is 2.31. In a related work, Kim and Sung [2002] reported a recognition rate of about 60% for intonation assessment in English learning, while Neumeyer *et al*. [2000] reported a machine-to-human correlation of 70% for American speakers learning French. Note that the result by Neumeyer is evaluated in the speaker level in their posterior scoring approach, which is different from our evaluation method.

Our goal in this study is to identify effective features/parameters that can approximate the scoring of a single human expert. Actually, having two or more human experts can definitely help shape our system in a more reliable manner. One way to take advantage of multiple human experts is to create a scoring system for each of them, and then combine the results by voting or weighted average. This will be an important direction of our future work.

5. Overview of the Software System Using the Proposed Approaches

Our software system, primarily for Japanese students, provides three different approaches to the learning of Mandarin Chinese, including pronunciation practice, interactive dialogue, and video-aided real-world dialogue. Figure 6 is a screen shot of our system when the pronunciation practice is in action. First, the student can choose a sentence, then the system will show the reference utterance at the bottom. The student can listen to the reference utterance before recording. After finishing the recording, the score of the test utterance is shown at the right-upper corner. The user can click the button labeled "點數結果" to check detailed scores to the phone level, as shown in the popup menu in Figure 6.

Figure 6. The screen shot of the pronunciation practice in action.

To make a vivid visual feedback, low scores for phones/syllables are displayed in red, as shown in the popup dialog, where "這" and "剛" have scores in red. In fact, "這" and "剛" were pronounced as "那" and "不", respectively, in the test utterance. The system was able to detect the mispronunciation and gave both syllables poor scores of 27 and 52, respectively. Note that the target sentence is displayed with Chinese characters, their Hanyu Pinyin transcription, and the corresponding tone symbols, including "－" (tone 1), "ˊ" (tone 2), "ˇ" (tone 3) and "ˋ" (tone 4). Hanyu Pinyin without any tone symbol represents "tone 5".

Figure 7 illustrates the screen shot of video-aided real-world dialogue. The scenario in this example is a conversation carried in a hotel checkout procedure. The video stops whenever the student needs to say a sentence in response to the counter clerk. If the scores of the student's utterance are higher than 80, then the video continues. Otherwise the student needs to practice the sentence until the score is higher than 80. The procedure is almost the

same as that of interactive dialogue, except that a video is played to imitate real-world conversations.

Figure 7. A screen shot of the video-aided real-world dialogue.

6. Conclusions

In this paper, we have developed the algorithms to construct a CAPT system for evaluating the pronunciation of Mandarin Chinese. The proposed system uses several techniques from speech signal processing and recognition, including the HMM-based forced alignment, a pitch determination using autocorrelation, and tone recognition using GMM. By using downhill Simplex search, we successfully derived a set of parameters for a scoring function that can approximate the scores from a human expert. Similar approaches can be applied to construct CAPT systems for other tonal languages, such as Taiwanese, Minnan, Cantonese, Tibetan, Punjabi, and so on.

References

Chen J.C. and J.S. R. Jang, "Extended Supratone Modeling for HMM-based Continuous Tone Recognition," *ACM Transaction on Speech and Language Processing*, 2007. [submitted]

Chen J.C. , and J.S. R. Jang, J.Y. Li and M.C. Wu, "Automatic Pronunciation Assessment for Mandarin Chinese," *IEEE International Conference on Multimedia & Expo*, 2004, pp. 1979-1982.

Chen S.H. and Y.R. Wang. “Tone Recognition of Continuous Mandarin Speech Based on Neural Networks,” *IEEE Transactions on Speech and Audio Processing*, 3(2), 1995, pp. 146-150.

Chen J.C., J.L. Lo, and J.S. R. Jang, “以語音辨識與評分輔助口說英文學習,” In *Proceedings of Conference on Computational Linguistics and Speech Processing (ROCLING)*, 2004, available at http://www.aclclp.org.tw/rocling/2004/M25.pdf

Huang S.C., ”Improvement and Error Analysis of Tone Recognition for Mandarin Chinese”, MD thesis, National Tsing Hua University, 2006.

Huang X., A. Acero, and H.W. Hon, Chapter 12 of “*Spoken Language Processing: A Guide to Theory, Algorithm, and System Development*,” Prentice Hall PTR, Upper Saddle River, New Jersey, 2001, pp. 585-636.

Jang J.S. R., and S.S. Lin, "Optimization of Viterbi Beam Search in Speech Recognition," In *Proceedings of International Symposium on Chinese Spoken Language Processing*, 2002, paper 114.

Jang J.S. R., C.T. Sun and E. Mizutani, “*Neural-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*,” Prentice Hall PTR, Upper Saddle River, New Jersey, 1997.

Kim C. and W. Sung, "Implementation of An Intonational Quality Assessment System,” In *Proceedings of International Conference on Spoken Language Processing*, 2002, pp. 1857-1860.

Lee L.S., "Voice Dictation of Mandarin Chinese," *IEEE Signal Processing Magazine*, 14(4), 1997, pp. 63-101.

Li J.Y., “Speech Evaluation,” MD thesis, National Tsing Hua University, Taiwan, 2002.

Lin W.Y., and L.S. Lee, “Improved Tone Recognition for Fluent Mandarin Speech Based on New Inter-Syllabic Features and Robust Pitch Extraction,” In *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding*, 2003, pp. 237-242.

Neri A., C. Cucchiarini, and W. Strik, “Automatic Speech Recognition for Second Language Learning: How and Why It Actually Works,” In *Proceeding of International Congresses of Phonetic Sciences*, 2003, pp. 1157-1160.

Neumeyer L., H. Franco, V. Digalakis, and M. Weintraub, “Automatic Scoring of Pronunciation Quality,” *Speech Communication*, 30(2-3), 2000, pp. 83-93.

Rabiner L. and B.H. Juang, “*Fundamentals of Speech Recognition*,” Prentice Hall PTR, Upper Saddle River, New Jersey, 1993

Sukkar R. A. and C.H. Lee, “Vocabulary Independent Discriminative Utterance Verification for Nonkeyword Rejection in Subword Based Speech Recognition,” *IEEE Transactions on Speech and Audio Processing*, 4(6), November 1996, pp. 420-429.

Tang Poetry Corpus, 2002 Recordings, available at http://mir.cs.nthu.edu.tw/research/corpus/tangPoetry

Computational Linguistics and Chinese Language Processing
Vol. 12, No. 4, December 2007, pp. 459-484

A Knowledge-Based Approach for Unsupervised Chinese Coreference Resolution

Grace Ngai* and Chi-Shing Wang*

Abstract

Coreference resolution is the process of determining the entity that noun phrases refer to. A great deal of research has been done on this task in English, using approaches ranging from those based on linguistics to those based on machine learning. In Chinese, however, much less work has been done in this area. One reason for this is the lack of resources for Chinese natural language processing. This paper presents a knowledge-based, unsupervised clustering algorithm for Chinese coreference resolution that maximizes performance using freely and easily available resources. Experiments to demonstrate the efficacy of such an approach are performed on two data sets: TDT3 and ACE05, and the ACE value coreference resolution results achieved through our approach are 52.5% and 55.2% respectively. An oracle experiment using gold standard noun phrases achieved even more impressive results of 77.0% and 76.4%. To analyze the causes of errors, this paper also looks into false alarms and misses in documents.

Keywords: Coreference Resolution, Modified K-means Clustering, Stacked Transformation-based Learning, Unsupervised Learning

1. Introduction

Noun phrase (NP) coreference resolution is an important subtask in natural language processing (NLP) applications such as text summarization, information extraction, data mining, and question answering. The subject has attracted much attention in recent years, although much more in regards to the English language than to the Chinese language, and has been included as a subtask in the MUC (Message Understanding Conferences) and ACE (Automatic Content Extraction) programs. NP coreference resolution is the process of detecting noun phrases in a document and determining whether these noun phrases refer to the same entity. As defined in ACE [2005], an entity is "an object or set of objects in the world."

* Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong
Tel: +852-27667279 Fax: +852-22154652
E-mail: {csgngai; cscswang} @comp.polyu.edu.hk

[Received April 11, 2007; Revised August 31, 2007; Accepted September 10, 2007]

Phrases that refer to an entity are known as *mentions*, which may be either anaphors or antecedents. An anaphor is an expression that refers back to something mentioned previously in a discourse, and the something that the anaphor refers back to is its antecedent. Thus, in the passage in Figure 1, the term 克林頓總統 *(President Clinton)* in the second line of the passage is an anaphoric reference to its antecedent 克林頓 *(Clinton)*, which begins the passage. This anaphor 克林頓總統 *(President Clinton)* is in turn the antecedent of the second 他 *(he)*. All three of these terms, 克林頓 *(Clinton)*, 克林頓總統 *(President Clinton)*, and the second 他 *(he)*, are mentions of the same entity and refer, of course, to former U.S. president Bill Clinton. Generally speaking, it is a simple matter for human beings to quickly and accurately identify such coreferences. However, the cues that are used by humans for noun phrase coreference resolution are not easily transferred to the computer. Even in English, the most heavily studied language, the accuracy of automated NP coreference resolution is currently unsatisfactory. In Chinese, which has its own particular characteristics and difficulties, NP coreference resolution is a topic where even more work remains to be done.

[克林頓 $_1$]說，華盛頓將逐步落實對[韓國 $_2$]的經濟援助。[金大中 $_3$]對[克林頓 $_1$]的講話報以掌聲。[他 $_3$]說：「[克林頓總統 $_1$]在會談中重申，[他 $_1$]堅定地支持[韓國 $_2$]擺脫經濟危機。」

[Clinton$_1$] said that Washington would progressively follow through on economic aid to [Korea$_2$]. [Kim Dae-Jung$_3$] applauded.

Figure 1. An excerpt from the text, with coreferring noun phrases annotated. English translation in italics.

Central to the development of efficient and reliable approaches to automatic NP coreference resolution is the issue of what features should be used to identify the coreference. Ng and Cardie [2002b] listed 53 features, including gender agreement, number agreement, head noun matches, semantic class agreement, positional information, contextual information, apposition, abbreviation, and others. At one extreme, efficiency alone forbids the use of all of these features; at the other, no single linguistic feature is completely reliable. With the careful selection of combinations of suitable features, there may be a tradeoff to be made between the efficiency of using fewer features and the accuracy to be obtained from using more. Before such an approach can be tested, there are a number of difficulties that need to be addressed, not the least of which being the limitations of currently available NLP applications and ontologies used in coreference resolution. For example, applications, such as named entity

recognition, and ontologies, such as WordNet and HowNet, are currently used to identify features such as semantic class. However, these identifications are not always accurate, especially where new terms, domains or languages are concerned. Domain adaptation then becomes an issue, or ontology coverage becomes less than ideal.

As already mentioned, Chinese NP coreference resolution involves certain difficulties which are not found in the English language. First, from the point of view of NLP, Chinese suffers from a lack of usable morphological and orthographic features. For example, in English, morphological features such as number agreement can indicate coreference, and this contributes to the accuracy of automatic part-of-speech (POS) tagging. Chinese, however, does not use morphological changes to indicate number agreement. As for orthography, Chinese does not, for example, use capitalization whereas English can make use of capitalization to mark elements such as proper names, place names, and abbreviations. Perhaps the greatest difficulty of written Chinese is that, unlike English, it does not mark word boundaries. Word segmentation is thus required, yet various segmentations of even a simple Chinese sentence may produce a variety of meanings, making a range of NLP tasks, for example, POS tagging, highly problematic.

A second important problem faced in Chinese NP coreference resolution is a lack of Chinese corpora (specifically coreference data sets) that are either free of charge, freely available, or sufficiently free of error for use as benchmarking data sets for training and for measuring performance. The principal reason for this is that building a reasonably large coreference corpus is a labor-intensive task, especially with regard to annotation, which cannot be undertaken by any but the largest institutions. For example, the ACE corpus from the ACE program is large and is annotated for a very comprehensive number of grammatical, semantic, and discourse features. It is available, at a cost, for use in problems involving coreference resolution.

In this paper, we propose an approach to Chinese NP coreference resolution that, with small amounts of training and time investment, can accurately identify chains of coreference in unannotated texts. The approach first uses an automatic, Penn Treebank trained parser [Zhang *et al*. 2003] to identify mentions and then filters out those that are not likely to refer to an entity using heuristic rules based on POS information. The resulting mentions are then linked into possible chains using a clustering algorithm and specific linguistic features. The advantages of this approach are, first, that the proposed algorithm is unsupervised and therefore requires no training set, relying instead on word lists, dictionaries, and gazetteers that are freely available and easily compliable; and second, that features may be easily added or deleted. This makes our method suitable for scenarios where such a system needs to be quickly compiled for a new genre or language, where pre-existing resources are not adequate. After describing the proposed system, we will demonstrate the efficacy of our algorithm by

achieving satisfactory performance on two different corpora.

The rest of the paper is laid out as follows: Section 2 gives an overview of the previous work in this area. Section 3 describes our algorithm, Section 4 introduces the experimental setup, and Section 5 gives details of our evaluation. Section 6 contains the analysis of our results, and is followed by our conclusions.

2. Previous Work

In this section, we will start with a description of the most common approaches to coreference resolution and contrast them with the approach that we will be taking. Since we will be concentrating on the problem in Chinese, we will also include an introduction to the work conducted to date on Chinese NP coreference resolution.

2.1 Supervised Machine Learning Approaches

Much of the previous work in NP coreference resolution has used statistical, machine learning approaches, and one of the most frequently used approaches is that of binary classification. These algorithms link up mentions into coreference chains by first identifying an anaphoric noun phrase, and then using a predetermined number of features in an effort to identify the best antecedent for each mention. Soon *et al.* [2001] proposed a 12-feature classifier based on a decision tree, which returns a number between 0 and 1 to indicate the likelihood that two noun phrases corefer. Their training data came from and was applied to the MUC corpora. Positive examples were generated from each anaphoric NP_j and its immediately adjacent antecedent NP_i. Negative examples were generated by taking all noun phrases between each antecedent-anaphor pair, NP_{i+1}, NP_{i+2} … NP_{j-1}, and pairing them with the anaphor, NP_j. They found that the *alias*, *appositive*, and *string match* features contributed the most to performance. Ng and Cardie [2002b] extended this approach with three extra-linguistic changes: the clustering approach, the creation of training instances, and the definition of string match features. They also made use of additional features. Their system achieved good results on the MUC-6 and MUC-7 data sets, with F-Measure scores of 70.4 and 63.4, respectively. Ultimately, however, binary classification is flawed in that, at any given time, it takes into account only the relationships between two NPs rather than a longer chain. For example, given three NPs: NP_a, NP_b, and NP_c, it is possible that the model might think that NP_a and NP_b are coreferential, and also that NP_b and NP_c are coreferential, yet at the same time think that NP_a and NP_c are not. This creates a problem when the system tries to create coreference chains where all of the phrases in the chain refer to the same entity. Second, a phrase by itself usually lacks sufficient descriptive information to allow a completely confident decision to be made. Where the reference is to a human, it can be quite difficult to decide if two pronouns are anaphor-antecedent pairs simply by looking at the pronoun alone.

Several approaches have been proposed to compensate for these failings of the NP-NP approach. Yang *et al*. [2004b] adopted an NP-cluster framework, which considers the relationships between phrases and coreferential clusters. To describe the cluster properties, they introduced six additional features: cluster gender, number, semantic agreement, cluster length, cluster string similarity, and longest phrase similarity. Experiments have shown that this approach outperforms the NP-NP based approach. McCallum and Wellner [2004] introduced three conditional undirected graphical models of identity uncertainty based on conditional random fields. Their model avoids the problem of pair-wise coreference decisions being made independently of the relationships of each element of a pair. Rather than making a decision based on a single measurement to one other node, measurements are made to all nodes. This method improves upon the NP-NP based algorithm, but its supervised approach requires access to a large amount of data in order for meaningful statistics to be gathered.

2.2 Unsupervised Machine Learning Approaches

Supervised methods to coreference resolution have been successful at achieving good performance; however, they require annotated corpora as training data. This is not a problem with well-studied languages such as English, where language resources such as corpora and linguistics tools are plentiful, but it does create problems for other languages or even for less well-studied genres and domains.

Cardie and Wagstaff [1999] proposed an unsupervised approach that casts the problem of coreference resolution as a clustering task that applies a set of incompatibility functions and weights in the distance metric. Their algorithm starts by forming each entity into a singleton cluster, and then iteratively compares pairs of clusters. If the distance between two phrases in two clusters that are being compared is less than some threshold, the clusters are merged, provided that all their phrases are compatible. This mechanism can easily incorporate new constraints and preferences, but the merging algorithm is greedy in that it will take the first match rather than the best match.

2.3 Knowledge-Based Approaches

In addition to machine learning, knowledge-based approaches have also been widely used to provide rules for filtering features for NP resolution. Zhou and Su [2004] presented a constraint-based multi-agent strategy. This strategy first uses general heuristics such as morphological and semantic consistency to filter out invalid antecedent candidates, and then an antecedent for the anaphor is chosen based on the principle of proximity. This strategy offers two different types of agents: a set for filtering out less informative antecedent candidates and another set for matching coreference types. This strategy has been shown to be efficient and accurate. In addition, Bean and Riloff [2004] pioneered an approach to identify

NP coreferences by using information extraction patterns to identify contextual role knowledge. This approach first identifies definite, non-anaphoric noun phrases, and then uses case resolution to identify the most easily resolved phrases. Remaining non-resolved phrases are then evaluated against eleven sources of knowledge that include four contextual caseframes, that is, normalized extraction patterns. The final resolution is made using a Dempster-Shafer probabilistic model [Bean and Riloff 2004].

Knowledge-based approaches have the advantage in that, usually, little or no annotated corpora are required. However, they do rely heavily on hand-crafted heuristics or rules, which also require large investments of time and effort to create.

2.4 Feature Selection

The most desirable features for use in coreference resolution are robust and inexpensive, perform well over various domains, and can be obtained automatically. Features may be lexical, grammatical, semantic, syntactic, contextual, or heuristic. Given the broad range of features that may be chosen, there is currently no definitive classification of their relative merits or effects on system performance.

2.5 Coreference in Chinese Texts

To our knowledge, the previous work that has been conducted on the subject describes only two approaches to Chinese noun phrase coreference resolution, with both of them being supervised methods. Florian *et al*. [2004] used a language-independent framework to process Chinese data on the Entity Detection and Tracking (EDT) task, which is very similar to coreference resolution. EDT contains two subtasks, detection and tracking. The Entity Detection subtask finds all possibly coreferring phrases. The Entity Tracking subtask combines the detected phrases into groups referring to the same object. The authors formulate the detection subtask as a classification problem using a Robust Risk Minimization classifier combined with a Maximum Entropy classifier. Much like base noun phrase chunking, it labels each word token, indicating whether it starts a phrase, is inside a phrase, or is not within any phrase. They tackle the mention tracking subtask with a novel statistical approach that processes each phrase in turn, starting with the leftmost phrase in a document. For the current phrase, they make a decision to either link it with one of the existing clusters, or to make it start a new cluster. The authors reported achieving good results with English, Chinese, and Arabic. They obtained 58.8 on the ACE03 evaluation data on Chinese, but they noted that their algorithm was trained on only 90k characters for Chinese, in contrast to 340k words in English, which they believe to be insufficient for purposes of generalization.

Zhou *et al*. [2005] proposed a unified transformation-based learning (TBL) framework and tested it on Chinese EDT. They considered five types of entities: person, geographic or

political entity, organization, location, and facility. They use the MSRSeg word segmentation algorithm and integrate it with an adapter to chunk Chinese characters into words. The mention detection model then tags each segmented word with a semantic type. The TBL tracking model then looks at every pair of words and classifies them as being coreferent or not, based on the values of six features (string match, edit distance, token distance, mention type, entity type, and lexical string). They report a performance of 63.3 on the ACE03 data set.

One of the biggest obstacles in Chinese noun phrase coreference resolution is that the amount of available data and resources lags far behind what is available in English. As a comparison, the ACE03 training corpus for Chinese was 90k characters, compared with over 340k words for English. In addition, there are many gazetteers and lexicons available in English but not many for other languages. These factors combine to make it difficult to get good performance in supervised efforts at noun phrase coreference in languages other than English.

2.6 Evaluation Metrics

Noun phrase coreference resolution is unlike other NLP tasks in that it does not decompose readily into either a task of bracketing or classification. As a result, it is not easy to extend current evaluation metrics to noun phrase coreference resolution. In this section, we will look at two of the most common evaluation metrics and explain how they work.

Traditionally, performance of noun phrase coreference resolution has been measured using precision and recall, as measured by Vilain *et al*.'s scoring algorithm [Vilain *et al*. 1995]. The algorithm defines recall as follows:

$$R = \frac{\sum(|C_i| - |p(C_i)|)}{\sum(|C_i| - 1)}. \quad (1)$$

Each C_i is a gold standard entity (*i.e*., a set of mentions that we know refer to the same entity), and $p(C_i)$ is the partitioning of C_i by the automatically identified entities. For example, suppose that the gold standard annotation identifies two entities, C_1 and C_2, where C_1 contains the mentions {1,2,3,4,5} and C_2 contains the mentions {6,7,8,9,A,B,C}. Now, assume that the automatically identified entities are partitioned as {1,2,3,4,5} {6,7} {7,8,A,B,C}. $|C_1|$ would therefore be 5, and $p(C_1)$ would be 1. Likewise, $|C_2|$ would be 7 and $p(C_2)$ would be 2. The recall for this scenario would then be calculated to be 90%. For precision, the roles of the automatically identified and gold standard entities are reversed.

Vilain *et al*'s evaluation metric was used for the MUC program, but as Baldwin *et al*. [1998] pointed out, it does have the weakness of yielding unintuitive results for some scenarios. For example, the baseline method of assuming that all identified mentions refer to the same entity actually yields a fairly good result by Vilain's metric. There are several

reasons for this counterintuitive result: first, the metric does not distinguish between different kinds of errors; second, it inherently favors outputs with fewer entities; and third, it ignores single-mention entities.

The ACE program introduced a different evaluation metric, the ACE value [ACE 2005], which has often been referred to as a cost-based metric. The idea is to evaluate system output by application value. A system with a completely correct output would get an ACE value of 100%, while a system producing no output would get an ACE value of 0%. Negative ACE values can also be given to systems with outputs that are drastically incorrect. The overall value is calculated by looking at each of the system-generated entities and calculating its value based on a product of two factors:

$$Value_{sys\_entity} = Entity\_Value(sys\_entity) \cdot Mentions\_Value(\{sys\_mentions\})$$

Entity_Value is a function calculated over each gold standard entity. It takes into account how well the gold standard and system outputs match each other on the entity level (*e.g.* whether the mentions in the entity were detected and resolved correctly by the system). *Mentions_Value* is a function measuring how well the mentions detected by the system match those of the gold standard (*e.g.* they may match, the system may identify extra mentions, or may miss some altogether). Errors that are penalized are misses (mentions that are in the gold standard but not in the system output), false alarms (mentions that appear in the system output but not in the gold standard), and mistakes (inexact overlaps between system output and gold standard). The heaviest penalties come from misses and false alarms, with misses penalized at a heavier rate than false alarms.

Even though the ACE value was developed partly to correct some of the drawbacks of the MUC metric, it does have a number of problems of its own. One of the biggest complaints is that ACE values are difficult to interpret. For example, if a system achieves an ACE score of 90%, this does not mean that the system correctly identified 90% of the entities and mentions in the corpus, but rather, that the cost of the system is 10% of one that does not give any output [Luo 2005]. Other criticisms are that it tends to be inconsistent in how it penalizes the systems for various mistakes [Zelenko 2005].

Despite all of the problems associated with its use, the ACE score remains the most widely used and accepted metric for evaluating noun phrase coreference system performance. Therefore, we will use this metric for our own evaluations.

3. Our Algorithm

Coreference resolution, although often referred to as a single task, can actually be divided into two subtasks. The first is entity or mention detection, which identifies anaphors and antecedents in a document, followed by noun phrase coreference resolution, or mention

tracking, whereupon we decide upon the entities referred to by the identified phrases. Since trying to tackle both subtasks at once would necessitate the drawing up of an extremely complex model, almost all approaches in previous work have handled the two phases separately. Our algorithm will follow its predecessors and do the same.

3.1 Mention Detection

To start off the mention detection phase, we had our corpus parsed by a probabilistic Chinese parser [Zhang *et al*. 2003], which was trained on the Chinese Penn Treebank. As a precursor to doing a full parsing, the parser also performs word segmentation and POS tagging. The parser generates a full parse tree as its output. Since mentions usually correspond to noun phrases, we could simply have extracted all noun phrase chunks identified by the parser; however the boundaries of the parsed noun phrases do not usually correspond exactly with mention boundaries. In addition, since we followed the ACE conventions of only considering mentions that correspond to certain semantic types [ACE 2005], it is not too likely that all of the noun phrases are going to correspond to useful mentions. For example, the word 世界 *(world)*, although a noun phrase, is not tagged as a mention when it is not being used in the sense of a geographical location. We, therefore, used a filtering approach to identify and remove these spurious noun phrases. Filtering approaches have been successfully used by Bean and Riloff [1999], who used an unsupervised filter to construct a list of non-anaphoric phrases and NP patterns from an unannotated training corpus to identify mentions in definite noun phases. For their part, Ng and Cardie [2002a] employed a decision tree to filter out non-anaphoric phrases. Their approach achieved a large improvement in precision, but at a significant cost to recall.

The objective of filtering identified noun phrases is to identify only the noun phrases that are likely to correspond to mentions, while discarding the rest. Since the following phase, mention resolution, will work on top of these identified mentions, it is reasonable to aim for as accurate a performance on this phase as possible. The problem, however, is that precision and recall are usually inversely proportional to each other: having good precision usually means bad recall and vice-versa, and a balanced precision/recall performance usually means mediocre figures for both.

Our principle was this: the mention resolution phase will not identify additional mentions, and the ACE metric penalizes misses more heavily than false alarms. Therefore, we would go for high recall during the detection phase to minimize misses in the system output. To achieve this, we used a few simple heuristics to filter out noun phrases that are extremely unlikely to correspond to mentions. These heuristics are mostly based on the POS tags of the words, were previously developed for unrelated work in English named-entity resolution, and were not written with foreknowledge of the gold standard entities. A list of the heuristics can be found

in Appendix 1.

In addition, in order to filter out spurious phrases, a stoplist was used to discard frequently occurring noun phrases such as 前提 *(the aforementioned)*, 什麼 *(what)*, 特色 *(feature)*, and 同時 *(at the same time)*. In addition, we also used a large gazetteer compiled from web sources to correct segmentation errors in proper names: *e.g*. to correct nr(埃斯特) v(拉) v(達) to (埃斯特拉達*, T. Estrada, former Cuban president*).

3.2 Mention Resolution

Once mention detection has been completed, the next step in the pipeline is that of mention tracking or resolution. In this step, the task of the system is to determine which noun phrases refer to the same entity, or are coreferent.

As defined by Trouilleux *et al*. [2000], "referential chains" are sets of expressions, or mentions, that denote the same referent. That is, given a text T, for each referential chain RC there exists a unique discourse referent DR, such that:

$$RC = \{x \mid x \text{ is an expression denoting } DR \text{ in } T\}. \quad (2)$$

While most referential chains contain multiple elements, a referential chain may also consist of a single expression. For example, in the sentence "彼得愛加菲貓" (*Peter likes Garfield*), the set {彼得 (*Peter*)} is a referential chain. The task of coreference resolution consists of identifying these sets, which are also called "coreference chains."

Our algorithm relies on an unsupervised clustering approach for this task, which is a natural choice as it partitions the data into groups. For mention tracking, we expect the clustering algorithm to gather coreferent phrases into the same cluster, where each cluster will hopefully correspond to one coreference chain.

3.3 Modified K-Means Clustering

Most of the previous work in clustering-based noun phrase coreference resolution has centered around the use of bottom-up clustering methods [Cardie and Wagstaff 1999; Angheluta *et al*. 2004], where each noun phrase is initially assigned to a singleton cluster by itself, and clusters that are "close enough" to each other are merged.

In our system, we use a method called modified k-means clustering [Wilpon and Rabiner 1985], which takes the opposite approach and uses a top-down approach to split clusters, interleaved with a k-means iterative phase. Modified k-means clustering has been successfully applied to speech recognition. Compared with k-means clustering, modified k-means has the advantages of neither requiring a pre-set number of clusters nor being dependent upon an arbitrary starting state [Fung *et al*. 2003].

Modified k-means starts off with all of the instances in one big cluster. The system then iteratively performs the following steps:

1. For each cluster, find its centroid, defined as the instance that is the closest to all other instances in the same cluster.
2. For each instance:
    a. Calculate its distance to all of the centroids.
    b. Find the centroid with the minimum distance, and join its cluster.
3. Iterate 1-2 until instances stop moving between clusters.
4. Find the cluster with the largest intra-cluster distance, defined as the mean of the distances of all the instances in the cluster to the centroid instance. (Let this cluster be called $Cluster_{max}$ and its centroid, $Centroid_{max}$.)
    a. If the intra-cluster distance of $Cluster_{max}$ is smaller than some pre-set threshold r, stop.
5. Calculate the distances between all pairs of instances inside $Cluster_{max}$ and find the pair of instances that are the furthest apart.
    a. Add the pair of instances to the list of centroids and remove $Centroid_{max}$ from the list.
6. Repeat from Step 2.

The algorithm thus alternates traditional k-means clustering with a step that adds new clusters to the pool of existing ones. Used for coreference resolution, it splits up the instances into clusters in which the instances are more similar to each other than to instances in other clusters.

The next step is to determine a suitable threshold and a distance function with suitable parameters. As functions that check for compatibility return negative values while positive distances indicate incompatibility, a threshold of 0 would separate compatible and incompatible elements. However, since the feature extraction will not be totally accurate, we chose to be more lenient with deciding whether two phrases should be clustered together (*i.e.*, to go for recall over precision) and used a threshold of $r = 1$ to allow for possible errors.

3.4 Feature Selection

One of the advantages of using a clustering algorithm is that most clustering methods can easily incorporate both context-dependent and independent constraints into their features. This is attractive for us since we use a variety of features, which are designed both to capture the content of the phrase and its role within the sentence and document.

Most of our features give us information on a single phrase:

- **String Content** – The string of words in the phrase.
- **Head Noun** – The head noun in a phrase is the noun that is not a modifier for another noun.
- **Sentence Position** – The position of the sentence that contains the phrase, relative to the document. The first sentence is in position 1, the second in position 2, and so on.
- **Gender** – For each phrase, we use a gazetteer to assign it a gender. The possible values are male (*e.g.*, 先生, *mister*), female (*e.g.*, 小姐, *miss*), either (*e.g.*, 團長, *leader*), and neither (*e.g.*, 工廠, *factory*).
- **Number** – A phrase can be either singular (*e.g.*, 一隻貓, *one cat*), plural (*e.g.*, 兩隻狗, *two dogs*), either (*e.g.*, 產品, *product*) or neither (*e.g.*, 安全, *safety*).
- **Semantic Class** – To give the system more information on each phrase, we compiled our own gazetteer from web sources. Our gazetteer consists of 12,000 entries, each of which is labeled with the following semantic classes: person, organization, location, facility, GPE, date, money, vehicle, and weapon. Phrases in the corpus that are found in the gazetteer are given the same semantic class label; phrases not in the gazetteer are marked as *unknown*.
- **HowNet Definition** – The semantic class gazetteer covers about 80% of the phrases that are extracted. To increase the coverage of the phrases, we turned to HowNet [Dong and Dong 2000], an ontological knowledge base that encodes inter-conceptual relations and inter-attribute relations for the Chinese language. HowNet contains 120,496 entries for about 65,000 Chinese words defined with a set of 1503 sememes, which are considered atomic semantic units that cannot be reduced further. Examples of such sememes are "human," or "aValue" (attribute-value). Higher-level concepts, or definitions, are composed of subsets of these sememes, sometimes with pointers that denote certain kinds of relationships, such as "agent" or "target." For example, the word "疤" is associated with the definition "trace|疤, #disease|疾病, #wounded|受傷." As an additional feature, we labeled phrases that appeared as HowNet concepts with their sememe definitions. Phrases that do not exist in HowNet are marked as *unknown*. Overall, we found that about 66% of the extracted mentions in our corpus were covered under HowNet.
- **Proper Noun** – The part-of-speech tags "nr" (person name), "ns" (country name), "nt" (organization name), "nz" (other proper name), and a list of common proper names compiled from the Internet were used to label each noun phrase, indicating whether or not it is a proper noun.
- **Pronoun** – The part-of-speech tag "r" (pronoun) is used to determine whether the phrase is indeed a pronoun.

- **Demonstrative Noun Phrase** – A demonstrative noun phrase is a phrase that consists of a noun phrases preceded by one of the characters [此這那該] (*this/that/some*).

The following features give us information on how two phrases relate to each other:

- **Appositive** – Two noun phrases are in apposition when the first phrase is headed by a common noun, while the second one is a proper name and there no space or punctuation between the two phrases; *e.g*., [美國總統][克林頓]上星期到朝鮮訪問 (*[US president] [Clinton] visited Pyongyang last week*). This differs from English, where two nouns are considered to be in apposition when one of them is an anaphor and separated by a comma from the other phrase, which is the most immediate proper name; *e.g*., "Bill Gates, the chairman of Microsoft Corp".
- **Abbreviative** – A noun phrase is an abbreviation when it is formed using part of another noun phrase; *e.g*., 朝鮮中央通訊社 (*Pyongyang Central Communications Office*) is commonly abbreviated as 朝中社. Since name abbreviations in Chinese are often given in an ad-hoc manner, it would be infeasible to generate a list of names and abbreviations in advance. We, therefore, use the following heuristic: given two phrases, we test if one is an abbreviation of another by extracting each successive character from the shorter phrase and testing to see if it is included in the corresponding word from the longer phrase. Intuitively, we know that this is a common way of abbreviating terms; empirically, we found it to be a highly precise test: a positive result was very rarely wrong.
- **Edit Distance** – Abbreviations and nicknames are very commonly used in Chinese and even though the previous feature will work on most of them, there are some common exceptions. For example, some name-abbreviation pairs that would not get picked up are 北大西洋公約組織 (*North Atlantic Treaty Organization*) and 北約, or 奧運會 (*Olympics*) and 奧運. To make sure that those are caught as well, we introduced a Chinese-specific feature as a further test. Since abbreviations and nicknames are not usually substrings of the original strings but will still share some common characters, we measure the Levenshtein distance, defined as the number of character insertions, deletions, and substitutions, between every potential antecedent-anaphor pair.

To calculate the distance between two noun phrases, a set of functions is defined over the features. For features that give information on a single mention, functions compare the value of the same feature over a pair of phrases. For features defined relative to two mentions such as *edit distance* and *appositive*, the function simply returns the value of the feature itself.

The idea behind the functions is this: some features are indicators of whether two phrases are compatible with each other, with respect to coreferentiality. These features are *string*

content, *head noun*, *demonstrative*, *appositive*, *abbreviation*, and *edit distance*. If two phrases match on this particular feature (for example, if the *head noun* feature for NP_i and NP_j are identical), then this is a strong indicator that these two phrases are coreferential. However, if they do not match, this does not necessarily mean that the two phrases are non-coreferential. Hence, these functions return negative values (decreasing the distance) when the two phrases match, but 0 (neutral) when they do not.

Table 1. Features and Functions Used for Clustering.

Feature *f*	**Function (*Incompatibility*$_f$*(NP$_i$, NP$_j$)*)**
String Match	-1 if the string of NP_i matches the string of NP_j; else 0
Head Noun Match	-1 if the head noun of NP_i matches the head noun of NP_j; else 0
Sentence Distance	0 if NP_i and NP_j are in the same sentence; For non-pronouns: 1/10 if they are one sentence apart; and so on with a maximum value of 1; For pronouns: if more than two sentences apart, then 1
Gender Agreement	1 if they do not match in gender; else 0
Number Agreement	1 if they do not match in number; else 0
Semantic Agreement	1 if they do not match in semantic class; else 0
HowNet Definition	1 if neither phrase is labeled as *unknown* and all of the sememes do not match, else 0.
Proper Name Agreement	1 if both are proper names, but mismatch on every word; else 0
Pronoun Agreement	1 if either NP_i or NP_j is a pronoun and the two mismatch in gender or number; else (e.g. if either one is unknown, or either one is not a pronoun), 0
Demonstrative Noun Phrase	-1 if NP_i is demonstrative and NP_i contains NP_j; else 0
Appositive	-1 if NP_i and NP_j are in an appositive relationship; else 0
Abbreviation	-1 if NP_i and NP_j are in an abbreviative relationship; else 0
Edit Distance	-1 if NP_i and NP_j are the same, -1/(length of longer string) if one edit is needed to transform one to another, and so on.

On the other hand, there are some features where a mismatch would strongly indicate that the two NPs are non-compatible and are not likely to refer to the same entity. The *gender*, *number*, *semantic*, *HowNet*, *proper name*, *pronoun*, and *sentence distance* features are all indicators of non-compatibility; hence, their associated functions return positive values, increasing the distance and making it less likely that the two phrases will be grouped into the same cluster.

Table 1 presents details of the features and the corresponding functions that were used in our system. Combining the values of all these functions gives us the distance between two phrases, with greater distances indicating greater incompatibility. For our system, we borrowed a simple distance metric from Cardie and Wagstaff [1999] that sums up the results of a series of functions over the two phrases:

$$dist(NP_i, NP_j) = \sum_{f \in F} w_f * incompatibility_f(NP_i, NP_j) \quad (3)$$

where w_f is the weight of that particular feature (all features carry equal weight for us), and $incompatibility_f(NP_i, NP_j)$ is the result of the function corresponding to that feature when those two noun phrases are considered.

To summarize our efforts thus far, we have proposed an approach that adapts an unsupervised machine learning method for Chinese coreference resolution under limited resources. We have proposed a new methodology for mention detection and designed new features for mention resolution that are specifically geared towards our task.

4. Experimental Setup

To validate our algorithm, two data sets are used for evaluation. The first data set is an annotated version of TDT3 Chinese corpus, which was created by selecting 30 documents from the TDT3 corpus and then having it annotated by a native Chinese speaker following the MUC-7 [Hirschman and Chinchor 1997] and ACE Chinese entity guidelines [NIST 2005a]. We annotated proper nouns, nominal nouns, and pronouns, and according to MUC-7 guidelines, each phrase participates in exactly one entity, and all phrases in the same entity are coreferent. Using the MUC and ACE guidelines, we annotated noun phrases of the following nine types of entities, which are a combined set of those used in MUC and ACE:

- **Person** – Humans.
- **Organization** – Corporations and groups of people defined by an organizational structure.
- **Location** – Geographical areas, landmasses, and bodies of water.
- **Geopolitical entity (GPE)** – Comprised of a population, a government, a physical location, and a notion.
- **Facility** – Buildings and man-made structures.
- **Vehicle** – Physical devices designed to move an object from one location to another.
- **Weapon** – Physical devices used as instruments for physically harming or destroying.

- **Date** – Numbered days with a combination of the name of the day, the month, and the year.
- **Money** – Amounts of cash or currency.

The second corpus comes from the Chinese data in the ACE05 Entity Detection and Recognition evaluation task, which is similar to coreference resolution. This task requires that seven types of entities that are mentioned in the source data be detected and that the selected noun phrase about these entities be organized into a unified representation. The seven types of entities are *facility*, *GPE*, *location*, *organization*, *person*, *vehicle*, and *weapon*. The source data consists of three domains: newswire, broadcast news, and weblogs. For our experiments, we used the newswire and broadcast news domains. Table 2 shows some statistics from the corpora.

Table 2. Corpus Statistics

	Annotated TDT3	**ACE05 nwire**	**ACE05 bnews**
Documents	30	69	73
Character	23k	36k	32k
Entity	592	2044	1632
Mention	2997	4347	3678
Semantic Classes	32.7% person, 33.9% GPE, 13.5% organization, 7.7% facility, 3.9% location, 2.7% vehicle, 3.9% weapon, 1.1% date, 0.5% money	40.8% person, 30.7% GPE, 17.2% organization, 3.6% facility, 5.7% location, 1.7% vehicle, 0.3% weapon	44.5% person, 24.3% GPE, 17.9% organization, 7.7% facility, 4.6% location, 2.0% vehicle, 0.9% weapon

5. Evaluation

Since our algorithm breaks down the coreference resolution task into two subtasks, we will evaluate them separately and also investigate how or whether mistakes made in one subtask affect performance in the other.

5.1 Mention Detection

The subtask of mention detection is similar to that of noun phrase chunking, and we will evaluate it in the same fashion. We compare the output of the algorithm with the gold standard mentions, and count the number of mentions that are correctly identified. As an evaluation measure, we use the usual precision, recall, and f-measure metrics:

$$F = \frac{2PR}{P+R}. \tag{4}$$

Table 3 shows the results of the mention detection subtask achieved by our system on the TDT and ACE corpora, respectively.

Table 3. Mention Detection Results

	Recall	**Precision**	**F-Measure**
Annotated TDT3	88.5	48.4	62.6
ACE05 nwire	77.5	65.5	70.8
ACE05 bnews	73.8	64.0	68.5

5.2 Mention Resolution

As described in the section on Previous Work, both of the most commonly used noun phrase coreference resolution metrics have their detractors. In our work, we chose to use the ACE metric, which is currently the most widely accepted metric for this task.

Table 4 presents the performance of the second phase of our algorithm – the mention detection subtask – as measured by the official ACE05 scoring program. The entry "Our Algorithm" corresponds to the performance of our algorithm for each of the separate corpora. To get a sense of the difficulty of the task, we present a baseline system that simply assumes that mentions are coreferent if the "String Match" function (the most indicative feature) tests true. From the results, it can be seen that our system achieves a performance gain of over 20% on both the TDT3 and ACE05 newswire corpora, and over 10% on the ACE05 broadcast news corpora.

Table 4. Coreference Resolution Performance

Corpus	**Experiment**	**ACE value**
TDT3	Our Algorithm Baseline (string match only) Gold Standard Entities (upper bound)	52.5 43.7 77.0
ACE05 nwire	Our Algorithm Baseline (string match only) Gold Standard Entities (upper bound)	55.3 46.3 75.6
ACE05 bnews	Our Algorithm Baseline (string match only) Gold Standard Entities (upper bound)	55.1 49.0 77.2

Another point of comparison can be made when we compare the results obtained by our entire algorithm against the performance obtained *if we had performed the mention detection on gold standard entities*. The performance for this experiment is illustrated in the "Gold Standard Mentions" entry, and it gives us an idea of the upper bound that we could potentially achieve if we got 100% accuracy on the mention detection subtask. From the figures, it can be seen that there is substantial degradation of the overall performance of the algorithm as a result of errors in the first subtask cascading down the second subtask. This propagation of errors in pipelined systems is well known and documented.

6. Analysis

One interesting question to ask about the results is the contribution of any given individual feature to the result of the overall system. We have already investigated the effect of mention detection on the overall performance, and in this section we take a look at the features for the clustering algorithm used in the mention tracking subtask.

Table 5. Analysis: Contribution of each feature

Feature Removed	ACE score (TDT3)	Change	ACE score (ACE05bn)	Change
String Match	**71.9**	**-5.1**	**68.2**	**-9.0**
Head Noun Match	74.8	-2.2	75.5	-1.7
Sentence Distance	74.0	-3.0	75.1	-2.1
Gender Agreement	75.9	-1.1	74.1	-3.1
Number Agreement	75.5	-1.5	76.8	-0.4
Semantic Agreement	**71.1**	**-5.9**	**69.4**	**-7.8**
Proper Name Agreement	*76.7*	*-0.3*	*76.9*	*-0.3*
Pronoun Agreement	*76.6*	*-0.4*	*77.0*	*-0.2*
Demonstrative Noun Phrase	*76.0*	*-1.0*	*76.7*	*-0.5*
Appositive	73.2	-3.8	73.9	-3.3
Abbreviation	75.1	-1.9	76.5	-0.7
Edit Distance	**72.7**	**-4.3**	**71.7**	**-5.5**
HowNet Class	73.5	-3.5	74.3	-2.9
None (All Features)	77.0	--	77.2	--

In order to get a result that reflects the contribution of each feature alone, and to ensure that any conclusions we draw are extendable to other corpora, we performed a series of experiments of the mention tracking subtask on the gold standard entities of the TDT3 and the broadcast news portion of the ACE05 corpora. The first experiment was performed using all the features that were available to us, and then, one at a time, features were removed from the

clustering algorithm.

Table 5 presents the results of the experiments. The last entry in the table shows the results of the full system; the drop in performance when a feature is removed is indicative of its contribution.

Judging from the results, the three features that contribute the most to performance are the *string match*, *semantic agreement*, and *edit distance* features. Two out of the three, *string match* and *edit distance*, operate on lexical information. The importance of string matching to coreference resolution is consistent with the findings in previous studies [Yang *et al*. 2004a], which arrived at the same conclusion for English. Edit distance, which captures certain phenomena not covered by string match, has also been found to be effective by Strube *et al*. [2002] for English coreference resolution, though their results focus on words rather than characters. The fact that *string match* and *edit distance* represent some overlapping information is not a problem for the k-means clustering algorithm, as it does not assume independent features.

Of our features, those that contribute the least to the overall performance of the system are the *proper name agreement* and *pronoun agreement* features. The reason for this is that the information of these features is already covered by *string match* and *head noun match*; thus, there are not enough distinct examples for them to make any significant impact.

In addition to feature coverage, another factor in determining the performance of the system is accuracy – both in the mention detection subtask as well as in feature generation. We have already seen the drop in system performance as a result of incorrectly identified mentions. For feature generation, we know that some of our features are always going to be generated correctly (for example, *string match* or *edit distance*), while others, such as *number agreement*, are generated using heuristics or gazetteers; therefore, even the values of the features themselves will be prone to errors.

To get a better sense of the source of errors, we randomly selected two documents in our corpus for closer examination. This revealed to us the reason why the *gender*, *number*, and *semantic class* features were not as useful as we had first thought they would be. Table 6 shows some of the statistics from our examination. Over 80% of the identified mentions are tagged with the correct value for the aforementioned features, which is a positive sign. However, the ability of the features to determine whether two mentions refer to the same entity is decreased by the coarse resolution of the feature values: about 50% of the mentions are tagged as *neither* for *Gender*, over 60% are tagged as *singular* for *Number*, and almost 70% of the mentions are either tagged as *person* or *GPE*.

Table 6. Automatic Feature Generation Statistics from Sampled Documents

Feature	Remarks
Gender agreement	50.4% neither, 22.8% either, 22.2% male, 4.6% female
Number agreement	66.0% singular, 14.4% neither, 11.1% either, 8.5% plural
Semantic agreement	50.3% person, 17.6% GPE, 12.4% organization, 7.8% location, 4.5 % unknown, 7.4% others
HowNet	41.0% unknown
Abbreviation	75% correct
Appositive	84.4% correct

The same table also shows why the *HowNet* feature does not contribute much to the performance of the system: its coverage is very limited, as only about 41% of the mentions exist as concepts in HowNet and thus receive a feature value.

On the positive side, it is heartening to see that our heuristics for checking for abbreviations and apposition work well: *abbreviation* was correctly tagged 75% of the time, and *apposition* achieved an accuracy of almost 85%.

The investigation also revealed the extent of segmentation errors upon our system performance. Upon examination, it was found that 35.2% of the missing link errors and 24.0% of the spurious link errors had been caused by segmentation errors. This finding illustrates the importance of the preprocessing step, and it also demonstrates the difficulty involved with working with relatively resource-poor languages or genres.

The length of the mentions in the examined documents also provides us with clues as to where our system could be improved. Our system relies heavily on lexical features, which work best with long strings of many characters. However, the mentions in the documents average a little over two characters in length. The result is that the lexical features have limited usefulness, at least in our document.

Another apparent problem with our approach is that almost all our features are designed to describe intra-mention information. The problem with this approach is that determining coreference resolution uses quite a lot of contextual information. For example, one of the entities in our two randomly-sampled documents was the one referring to 陳水扁總統 (*Taiwanese president Chen Shui-bian*). The mentions referring to this entity include 總統 (*president*), 陳總統 (*president Chen*), 我 (*myself*), 陳 (*Chen*), 他 (*him*), 一個台南小孩 (*a child from Tainan*), 導游 (*tour guide*), as well as 陳水扁 (*Chen Shui-bian*). While intra-mention information can (and does) distinguish 總統 (*president*), 陳總統 (*president Chen*), 陳 (*Chen*), and 陳水扁 (*Chen Shui-bian*) as referring to the same entity, it is not possible to realize that the other mentions also refer to this entity without using contextual information. The result is that these other mentions end up being separated out into singleton

entities – entities with just one mention in them. This is a direction that we are definitely planning to work on in the future.

Table 7. Comparison of our system with results reported in previous work.

	ACE05 nwire	**ACE05 bnews**	**ACE03**
Our hybrid approach	**55.3**	**55.1**	
Florian *et al*. [Florian *et al*. 2004]	--	--	58.8
Zhou *et al*. [Zhou *et al*. 2005]	--	--	63.3
IBM	70.5	69.6	--
BBN Technologies	67.9	70.1	--
New York University	64.3	69.9	--
University of Colorado	64.9	57.4	--
Hong Kong Polytechnic University	51.3	50.2	--
XIAMEN University	44.8	51.0	--
Harbin Institute of Technology	44.1	48.0	--
Basis Technology, Inc.	3.0	4.7	--

To our knowledge, this is the first published result on unsupervised Chinese coreference resolution. To get a general idea of the performances achieved by other systems, Table 7 shows the performance of our system together with other previously reported results, some of which are from published reports while others are from the official evaluation of Entity Detection and Recognition task on Chinese [NIST 2005b]. It shows that our system achieves numerical results comparable to those from previous systems.

7. Conclusions and Future Work

In this paper, we have presented an unsupervised approach to Chinese coreference resolution. Our approach performs resolution by clustering, with the advantage that no annotated training data is needed. We evaluated our approach using an annotated version of TDT3 corpus and the ACE05 Chinese data, and found that our system achieves results comparable to the official results of using an unsupervised approach. We also analyzed the performance of our system by investigating the contribution of individual features to our system. The analysis illustrates the contribution of the new language-specific features, and also demonstrates that a reasonable coreference resolution system can be implemented quickly and efficiently through the use of readily-available resources.

While the results produced by our system are impressive, it is noted that all of our features consider only intra-mention information, which our in-depth analysis shows to be inadequate for coreference resolution. In future work, we plan to investigate the use of more sophisticated features, including contextual clues, to improve the performance of our system

and implement entity-based clustering.

Acknowledgements

The authors would like to thank the Hong Kong Research Grants Council for supporting this work through research grant PolyU5191/04E, and also to the three anonymous reviewers for their insightful comments and suggestions.

References

ACE, The ACE Evaluation Plan, http://www.nist.gov/speech/tests/ace/ace05/index.htm, 2005.

Angheluta, R., P. Jeuniaux, M. Rudradeb, and M.F. Moens, "Clustering Algorithms for Noun Phrase Coreference Resolution," *Proceedings of the 7th International Conference on the Statistical Analysis of Textual Data*, 2004, Louvain La Neuve, Belgium, pp. 60-70.

Baldwin, B., T. Morton, A. Bagga, J. Baldridge, R. Chandraseker, A. Dimitriadis, K. Snyder, and M. Wolska, "Description of the UPENN CAMP System as Used for Coreference." In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, 1998, Fairfax, Virginia.

Bean, D., and E. Riloff, "Corpus-Based Identification of Non-Anaphoric Noun Phrases," In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-99)*, 1999, Maryland, USA, pp. 373-380.

Bean, D. and E. Riloff, "Unsupervised Learning of Contextual Role Knowledge for Coreference Resolution," In *Proceedings of the Human Language Technology Conference / North American Chapter of the Association for Computational Linguistics Annual Meeting (HLT/NAACL-04)*, 2004, Boston, Massachusetts, pp. 297-304.

Cardie, C. and K. Wagstaff, "Noun phrase coreference as clustering," In *Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999, Maryland, USA, pp. 82-89.

Dong, Z.D., and Q. Dong, HowNet. http://www.keenage.com, 2000.

Florian, R., H. Hassan, A. Ittycheriah, H. Jing, N. Kambhatla, X. Luo, N. Nicolov, and S. Roukos, "Statistical Model for Multilingual Entity Detection and Tracking," In *Proceedings of the 2004 annual meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2004)*, 2004, Boston, Massachusetts, pp. 1-8.

Fung, P., G. Ngai, and C. Cheung, "Combining Optimal Clustering and Hidden Markov Models for Extractive Summarization," *Workshop on Multilingual Summarization and Question Answering, ACL-2003 Workshop*, 2003, Sapporo, pp. 21-28.

Hirschman, L., and N. Chinchor, MUC7 Coreference Task Definition, http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/co_task.html, 1997.

Luo, X., "On coreference resolution performance metrics,". In *Proc. of HLT/EMNLP*, 2005, Vancouver, Canada, pp. 25-32.

Mccallum, A., and B. Wellner, "Conditional models of identity uncertainty with application to noun coreference". In *Proceedings of NIPS-17*, 2004, Vancouver, Canada.

Ng, V., and C. Cardie, "Identifying Anaphoric and Non-Anaphoric Noun Phrases to Improve Coreference Resolution," *Proceedings of the 19th International Conference on Computational Linguistics (COLING-2002)*, 2002, Taipei, Taiwan.

Ng, V., and C. Cardie, "Improving machine learning approaches to coreference resolution," In *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics*, 2002, Philadelphia, pp. 104-111.

NIST, ACE Chinese Annotation Guidelines for Entities, http://www.ldc.upenn.edu/Projects/ACE/, 2005.

NIST, NIST 2005 Automatic Content Extraction Evaluation Official Results, http://www.nist.gov/speech/tests/ace/ace05/doc/ace05eval_official_results_20060110.htm, 2005.

Soon, W., H. Ng, and D. Lim, "A machine learning approach to coreference resolution of noun phrases," *Computational Linguistics*, 27(4), 2001, pp. 521-544.

Strube, M., S. Rapp, and C. Muller, "The influence of minimum edit distance on reference resolution," In *Proc. of the 2002 Conference on Empirical Methods in Natural Language Processing*, 2002, Philadelphia, pp. 312-319.

Trouilleux, F., E. Gaussier, G. G. Bes, and A. Zaenen, "Coreference resolution evaluation based on descriptive specificity," In *Proceedings of the LREC 2000 Workshop on Linguistic Coreference*, 2000, Athens.

Vilain, M., J. Burger, J. Aberdeen, D. Connolly, and L. Hirschman, "A model-theoretic coreference scoring scheme," In *Proc. of the Sixth Message Understanding Conference (MUC-6)*, 1995, San Francisco, CA. Morgan Kaufmann, pp. 45-52

Wilpon, J., and L. Rabiner, "A modified K-means clustering algorithm for use in isolated word recognition," In *IEEE Transactions on Acoustics, Speech, Signal Processing.* ASSP-33(3), 1985, pp. 587-594.

Yang, X., G. Zhou, J. Su, and C. L. Tan, "Improving noun phrase coreference resolution by matching strings," In *Proc. of the 1st Int'l Joint Conference on Natural Language Processing*, 2004, Hainan, China, pp. 22-31.

Yang, X., G. Zhou, J. Su, and C. L. Tan, "An NP-Cluster Based Approach to Coreference Resolution," *Proceedings of the 20th International Conference on Computational Linguistics*, 2004, Geneva, Switzerland, pp. 226-232.

Zelenko, D., Scoring Problems, http://cio.nist.gov/esd/emaildir/lists/ace_list/msg00849.html, 2005.

Zhang, H., Q. Liu, K. Zhang, G. Zou, and S. Bai, "Statistical Chinese Parser ICTPROP", Technical Report No. 2003.06, Institute of Computing Technology, Chinese Academy of Sciences, 2003.

Zhou, Y., C. Huang, J. Gao, and L. Wu, "Transformation Based Chinese Entity Detection and Tracking," *Proceedings of the Second International Joint Conference on Natural Language Processing*, 2005, Korea, pp. 232-237.

Zhou, G., and J. Su, "A High-Performance Coreference Resolution System using a Constraint-based Multi-Agent Strategy," *Proceedings of the 20th International Conference on Computational Linguistics (COLING-2004)*, 2004, Geneva, Switzerland, pp. 522-528.

Appendix

List of Heuristics used in Mention Detection

- Keep all non-recursive noun phrases
  e.g. 韓國外交部 (*Korean Foreign Service*), 官員 (*officials*) from NP (NP (ns(韓國 *Korea*) nt(外交部 *Foreign Service*)) NP (n(官員 *officials*))).
- Keep all quantifier phrases.
  e.g. 一名旅客 (*a certain traveler*) from NP (QP (m(一 *one*) CLP (q(名))) NP (n(旅客 *traveler*))).
- Keep all determiner phrases.
  e.g. 這次旅遊(*this tour*) from DP (r(這次 *this*)) NP (vn(旅遊 *tour*)))
- Keep all pronouns.
  e.g. r(他們 *they*), r(他 *he*), r(自己 *myself*), r(我們 *we*), r(您 *you*).
- Keep all proper noun sequences.
  e.g. 小淵惠三 *(Obuchi Keizo)* from NP (nr(小淵) nr(惠)) NP (nr(三))
- Keep all noun sequences.
  e.g. 核子設施 *(nuclear facilities)* from NP (n(核子 *nuclear*) n(設施 *facilities*)))
- Keep frequently appearing proper nouns from the gazetteer.
  e.g. 埃斯特拉達 *(T. Estrada, former Cuban president)* from nr(埃斯特) v(拉) v(達)
- Keep all sequences matching certain regular expression-like patterns.
  e.g. mq.*n: m(五 *five*) q(天 *day*) dec(的 *'s*) n(國事訪問 *official visit*); r.*n: r(其他 *other*) ns(中國 *Chinese*) n(官員 *officials*)
  (Notation: '*' is the Kleene star operator, '.' is a wildcard corresponding to a single POS tag, other characters correspond to POS tags.)
- Keep two noun phrases with POS tagging pattern *noun-propernoun-propernoun*.
  e.g. n(記者 *journalist*) and nr(陳占杰 *Chen Chanchieh*) from NP(n(記者 *journalist*) nr(陳 *Chen*) nr(占杰 *Chanchieh*))
- Keep two noun phrases with POS tagging pattern *noun-dec-noun*.
  e.g. ns(中國 *China*) and n(政策 *policy*) from NP (ns(中國 *China*) dec(的 *'s*) n(政策 *policy*))
- Keep two noun phrases with POS tagging pattern *noun-conjunctive-noun*.
  e.g. ns(中國 *China*) and ns(美國 *USA*) from NP (ns(中國 *China*) c(和 *and*) ns(美國 *USA*))
- Keep all proper nouns with POS tagging pattern *nr ns nt nz*.
  e.g. ns(新疆 *Xinjiang*) and nz(維吾爾 *Uygur*) from NP (ns(新疆 *Xinjiang*) nz(維吾

爾 *Uygur*) n(自治區 *Autonomous Region*) n(領導 *leader*)).

- Discard noun phrases with POS tag *t* inside.
  e.g. NP(t(兩日 *two days*) t(下午 *afternoon*)); NP (t(目前 *present*))
- Discard noun phrases with only quantifier characters
  e.g. NP(m(兩 *two*) q(名 *persons*)); NP (m(十 *ten*) q(年 *years*))
- Discard noun phrases starting with prepositions.
  e.g. NP (p(對 *to*) ns(中國 *China*) n(人民 *people*))
- Discard noun phrases that contain verbs or punctuations.
  e.g. NP (n(總統 *presidential*) vn(大選 *election*)); NP (ns(日本 *Japan*) w、(、) ns(香港 *Hong Kong*))
- Discard single character noun phrases excepting those that have been tagged as proper nouns or pronouns
  e.g. n(字 *character*); n(月 *moon*)
- Discard noun phrases that are found in the stoplist.
  e.g. 前提 *(the aforementioned)*, 什麼 *(what)*, 特色 *(feature)*, 同時 *(at the same time)*.
- Discard noun phrases with stopwords appearing inside them: i.e. those with 的 *(dec)*, 說 *(say)*, 經 *(after)*, 爲 *(for)*.
  e.g. NP (n(車廂 *compartment*)) f(內 *inside*)) dec(的)); NP (r(他 *he*) v(說 *says*)); NP (p(經 *after*) vn(大賽 *competition*) n(評委會 *committee*)); NP(vl(爲 *for*) n(我國 *our country*))