NAMED ENTITY RECOGNITION FROM SPOKEN DOCUMENTS USING GLOBAL EVIDENCES AND EXTERNAL KNOWLEDGE SOURCES WITH APPLICATIONS ON MANDARIN CHINESE

Yi-cheng Pan[†], Yu-ying Liu[‡], and Lin-shan Lee[†]

Graduate Institute of Computer Science and Information Engineering, National Taiwan University
Taipei, Taiwan, Republic of China

[†]{thomas,lsl}@speech.ee.ntu.edu.tw,[‡]tammy@realtek.com.tw

ABSTRACT

In this paper, we propose two efficient approaches for Named Entity recognition (NER) from spoken documents. The first approach used a very efficient data structure, the PAT trees, to extract global evidences from the whole spoken documents, to be used with the well-known local (internal and external) evidences popularly used by conventional approaches. The basic idea is that a Named Entity (NE) may not be easily recognized in certain contexts, but may become much more easily recognized when its repeated occurrences in all the different sentences in the same spoken document are considered jointly. This approach is equally useful for NER from text and spoken documents. The second approach is to try to recover some Named Entities (NEs) which are out-of-vocabulary (OOV) words and thus can't be obtained in the transcriptions. The basic idea is to use reliable and important words in the transcription to construct queries to retrieve relevant text documents from external knowledge sources (such as Internet). Matching the NEs obtained from these retrieved relevant text documents with some selected sections of the phone lattice of the spoken document can recover some NEs which are OOV words. The experiments were performed on Mandarin Chinese by incorporating these two approaches to a conventional hybrid statistic/rule-based NER system for Chinese language. Very significant performance improvements were obtained.

1. INTRODUCTION

Named Entities (NEs) such as person names, location names and organization names usually carry the core information of spoken documents, and are usually the key in understanding spoken documents. Therefore, Named Entity recognition (NER) has been the key technique in applications such as information retrieval, information extraction, question answering, and machine translation for spoken documents. In the last decades, substantial efforts have been made and impressive achievements have been obtained in the area of Named Entity recognition (NER) for text documents. Most of these works, either rule-based or statistics-based , rely on two major categories of evidences to extract the Named Entities (NEs) [1]. The first is the internal evidences within the Named Entities (NEs) such as the Capitalization of the characters, while the second is the external evidences such as the word co-occurrences observed from its context. However, these two categories of evidences are definitely not enough. For example, in the sentence "Even Language Technologies can't handle such problem", it is very difficult to recognize that "Language Technologies" instead of "Even Language Technologies" is an organization name with these two evidences alone. Considering the case of spoken documents, NER becomes more difficult since we can't simply rely on text NER techniques to be applied on the transcriptions of the spoken documents. The automatic speech recognition (ASR) outputs always have many speech recognition errors, and many NEs themselves are out-of-vocabulary (OOV) words and therefore can't be obtained in the transcriptions at the beginning.

In this paper, we propose two important new approaches to alleviate the above difficulties. The first is an efficient data structure, the PAT trees, to extract the global evidences from the whole spoken document to help the NER task. This approach is equally useful to text and spoken documents. The second is to retrieve relevant text documents from some external knowledge sources (such as the Internet) using queries constructed by reliable and important words in the transcription, and recover some NEs which are OOV words by matching with the NE candidates obtained from these relevant text documents. These approaches have been applied on Chinese broadcast news, and the initial experiments indicated that these approaches are very useful.

Below, in Section 2 we will briefly present the approach of using PAT trees to extract global evidences for NER. This approach can be applied to text documents or ASR outputs of spoken documents. In Section 3, we describe the approach of recovering OOV words by retrieving relevant text documents from some external knowledge sources.

0-7803-9479-8/05/$20.00 © 2005 IEEE

In Section 4, a baseline NER system for Chinese language is briefly summarized, on which the above two new approaches were integrated and with which all the experiments were performed. The experiments with Chinese broadcast news and discussions are then given in Section 5, with concluding remarks finally given in Section 6.

2. EXTRACTION AND ORGANIZATION OF GLOBAL EVIDENCES WITH PAT TREES

Many NEs may appear repeatedly many times in different sentences in the same documents. It may be difficult to recognize it from each single sentence individually, but if we can record such repetitive occurrences along with their different contexts in different sentences, we can consider the internal/external evidences for them in different sentences jointly. Such repetitive occurrences and the associated internal/external evidences contexts in different sentences in the same document are referred to as global evidences in this paper. Almost all NER approaches can be decomposed into two stages, NE candidate generation and NE verification and classification. The global evidences proposed here can be very helpful in NE candidate generation.

2.1. PAT tree

We need to efficiently organize and record the global evidences for a text document or the transcription of a spoken document. This is not trivial since there are a huge number of different segments of symbols (words or characters) in a document or a transcription. That's why we use the PAT trees to do this job. A PAT tree is an efficient data structure that has been successfully used in the area of information retrieval [2]. It is well known that the binary search tree is a very simple data structure for data search, but it requires more space and time because all the branch nodes and leaf nodes need to be constructed, stored, and traversed during search. As in the example in Figure 1, a complete tree is needed even if only 6 data are recorded. To reduce the space and time requirements, a comparison bit (CB) is usually used to indicate the bit differentiating the data to construct a compressed binary search tree as shown in Figure 2 (CB = 2 indicating the second bit is the comparison bit and so on), in which only necessary branch/leaf nodes are kept. The PAT tree, developed based on the PATRICIA algorithm [3], is conceptually equivalent to a compressed binary search tree, but further reduced using augmented branch nodes which can also serve as leaf nodes. As shown in Figure 3, the leaf nodes are actually absorbed by the branch nodes and only 6 nodes are needed for the 6 data in Figures 1 and 2.

When the PAT tree is used here for NER, a PAT tree can be constructed for each document, and the data recorded in the PAT tree are the symbol (e.g. character for Chinese or word for English) strings in the document, probably divided by punctuation marks such as "." and ",", together with all possible suffix strings. Also, CB in Figure 3 now plays the role as checking symbols (character or word) rather than bits. For example, consider a Chinese sentence or character string ". $\alpha\beta\gamma\nu$,$\omega\varepsilon$.", where each Greek character represents a Chinese character. It is first divided into two character strings, "$\alpha\beta\gamma\nu$" and "$\omega\varepsilon$", and then all possible different suffix strings, i.e., "$\alpha\beta\gamma\nu$", "$\beta\gamma\nu$", "$\gamma\nu$", "ν", "$\omega\varepsilon$", "ε" are recorded in the PAT tree. During the construction of the PAT tree, each distinct suffix string is represented as a node in the PAT tree, but the repeated suffix strings are recorded on the same node with frequency counted. In this way, not only all suffix strings are recorded, but also the repeated suffix strings are counted in the PAT tree. At the same time the number of branches or child nodes split from a node indicates the variety of the nodes right context symbols. As in the example shown in the left half of Figure 4, where A, B, C are three distinct right context symbols for the symbol segment "$\alpha\beta\gamma$" appearing in the document. So the PAT tree not only efficiently records all the symbol strings and possible suffix strings with minimum storage and fast search, but offer a means to count all the right contexts for a specific symbol segment. Similarly we can construct an inverse PAT tree in a reversed order as the right half of Figure 4, i.e.,

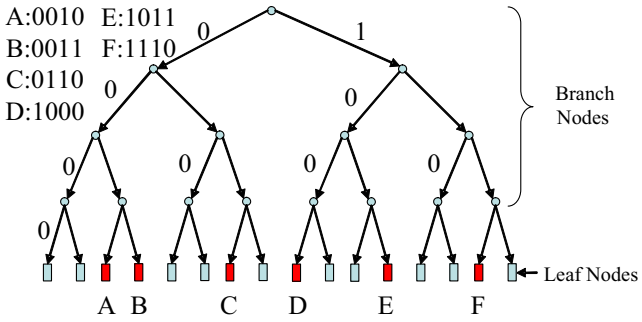

Fig. 1. An example of a complete binary search tree

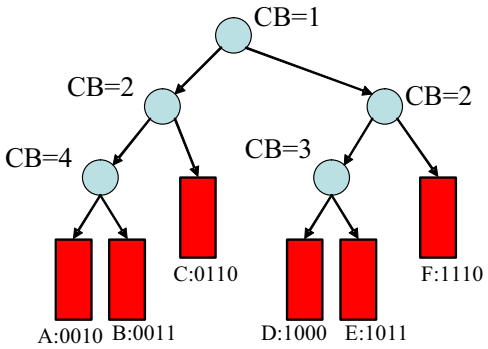

Fig. 2. An example of a compressed binary search tree

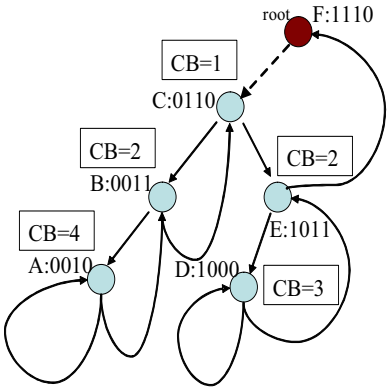

Fig. 3. An example of a PAT tree, recording 6 data:A, B, C, D, E, and F

including "$\nu\gamma\beta\alpha$", "$\gamma\beta\alpha$", "$\beta\alpha$", "α", "ε", "ω" as in the above example, so as to count the left contexts of a specific symbol segment, where D, E, F are the left context symbols for the symbol segment "$\alpha\beta\gamma$" appearing in the document.

2.2. NE candidate generation using global evidences

The process to generate NE candidates based on global evidences using Pat tree is easy. We first construct a PAT tree and an inverse PAT tree for the whole document. Each symbol segment s that has frequency count more than 2 will be first selected along with its number of child nodes for right/left context counts. But these selected symbol segments s are just segments of symbols, not necessarily represent complete entities. Therefore all these selected symbol segments s should be tested by the criteria as given in inequalities (1) and (2) below to make sure they represent a complete entity with a clear concept and correct boundaries,

$$|LC| > t_c \quad \text{and} \quad |RC| > t_c, \tag{1}$$

$$\max_{l \in LC} \frac{f(l,s)}{f(s)} < t_b \quad \text{and} \quad \max_{r \in RC} \frac{f(s,r)}{f(s)} < t_b, \tag{2}$$

where LC is the set of distinct left contexts of the symbol segment s and RC the set of distinct right contexts, $|\cdot|$ is the number of elements in the set, $f(s)$ is the frequency count of s, and $f(l,s)$, $f(s,r)$ are the frequency counts for the symbol segments s preceded by the left context l or followed by the right context r. The meaning of inequalities (1) and (2) is that in order for s to represent a complete entity with a clear concept and correct boundaries, it should have enough variety in its left and right contexts (inequality (1)), and its left and right contexts can't be dominated by a certain specific context (inequality (2)), i.e., $\max_{l \in LC} \frac{f(l,s)}{f(s)} < t_b$ means the counts for the most frequent left context l for s can't exceed a given ratio. The thresholds t_c and t_b can be empirically determined. Note that the tests in inequalities (1) and (2) can be very efficiently performed with the PAT tree. Those symbol segments satisfying these criteria are then the NE candidates extracted. In this way, the global evidences for NEs can be efficiently extracted and recorded by PAT trees.

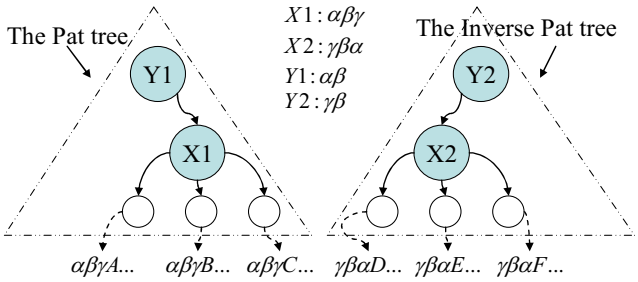

Fig. 4. An example of the PAT tree and the inverse PAT tree

2.3. Practical issues

Note that when a PAT tree is constructed for a document, all the n-gram statistics of the document are automatically included in the PAT tree. But the PAT tree also records much more context information about the document in addition to the n-gram statistics, in particular those required by inequalities (1) and (2). This is why PAT trees are proposed here as a special approach to NER. Also, PAT tree is specially efficient in using very limited space to record huge quantities of context information. But the time complexity to search through all the nodes in a PAT tree grows exponentially with the length of the document. A simple approach to this problem is to divide a long document into several parts, and construct sub-PAT trees for each part of it. In the experiments discussed below with broadcast news, all news stories are not long, and the search through the PAT trees can always be completed in real time. In fact, in real implementation it may not be necessary to check all the nodes in the PAT tree and the inverse PAT tree with inequalities (1) and (2). Some simple rules, such as a person name must include a surname or be followed by some cue words like "point out" or "say", are very helpful in deleting the impossible nodes and making the search through PAT trees very efficient.

Also, those NEs appearing only once in the whole document can't be extracted as NE candidates by the PAT tree. But the PAT tree doesn't hurt anyway and the correct recognition of such NEs is determined by the stage of NE verification and classification in any case. PAT tree helps as long as some NEs repeat themselves in a document with some difficult context as the example mentioned in Section 1.

3. CONFIDENCE MEASURES AND RECOVERY OF OOV WORDS

For spoken documents, the basic process of NER is to perform speech recognition first and then perform NER on the transcription obtained using NER approaches for text documents. But there are always recognition errors in the transcription. Also,many NEs themselves are out-of-vocabulary (OOV) words and thus can't be obtained in the transcription. Confidence measures are helpful to both issues.

3.1. Use of confidences measure to identify reliable words in the transcription

We first carefully evaluated the confidence measure $C(w)$ for each word w in the obtained word graph for the spoken document. Two levels of confidence measure are incorporated here. The lower level is from the *posterior word probability* computed for each word hypothesis w from the word graph [4]. We then clustered together all word hypothesis on the word graph corresponding to the same word w with close enough *begin-time* and *end-time* to construct a *sausage* [5] In the *sausage* the *posterior probabilities* for all entries merged into a single entry were summed up. The *posterior probabilities* are regarded as the lower level confidence measure. The higher level confidence measure, on the other hand, is based on the *probabilistic latent semantic analysis* (PLSA) using the knowledge about the words in the transcription [6] In this approach we evaluated the probability of observing each word w in the transcription given the transcription of the whole spoken document D, $p(w|D)$, and assign a score related to this probability. It was found that such high-level confidence measure really helped. We then linearly combined the two different levels of confidence measures into a single value $C(w)$. It was used to identify the reliable words in the transcription.

3.2. Extra NE candidates from external knowledge sources

The basic idea here is that the OOV NEs can't be obtained in the transcription, but may exist in some other relevant text documents, which can be retrieved from external knowledge sources (such as the INternet) using reliable and important words in the transcription of the spoken document. The NEs recognized from those text documents may then be good NE candidates for the spoken document. Therefore the first step is to construct good queries from the transcription of the spoken document to be used on some search engine, scuh as *Google*, to retrieve relevant text documents. There are several ways to construct such queries. A simple approach was to take every three adjacent words in the transcription of the spoken document and use them to construct disjunctive (***Or****ed*) queries. But queries obtained this way may bring many noisy text documents. A more efficient approach, however, was to find more reliable and important words to construct the conjunctive (***And****ed*) queries. In this approach, we traversed the word graph and selected the word sequences within the graph with component words having relatively higher confidence measure and relatively higher term frequency (*tf*) $\cdot$ inverse document frequency (*idf*) scores, and use them to construct the queries [7]. As an example shown in Figure 5, the solid edges of the word sequence ABC and the word D are two selected queries. Less number of more relevant text documents were retrieved with queries constructed in this way. NER processes including using global evidences with PAT trees were then performed on these documents to produce a set of NE candidates to be considered.

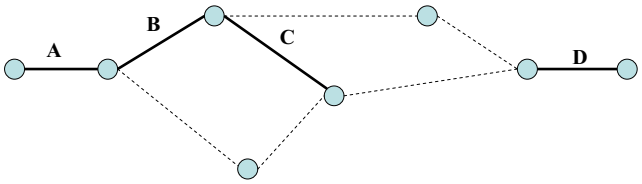

Fig. 5. Query construction based on the word graph by confidence measures and *tf·idf* scores

3.3. NE matching with phone lattices in the spoken documents

With NE candidates obtained from the relevant text documents as mentioned above, there are several ways to use them. The first is to add them to the lexicon, if they are not in the original lexicon, and then perform a second-phase transcription. Hopefully this time some NEs are not OOV any longer. A problem with this approach is these new NEs also need language model scores in the recognition process. Such language model scores can be estimated [8], but the results may not be very satisfactory. Here we used a different approach. The basic idea is that those word segments in the transcription with relatively lower confidence measure are likely to be recognition errors due to OOV NEs. So we can match the recognized phone lattices for these segments with the NE candidates obtained from the relevant text documents. If the similarity is higher than a threshold, we then include the matched NE as a new NE candidate for the spoken document to go through the standard NE verification/classification procedure. In order to perform the matching between two phone sequences, we define a phone similarity matrix, which is based on both the acoustic distance (from the *Mahalanobis Distance* between the HMM models of the phones) and the pronunciation distance (from the probability that a phone is likely to be pronounced as another phone in the pronunciation model [9]). In this matrix, every pair of phones has a distance between 0 and 1

between them. The phone sequence matching is then based on the total distance normalized with the number of phones in the sequences.

4. THE BASELINE NER SYSTEM FOR CHINESE TEXT DOCUMENTS

The above two new approaches were incorporated with a baseline NER system for Chinese text documents briefly presented here. It is a hybrid Chinese system, primarily based on a previously proposed statistics-based framework [10], but integrated with many carefully developed rules. In this system, we not only recognize NEs as person names (PN), location names (LOC), and organization names (ORG), but simultaneously solve the problem of Chinese word segmentation (because there are no blanks serving as word boundaries in Chinese texts).

Given a sequence os Chinese characters $S = s_1, ..., s_n$, there exist many possible word sequences $W = w_1, ..., w_m$ (each word is composed of one to several characters) with corresponding class sequences $C = c_1, ..., c_m$, (the classes here are PN, LOC, ORG and all other words in the vocabulary) and the purpose here is to find the best word sequence $W^* = w_1^*, ..., w_m^*$ and its corresponding class sequence $C^* = c_1^*, ..., c_m^*$ that maximize $P(C, W)$ as shown in equation 3

$$\begin{aligned}(C^*, W^*) &= \arg_{C,W} P(C, W) \\ &\cong \arg_{C,W} P(W, C)P(C). \end{aligned} \quad (3)$$

Here P(C) can be estimated by a class trigram language model according to equation (4),

$$P(C) \cong P(c_1) \cdot P(c_2|c_1) \prod_{i=3}^{m} P(c_i|c_{i-2}c_{i-1}), \quad (4)$$

and $P(W|C)$ can be approximated as given below,

$$P(W|C) = P(w_1 \dots w_m|c_1 \dots c_m) \cong \prod_{i=1}^{m} P(w_i|c_i). \quad (5)$$

The class trigram $P(c_i|c_{i-2}c_{i-1})$ and the probabilities $P(w_i|c_i)$ were trained using an NE labeled corpus.

In the hybrid approach here, we first generated all possible NE candidate classes to construct a word lattice and evaluate the corresponding probabilities $P(w_i|c_i)$. At this stage, quite many carefully developed rules were utilized. For example, for a Chinese person name candidate, we required that it included a surname predefined in a surname list, or has some person name related keywords in its context. We also incorporated the global evidences obtained with the PAT tree. For all character segments that fulfill the inequalities (1) and (2) as defined in Section 2.3 but not yet constructed as a NE candidate, we tag them as NE candidates with all possible class types PN, LOC, ORG. We finally assembled all candidates into a word lattice and performed the Viterbi search based on equation (3). The Viterbi search was multi-layered, because a Chinese NE can be a concatenation of several NEs.

5. EXPERIMENT RESULTS

We performed two sets of experiments for NER from text and spoken documents respectively in the evaluation. For the text documents, the standard Chinese test set of MET (Multilingual Entity Task)-2 defined by MUC-7 (Seventh Message Understanding Conference) was used, which includes 100 text documents. For the spoken documents, we used a total of 200 Chinese broadcast news recorded from radio stations at Taipei in 8 days in Sept. 2002 as the test corpus. NEs manually identified from these news stories, including 315 person names, 457 location names and 500 organization names, were taken as references. The extended knowledge sources to be retrieved for relevant documents are the Chinese text news available at "Yahoo! Kimo News Portal" [11] for the whole month of Sept. 2002, including about 38,800 text news stories. The evaluation metric we used was Precision and Recall rates plus F1 score, as defined by MUC-7. The results are listed in Table 1. For

Experiment Cases		NE	Recall	Precision	F1 score	Overall F1
Text Documents	(A)	PER	94	96	95.0	89.5
		LOC	89	93	91.0	
		ORG	87	96	91.3	
	(B)	PER	95	96	95.5	91.1
		LOC	94	92	93.0	
		ORG	89	95	91.9	
Spoken Documents	(C)	PER	71	86	77.8	77.6
		LOC	86	91	88.4	
		ORG	64	95	76.5	
	(D)	PER	73	85	78.5	80.0
		LOC	87	91	89.0	
		ORG	67	95	78.6	
	(E)	PER	76	87	81.1	80.9
		LOC	87	90	88.5	
		ORG	68	95	79.3	

Table 1. baseline in part (A), with global evidences by PAT trees in part (B) for text documents; and baseline in part (C), with global evidences in part (D) and OOV recovering in part (E) for spoken documents.

the text document, part (A) lists the results for the baseline Chinese NER system presented in Section 4, while part (B) are the results when the global evidences obtained in the PAT tree were used. Very significant improvements were

Character Accuracy for ASR	
Baseline	87.99%
After NER approach	88.26%

Table 2. Character Accuracy for ASR.

obtained (91.1 vs. 89.5) and the help from the global evidences was verified. Note that here the global evidences were used for NE candidate generation only, so very significant improvements in recall rates were obtained, but at the price of slight degradation in precision rates. Such degradation can actually be avoided if more deliberate NE verification techniques were applied, but left out here.

For the spoken documents, the results in part (C) are obtained with the baseline system presented in Section 4 directly performed on the transcriptions of the 200 news stories without global evidences considered, and the overall F1 score was 77.62. Part (D) are then the results after the global evidences recorded in the PAT tree were utilized. Very similar to the situation for the text documents, recall rates were significantly improved, resulting in a much higher overall F1 score of 80.01. The final results after filtering with confidence measures and the OOV recovery techniques as mentioned in Section 3 are listed in part (E) after careful selections of the three thresholds, including those for the confidence measure, for tf/idf score and for phone sequence matching. Comparing parts (D) and (E), both recall and precision can be improved by the OOV recovery scheme, with the overall F1 score achieving 80.93. In fact, the character accuracy for the 200 news stories was also improved from 87.99% (baseline system for part (C)) to 88.26% (part(E)). This may look marginal, but the point here is that the key information in NEs was extracted much better.

6. CONCLUSION

In this paper, we propose two new approaches for NER for spoken documents. In the first approach, the PAT tree was used to extract and organize the global evidences. Such evidences exist everywhere and are independent of languages and domains. The second approach tried to recover the OOV NEs in the spoken documents using external knowledge sources by finding NE candidates in retrieved relevant text documents. Preliminary experiments on Chinese broadcast news indicated that significant improvements were obtained.

7. REFERENCES

[1] D. McDonald, "Internal and external evidence in the identification and semantic categorization of proper names," in *Corpus Processing for Lexical Acquisition*. 1996, MIT Press. Cambridge, MA.

[2] L.-F. Chien, "Pat-tree-based keyword extraction for chinese information retrieval," in *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Developement in Information Retrieval*, 1997, pp. 50–58.

[3] G.H Gonnet, R.A. Baeza-Yates, and T. Snider, "New indices for text: Pat trees and pat arrays," in *Information Retrieval: Data Structure and Algorithms*. 1992, Prentice Hall, Inc.

[4] F. Wessel, R. Schlüter, K. Macherey, and H. Ney, "Confidence measures for large vocabulary continuous speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 9, pp. 288–298, 2001.

[5] L. Mangu, E. Brill, and A. Stocke, "Finding consensus among words: Lattice-based word error minimization," in *Proc. Eurospeech99*, 1999, pp. 495–498.

[6] S. Cox and S. Dasmahhapatra, "High-level approaches to confidence estimation in speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 7, pp. 460–471, 2002.

[7] P.-C. Chang and L.-S. Lee, "Improved language model adaptation using existing and derived external resources," in *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*, 2003, pp. 531–536.

[8] D. M. Bikel, R. schwartz, and R. M. Weischedel, "An algorithm that learns what's in a name," *Machine Learning (Special Issue on Natural Language Processing)*, vol. 34, no. 3, pp. 211–231, 1999.

[9] M.-Y Tsai and L.-S Lee, "Pronunciation variation analysis based on acoustic and phonemic distance measures with application examples on mandarin chinese," in *IEEE Wordkshop on Automatic Speech Recognition and Understanding*, 2003, pp. 117–122.

[10] J. Sun, M. Zhou, and J.-F. Guo, "A class-based language model approach to chinese named entity identification," *International Journal of Computaional linguistics and Chinese Language Processing*, vol. 8, no. 2, pp. 1–28, 2003.

[11] http://tw.news.yahoo.com/.