An integrated approach for Chinese word segmentation

Guohong Fu
Department of Linguistics
The University of Hong Kong
Pokfulam Road, Hong Kong
ghfu@hkucc.hku.hk

K.K. Luke
Department of Linguistics
The University of Hong Kong
Pokfulam Road, Hong Kong
kkluke@hkusua.hku.hk

ABSTRACT

This paper presents an integrated approach for Chinese word segmentation, which can perform disambiguation and unknown word identification simultaneously on the input. In this work, a hybrid model is used to score known word candidates and unknown word candidates equally by incorporating the modified word-formation models (viz. word-juncture models and word-formation patterns) into word bigram models, with which different types of features are statistically computed and combined for this integrated segmentation, including internal word-formation power of components in a word, affinity relations between these components and the external contextual information. To enhance the precision and avoid the problem of combination explosion in word candidate construction, a filter algorithm is also given to block ineligible unknown word candidates. In this way, ambiguity and unknown word can be resolved effectively. The results of our experiment on Peking University corpus show that the integrated approach outperforms the other two-stage methods under discussion.

1 INTRODUCTION

Chinese word segmentation aims to recognize the implicit word boundary delimiters in plain Chinese texts, which plays very important roles for most text-based applications, such as machine translation, information retrieval, text-t-speech synthesis and many more. During the past decades, many different techniques have been proposed for Chinese word segmentation, ranging from dictionary or rule based methods (Liang and Zheng 1991; Yeh and Lee 1991), statistical approaches (Fu and Wang 1999; Nie et al. 1995; Teahan et al. 2000; Wang et al. 2000; Yao 1997; Zhang et al. 2002), to machine learning approach (Hockenmaier and Brew 1998; Palmer 1997; Xue and Converse 2002). However, we are still faced with word boundary ambiguities and unknown words while developing a high-performance system for practical applications. What is more, ambiguity resolution and unknown word identification are often taken as two independent stages in previous word segmentation systems for Chinese. In this point, disambiguation is considered as a unique problem related to known word segmentation and unknown word identification is taken as a post-processing of known word segmentation. Although this two-stage strategy is simple and applicable, it usually fails to yield correct results for some complicated cases such as a mixture of ambiguities and unknown words. For instance, in a Chinese sentence 中行长葛支行注重健身, the correct segmentation for the fragment 中行长葛 should be 中行/长葛/ (zhong1hang2 chang2ge2, *Bank of China/Changge/*). However, Most two-stage systems cannot yield this segmentation because this string has been wrongly segmented as 中/行长/葛/ (zhong1 hang2zhang3 ge2, *middle/president/Ge/*) in process of known segmentation and there is no a mechanism to re-segment the word 行长 (hang2zhang3, *president*) during unknown word identification. Here, 中行 (zhong1hang2, *Bank of China*) is an abbreviation of organization name, and 长葛 (chang2ge2, *Changge*) is a place name.

Recently, a variety of methods have been reported for this problem. Wu and Jiang (1998&2000) take word segmentation, including unknown word identification as an integral part of sentence analysis. This mechanism provides a word lattice to store all the possible words and use a full sentence parsing to achieve the final disambiguation. However, the parser coverage may restrict its applications in practical systems. Furthermore, this mechanism requires extra linguistics knowledge such as part-of-speech to yield correct results for the input. Lately, Zhang et al (2002) presents a novel method for word segmentation based on role tagging. They define a set of unknown word roles about varied internal components and contexts in their system. As a result, their system can recognize different types of unknown words, including the case mentioned above, despite that their method is also a two-stage segmentation. However, a role-tagged corpus is needed in their work to learn role knowledge, which is not always available in practice.

To address above problems, this paper presents an integrated word segmentation approach for Chinese, which can perform disambiguation and unknown word identification simultaneously on the input. In this work, a hybrid model is used to score known word candidates and unknown word candidates equally, which incorporates the modified word-formation models (viz. word-juncture models and word-formation patterns) into word bigram models. In this way, different types of features are statistically computed and combined for the integrated segmentation, including internal word-formation power of component words of word candidates, affinity relations between these components and the external contextual information. Furthermore, a filter algorithm is also proposed to enhance correctness and avoid combination explosion in word candidate construction.

The rest of this paper is organized as follows: Section 2 focuses on statistical modelling for integrated segmentation. Section 3 describes in detail the algorithm for integrated word segmentation. In section 4, we report our experiments on Peking University corpus, and in the final section we give our conclusions on this work.

2 MODELLING FOR INTEGRATED SEGMENTATION

This section describes a hybrid model to handle both internal word-formation features and external contextual information for the integrated word segmentation.

2.1 Modified word juncture models

As we have mentioned, there are two groups of word candidates in integrated segmentation: known word candidates and unknown word candidates. For convenience, a known word is also called as lexicon word in that it is always found in the system lexicon. Since any character can be an independent word in Chinese, and all Chinese characters are collected into our lexicon for word segmentation. Therefore, an unknown word can be made up of any lexicon words in theory. Thus, the integrated segmentation can be viewed as a process of assigning word juncture types to a sequence of known words.

Given a sequence of candidate words $W=w_1w_2\cdots w_n$, between each word pair $w_iw_{i+1}(1\leq i\leq n-1)$ is a *word juncture*, which has in general two different types in integrated word segmentation, namely *word boundary* (denoted by t_B) and *non-word boundary* (denoted by t_N). Let $t(w_iw_{i+1})$ denote certain type of a word juncture w_iw_{i+1}, and $P_r(t(w_iw_{i+1}))$ denote the relevant conditional probability, then

$$P_r(t(w_iw_{i+1}))\overset{def}{=}\frac{Count(t(w_iw_{i+1}))}{Count(w_iw_{i+1})} \qquad (1)$$

In a sense, *word juncture model* mirrors the affinity of a pair of lexicon words in forming another word, especially an unknown word. For a word juncture (w_i, w_{i+1}), the larger the probability $P_r(t_N(w_iw_{i+1}))$, the more likely these two words appear together in one word after segmentation.

Based on the definition in equation (1), the overall probability $P_{WJM-I}(w_C)$ of word junctures inside a word $w_C=e_ie_{i+1}\cdots e_j$ (where e_m is a component word of w_C, $i\leq m\leq j$) can be calculated by

$$P_{WJM-I}(w_C)=\begin{cases}\prod_{l=i}^{j-1}P_r(t_N(e_le_{l+1})), & \text{if is } w_C \text{ unknown}\\ 1, & \text{if } w_C \text{ is known}\end{cases} \qquad (2)$$

As for the overall probability $P_{WJM-O}(w_C)$ of word juncture outside w_C, i.e. the probability of the juncture between w_C and its previous word $w_P=e_ke_{k+1}\cdots e_l$, it can be formulated as

$$P_{WJM-O}(w_C)=\begin{cases} P_r(t_B(w_P w_C)), \text{if both } w_P \text{ and } w_C \text{ are known} \\ P_r(t_B(e_l e_i)), \text{ if both } w_P \text{ and } w_C \text{ are unknown} \\ P_r(t_B(e_l w_C)), \text{ if } w_P \text{ is unknown and } w_C \text{ is known} \\ P_r(t_B(w_P e_i)), \text{ if } w_P \text{ is known and } w_C \text{ is unknown} \end{cases} \tag{3}$$

2.2 Word-formation patterns

As we can see, a segmented word may be an independent lexicon word or a combination of lexicon words. In other words, a lexicon word presents itself as one independent word or one component word of another word after segmentation. More formally, a lexicon word w may take one of the following four patterns to present itself in segmentation: (1) w itself is a segmented word. (2) w is the beginning component of a word. (3) w is at the middle of a segmented word. (4) w appears at the end of a segmented word. For convenience, we use S , B , M and E to denote these four patterns respectively.

Let $pttn(w)$ denote a particular pattern of w in segmentation and $P_r(pttn(w))$ denote its relevant probability, then

$$P_r(pttn(w)) \overset{def}{=} \frac{Count(pttn(w))}{Count(w)} \tag{4}$$

Different to the definition in (Fu and Luke, 2003; Wang et al. 2000), here $Count(w)$ refers to the total frequency of the character string corresponding to the word w in the training corpus, which is counted from word lattice or from raw corpus of the relevant training data, and $Count(pttn(w))$ is the frequency of the word w given its certain pattern $pttn(w)$, which can be counted directly from a segmented corpus. In particular, if we encounter an unknown word in process of counting, we firstly segment it into a sequence of component words using the forward maximum match method, and then count the relevant pattern frequency for each component word. Due to this difference, some known words that contain other short known words are not considered in this work, so $\sum_{pttn} P_r(pttn(w)) = 1$ does not always hold here. As for the word-formation power of the known word w, it can be computed by Equation (5).

$$WFP(w) = B(w) + M(w) + E(w) \tag{5}$$

Let $P_{pttn}(w_C)$ be the overall word-formation pattern probability of a word candidate $w_C = e_i e_{i+1} \cdots e_j$, then

$$P_{pttn}(w_C)=\begin{cases} P_r(S(w_C)), \text{if } w_C \text{ is known} \\ P_r(B(e_i))P_r(E(e_j)) \prod_{k=i+1}^{j-1} P_r(M(e_k)), \text{if } w_C \text{ is unknown} \end{cases} \tag{6}$$

Theoretically speaking, a known word can take any pattern in forming an unknown word. But it is not even in probability for different known words and different patterns. For example, the word 性 (xing4, nature) is more likely to act as the suffix of words, while the character 阿 (a1) tends to appear at the beginning of words.

2.3 Hybrid models for integrated segmentation

It is proved that internal word-formation feature and external contextual information are equally important for high-quality word segmentation, especially for unknown word identification (Fu and Wang 1999; Wang et al. 2000; Wu and Jiang 2000; Zhang et al. 2002). In practice, above word-formation patterns and the internal word juncture probability reflect the affinity of components inside a word in segmentation, and in a way, the external word juncture model partly capture the roles of contextual words in segmentation. However, it is not sufficient to handle contextual information for integrated segmentation only by the external word juncture probability in that word juncture model cannot characterize the effects of Markov chain on segmentation. To make up this, word n-gram language models are also introduced into the integrated word segmentation. In view of the data sparseness, we only employ word bigrams in our work.

Given a sequence of Chinese character string $C = c_1 c_2 \cdots c_n$, there is usually more than one candidate segmentation $W = w_1 w_2 \cdots w_m$, which is made up of words that are known or unknown to the system dictionary. The integrated word segmentation aims to find the most appropriate segmentation $\hat{W} = w_1 w_2 \cdots w_m$ that maximizes

$$\begin{aligned} P_H(W) &= P_{pttn}(W) P_{WJM-I}(W) P_{WJM-O}(W) P_{bigram}(W) \\ &= \prod_i P_{pttn}(w_i) P_{WJM-I}(w_i) \prod_i P_{WJM-O}(w_i) P_{bigram}(w_i \mid w_{i-1}) \end{aligned} \tag{7}$$

Equation (7) gives a hybrid model for integrated word segmentation. Where, $P_H(W)$ denote the overall probability of a possible segmentation for a sentence and $P_{bigram}(w_i \mid w_{i-1})$ is the word bigram probability, which is calculated by Equation (8).

$$P_{bigram}(w_i \mid w_{i-1}) = \begin{cases} P_r(w_i \mid w_{i-1}), \text{if both } w_{i-1} \text{ and } w_i \text{ are known} \\ P_r(e_i \mid e_{i-1})), \text{ if both } w_{i-1} \text{ and } w_i \text{ are unknown} \\ P_r(e_i \mid w_{i-1})), \text{ if } w_{i-1} \text{ is known and } w_i \text{ is } \textit{un}\text{known} \\ P_r(w_i \mid e_{i-1}), \text{ if } w_{i-1} \text{ is } \textit{un}\text{known and } w_i \text{ is known} \end{cases} \tag{8}$$

Where, e_{i-1} and e_i denote the ending component word of w_{i-1} and the beginning component word of w_i respectively. If a large segmented corpus is available, the bigram probabilities can be easily estimated using the maximum likelihood estimation (MLE).

3 ALGORITHM FOR INTEGRATED SEGMENTATION

This section describes an algorithm for integrated segmentation. In particular, a filter algorithm is given in detail in this section to enhance the precision and avoid the problem of combination explosion in the construction of candidate words.

3.1 Viterbi segmentation

Given the model in section 2, the segmentation algorithm aims to find a best segmentation for an input sentence that has the maximum score shown in equation (7). In our system, we employ the classical Viterbi algorithm to perform this task. This algorithm has two main steps: (1) Word candidate construction: In this step, all eligible word candidates for the input sentence, including known word candidates and unknown word candidates, are built by looking up dictionary and using a filer algorithm shown in section 3.2, and finally stored in a word lattice. (2) Viterbi decoding: This step uses the hybrid model in equation (7) to score all possible segmentations, and then applies Viterbi algorithm to search an optimal path in above word lattice that has the maximum score. This optimal path contains the best word sequence for the input.

3.2 Filter algorithm

Word candidate construction is a challenge in integrated word segmentation. Unlike the two-stage strategy mentioned above, all possible word candidates, including known words and unknown words, are considered equally in integrated segmentation. Obviously, it is easy to build all known word candidates for a sentence only by looking up the lexicon used. But for the construction of unknown word candidates, it is a very difficult task. In fact, Chinese unknown words constitute an open set in that they are generally built by rather free word-formation rules. In theory, any combination of characters in the input may be an unknown word candidate. If we freely consider all these possible combinations as word candidates, the number may increase exponentially, which will finally results in combination explosion and rapidly decreasing in segmentation efficiency. Furthermore, building unknown word candidate without any restriction may yield a number of pseudo unknown words into the final result. Consequently, how to build effectively all eligible unknown word candidates for the input is crucial for integrated segmentation.

The filter algorithm attempts to prevent some potential combinations of characters in the input from becoming an eligible word candidate, which have low likelihood in forming a word. Although an unknown word may be any combination of characters in Chinese, it does not mean that we can freely consider any character string as an eligible word candidate. In practice, some Chinese characters or character combinations are seldom or never used as parts of unknown words. As mentioned in (Nie, Hannan and Jin 1995), the so-called functional characters with low word-formation power, such as 的 (De3, of) and 了 (Liao3, already), are hardly found in unknown words. Nie et al.(1995) and Wu and Jiang (2000) use word-formation power to block such combinations in word candidate construction. However, their methods are character based. They only take into consideration the word-formation power of each component character in their work. In our opinion, whether a character string should be blocked from becoming an eligible word candidate depends not only on the word-formation power of each component itself but also on the internal juncture probabilities between these components. In particular, we employ the word-based word-formation power and the internal word juncture probability to capture these features and define three conditions for filtering these ineligible candidates in our algorithm.. Let $w = e_1 e_2 \cdots e_l$ denote a potential word candidate, then these conditions can be formally defined as followings:

(1) Word-length condition: If w is an eligible word candidate, then its length $|w| < T_L$. Where, T_L is a threshold for word-length. The value of T_L is determined in advance according to the system dictionary and the corpus for training.

(2) Word-formation power condition: If w is an eligible word candidate, then the minimum value of word-formation power of its component word must be greater than a threshold T_{WFP}, i.e. $\min(WFP(e_1), WFP(e_2), \cdots, WFP(e_l)) > T_{WFP}$.

(3) Word-juncture condition: If w is an eligible word candidate, then the minimum value of internal word juncture probability of its component word must be greater than a threshold T_{WJM-I}, i.e. $\min(P_{WJM-I}(e_1, e_2), \cdots, P_{WJM-I}(e_i, e_{i+1}), \cdots, P_{WJM-I}(e_{l-1}, e_l)) > T_{WJM-I}$.

As a group, these conditions serve as a sufficient condition for constructing unknown word candidates. In other words, if a character string is an eligible unknown word candidate, it must satisfy the three conditions simultaneously. In our implementation, the thresholds for word-formation power and internal word-juncture probability are empirically determined. The larger T_{WFP} and T_{WJM-I} generally results in higher precision and lower recall in unknown word identification.

4 EXPERIMENTS

In evaluating the effectiveness of our approach, we conduct an experiment on Peking University corpus. This section reports the results and discussions on this experiment.

4.1 Setting of the experiments

4.1.1 Measures

In our experiments, three measures, i.e. *recall* (R), *precision* (P) and the *balanced F-measure* (F), are used to evaluate the performance of our system, which are defined in equation (9), (10), (11) respectively.

$$\text{R}(\%) = \frac{\#\,\text{correctly segmented words}}{\#\,\text{words in test}} \times 100\% \tag{9}$$

$$\text{P}(\%) = \frac{\#\,\text{correctly segmented words}}{\#\,\text{segmented words}} \times 100\% \tag{10}$$

$$\text{F}(\%) = \frac{\text{Recall} \times \text{Precision} \times 2}{\text{Recall} + \text{Precision}} \times 100\% \tag{11}$$

Note that a word in the automatic segmentation is correct if and only if it matches exactly the related word in the segmentation by hand.

4.1.2 Lexicon and corpora

The lexicon used in our experiment contains about 65, 269 words in all. The experimental corpus consists of about 1,001,061 words, which is collected from *the People's Daily*, and has been segmented and tagged with part-of-speech by Peking University. As shown in Table 1, 90% of this corpus, namely about 910,247 words are used as training data, and the rest 10%, namely 90,814 words are used for the open-test. The relevant unknown-word rates are 6.55% and 6.24% respectively.

	# words	# unknown words	Unknown word rate
Training corpus	910,247	59,616	6.55%
Testing corpus	90,814	5,667	6.24%
Total	1,001,061	65,283	6.52%

Table 1: Experimental corpora

4.2 Results and discussions

Methods	F	R	P	F_{KW}	R_{KW}	P_{KW}	F_{UW}	R_{UW}	P_{UW}	T (s)
M1	96.1	96.9	95.4	97.1	98.2	96.0	81. 2	77.4	85.5	38.67
M2	95.5	96.4	94.6	96.4	97.8	95.1	80.4	75.5	86.1	16.70
M3	93.7	93.8	93.6	95.5	95.7	95.2	66.8	65.5	68.1	22.57
M4	92.2	94.7	89.8	94.1	98.9	89.8	47.0	31.9	88.9	14.94
M5	91.9	94.2	89.8	93.9	98.3	89.8	46.8	31.8	88.6	14.11
M6	90.7	92.7	88.8	92.6	96.8	88.8	46.8	31.8	88.8	10.49

Table 2: Results for different segmentation methods

In addition to our integrated segmentation (denoted by M1), other baseline methods are also introduced into our experiment for comparison, including the two-stage segmentation incorporating word-based word-formation patterns, word juncture models and word bigram and (Fu and Luke 2003, denoted by M2), the two-stage segmentation incorporating character-based word-formation patterns, character juncture models and word bigram (Wang et al. 2000, denoted by M3), the word bigram based segmentation (denoted by M4), the maximum word frequency based segmentation (denoted by M5) and the forward maximum match based segmentation (denoted by M6). Furthermore, we compute following measures in our experiments, i.e. the *overall F-measure* (F), the *overall recall*(R), the *overall precision* (P), the *F-measure on known words* (F_{KW}), the *recall on known words* (R_{KW}), the *precision on known words* (P_{KW}), the *F-measure on unknown words* (F_{UW}), the *recall on unknown words* (R_{UW}), the *precision on unknown words* (P_{UW}) and processing time (T). We hope these measures can give a complete and objective evaluation on these approaches. What is more, we also hope our experiments can answer how much contribution different strategies and models make to the performance in segmentation.

The results of this experiment are presented in Table 2. Note that only the first three methods have unknown-word identification capability, the measures on unknown words for the other methods should be zero in theory. However, they are not in fact due to the reason of non-standard unknown words. In general, there are two groups of unknown words in Chinese texts, i.e. the standard unknown-words that are made up of pure Chinese characters and the non-standard unknown-words that contain non-Chinese characters such as numerals and alphabets. We uses one and the same rule-based module to cope with the non-standard words throughout our experiments.

From these results, we can draw some conclusions. Firstly, the integrated segmentation outperforms other methods on a whole. As can be seen in Table 2, the integrated method achieves the best performances in all other measures, except for the recall on known word and the process time. In comparison with the typical two-stage segmentation (viz. M2) based on the same models, the integrated strategy leads to improvement in overall F-measure by 0.6% and the *F-measure on* *unknown words* (F_{UW}) by 0.8%. Secondly, word-based models outperform character-based models. Although both M2 and M3 use the two-stage strategy to perform segmentation, M2 improves the segmentation accuracy (overall F-measure) by 1.8% from 93.7% to 95.5% and the unknown word accuracy (F-measure on unknown word) by 13.6% from 66.8% to 80.4%, compared with M3. This indicates that word-based word-formation models (viz. word-based word-formation patterns and word juncture models) are more powerful than character-based word-formation models (viz. character-based word-formation patterns and character juncture models), especially in unknown word identification. Thirdly, M4 achieves the highest score in the recall on known word segmentation, which shows word n-gram is powerful to capture the important contextual information for disambiguation. It also shows that there are still a number of pseudo unknown words identified wrongly by our system. Finally, although integrated segmentation yields the best results, further efforts are still needed to improve its efficiency.

5 CONCLUSIONS

This paper presents an integrated word segmentation algorithm for Chinese. Unlike most previous methods that take known word segmentation and unknown word identification as two independent stages, this algorithm performs disambiguation and unknown word identification simultaneously. In this work, a hybrid model is proposed to score known word candidates and unknown word candidates equally by incorporating the modified word-formation models (viz. word-juncture models and word-formation patterns) into word bigram models. The significance of this model is that it can capture different types of features for this integrated segmentation, including internal word-formation features and the external contextual information. To enhance the effectiveness and avoid the problem of combination explosion in word candidate construction, a filter algorithm is also given to block ineligible unknown word candidates. In this way, ambiguity and unknown word can be resolved effectively. The results of our experiment on Peking University corpus show that the integrated approach outperforms the other two-stage methods under discussion. In future work, we hope to improve our algorithm on its efficiency.

ACKNOWLEDGEMENTS

We would like to thank the Institute of Computational Linguistics, Peking University for providing the experimental corpora.

REFERENCES

Fu, Guohong and Xiaolong Wang. 1999. Unsupervised Chinese word segmentation and unknown word identification. In *Proceedings of NLPRS'99*, Beijing, China, 32-37.

Fu, Guohong, and K.K. Luke. 2003. A two-stage statistical word segmentation system. *Proceedings of The 2nd SIGHAN Workshop on Chinese Language Processing*, Sapporo, Japan, 156-159.

Hockenmaier, Julia, and Chris Brew. 1998. Error-driven learning of Chinese word segmentation. *Communications of COLIPS*, 1(1): 69-84.

Liang, Nan-Yuan and Yan-Bin Zheng. 1991. A Chinese word segmentation model and a Chinese word segmentation system PC-CWSS. *Communications of COLIPS*, 1(1): 51-55.

Nie, Jian-Yuan, M.-L. Hannan and W.-Y. Jin. 1995. Unknown word detection and segmentation of Chinese using statistical and heuristic knowledge. *Communication of COLIPS*, 5(1&2): 47-57.

Palmer, David D. 1997. A trainable rule-based algorithm for word segmentation. In *Proceedings of the 35th Annual Meeting of ACL and 8th Conferenc3e of the European Chapter of ACL*, Madrid, Spain, 321-328.

Teahan, W.J., Yingying Wen, Robert McNab, and Ian H. Witten. 2000. A compression-based algorithm for Chinese word segmentation. *Computational Linguistics*, 26(3): 375-393.

Wang, Xiaolong, Fu Guohong, Danial S.Yeung, James N.K.Liu, and Robert Luk. 2000. Models and algorithms of Chinese word segmentation. In *Proceedings of the International Conference on Artificial Intelligence (IC-AI'2000)*, Las Vegas, Nevada, USA, 1279-1284.

Wu, Andi, and Zixin Jiang. 1998. Word segmentation in sentence analysis. In *Proceedings of the 1998 International Conference on Chinese Information Processing*, 169-180.

Wu, Andi, and Zixin Jiang. 2000. Statistically-enhanced new word identification in a rule-based Chinese system. In *Proceedings of the Second Chinese Language Processing Workshop*, Hong Kong, 46-51.

Xue, Nianwen, and Susan P. Converse. 2002. Combining classifier for Chinese word segmentation. *The First SIGHAN Workshop on Chinese Language Processing*, Taiwan, 57-63.

Yao, Yuan. 1997. *Statistics based approaches towards Chinese language processing*. Ph.D. thesis, National University of Singapore.

Yeh, Ching-Long and Hsi-Jian Lee. 1991. Rule-based word identification for Mandarin Chinese sentences – A unification approach. *Computer Processing of Chinese & Oriental Languages*, 5(2): 97-117.

Zhang, Hua-Ping, Qun Liu, Hao Zhang, and Xue-Qi Cheng. 2002. Automatic recognition of Chinese unknown words based on roles tagging. *The First SIGHAN Workshop on Chinese Language Processing*, Taiwan, 71-77.