PAT-tree-based Language Modeling with Initial Application of Chinese Speech Recognition Output Verification

Chun-Liang Chen[1], Bo-Ren Bai[2], Lee-Feng Chien[3] and Lin-Shan Lee[1,2,3]
[1]Dept. of Computer Science and Information Engineering, National Taiwan University
[2]Dept. of Electrical Engineering, Rm 531, National Taiwan University
[3]Institute of Information Science, Academia Sinica
Taipei, Taiwan, R.O.C.
E-mail: {liang,white}@speech.ee.ntu.edu.tw, {lfchien, lsl}@iis.sinica.edu.tw

ABSTRACT

In spontaneous speech recognition, there are always inevitable errors in the output due to the difficulties of acoustic recognition or linguistic decoding. In this paper, we present an output verification approach to detect and correct the errors automatically using the abundant Internet resources. The Syllable PAT tree (SPAT tree), a metamorphic data structure derived from the PAT tree concept, is a real N-gram language model and is first used as a verifier for speech recognition output in order to improve the accuracy of speech recognition. The verification approaches proposed here not only reduce the character error rate by 12.66% in preliminary experiments, but can make the recognition results more reliable for the following-up processing, such as semantic analysis in dialog control or speech understanding.

1. INTRODUCTION

In conventional speech recognition systems, N-gram language models have been widely used to estimate the probabilities of possible sentence hypotheses and find the path with maximal probability as most promising output for the input utterance. However the adopted language models in the system are very often simplified approximations, e.g., bigram or trigram models, due to the considerations of memory space and computational complexity in practical implementation, and the resulted word error rate is therefore usually not very satisfactory. Chase [1] classified the errors made by a large-vocabulary speech recognizer into seven categories: (1) out-of-vocabulary (OOV) word spoken (2) search error (3) homophone substitution (4) language model overwhelming correct acoustics (5) transcript/pronunciation problems (6) confused acoustic models and (7) miscellaneous/not possible to categorize. An intuitive and example-based approach for error correction in domain-specific speech recognition was proposed by using features of character co-occurrence [2] in Japanese language, where a set of erroneous-correct utterance pairs are in advance collected in an error-pattern database and a similar pattern matching algorithm is used to retrieve the pairs for corrections. It reduces over 8% of the errors. Another different approach is the development of a noisy-channel model to correct word-level errors in English [3,4].

In this paper, we present a PAT-tree-based text verification approach to automatically detect and correct possible errors from Mandarin speech recognition results

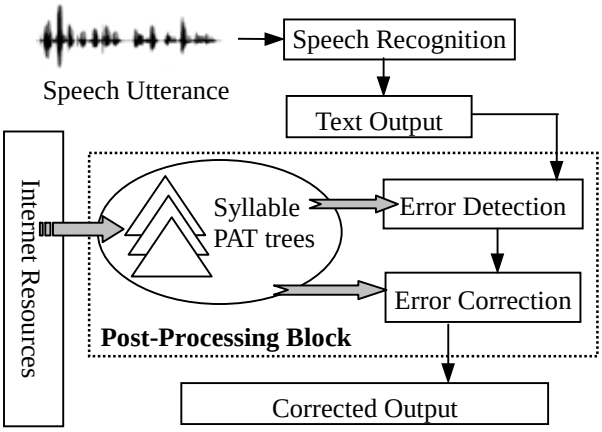

Fig.1: An abstract diagram showing the function of the proposed text verification approach for speech recognition

by means of a large scale corpora obtained from abundant Internet resources. An abstract diagram showing the processing of the proposed approach for speech recognition is depicted in Fig.1. In order to increase the accuracy of speech recognition based on the concept of text verification, an efficient working structure extended from original PAT tree, namely Syllable PAT tree (SPAT tree), is a syllable-character-pair N-gram language model [5]. Such a language model is served as an information base of a text verifier for post-processing of speech recognition, including the verification, detection and auto-correction of possible recognition errors. Actually, the role of the text verifier is very similar to conventional spelling checkers for western language document processing, except that it is designed to be able to perform sentence rather than word-level verification, and contain more rigid linguistic knowledge, such as that used for long-distance dependency estimation between words [6] and resolution of homonym characters [7]. At the same time, the constructed syllable-character-pair N-gram language model, as observed in experiments, owns more effective linguistic knowledge than conventional N-gram language models for the processing of text verification. The proposed approach is, in fact, especially useful for Chinese and some other Asian languages, in which there are no explicit word boundaries in the text and no commonly-accepted lexicon established.

2. SYLLABLE PAT TREE

PAT tree [8] is an efficient data structure

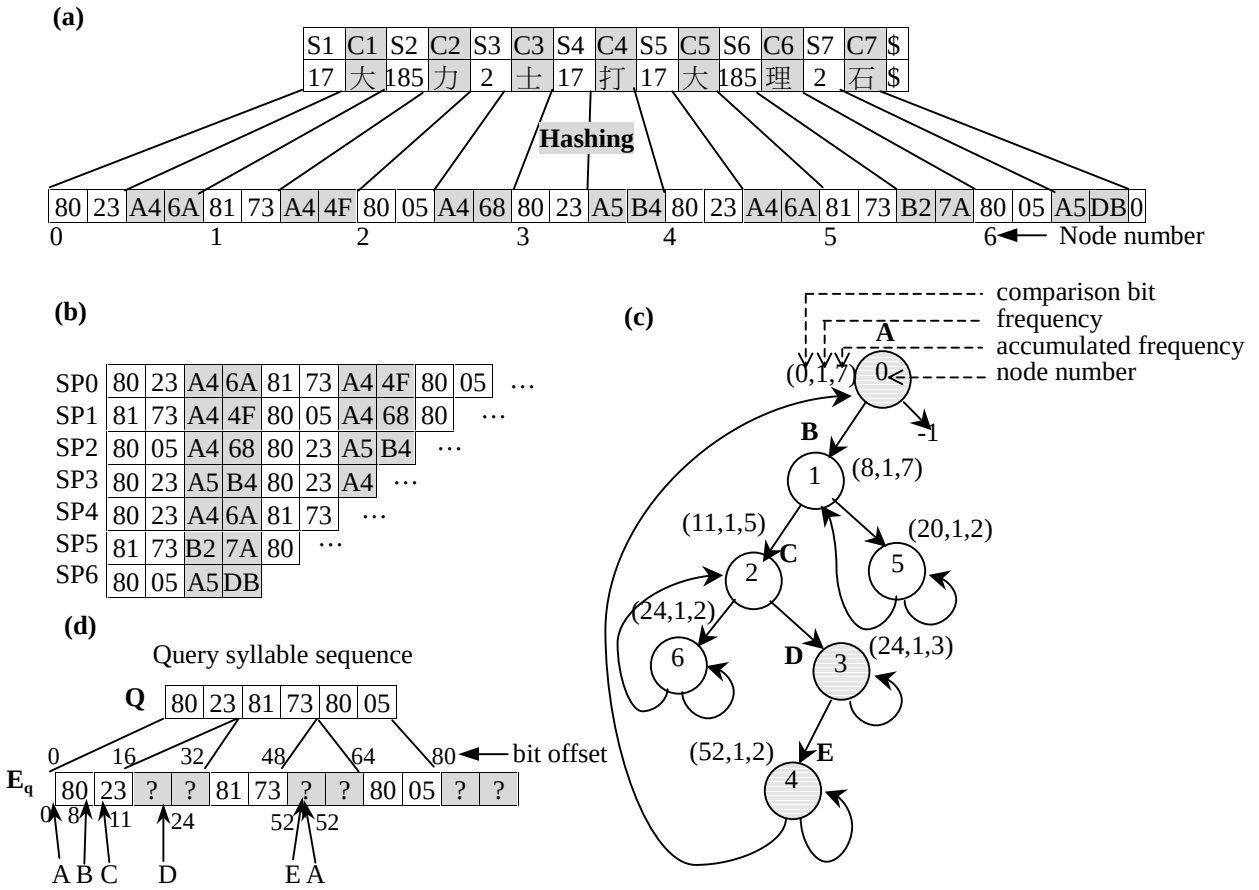

Fig. 2: SPAT tree data structure

successfully used in the area of information retrieval. Using this data structure for indexing full-text context of a large corpus, all possible segments of character strings including their frequencies in the corpus can be retrieved and updated in a very efficient way, but not every such segment needs to be stored. This makes the PAT tree especially useful in constructing higher-order language models to include huge amount of linguistic knowledge regarding all possible segments of character strings in a large corpus, which will be much more powerful than the conventional N-gram models if N is only 2 or 3.

SPAT tree, derived from original PAT tree, inherits both the same intrinsic structure and powerful properties in N-gram indexing and retrieval, except in replace of the character with a syllable-character pair as the basic indexing unit. The PAT tree has been extended to construct a text verifier for Chinese OCR-ed documents [9]. However, it is found not effective in verifying documents input by commercial speech or phonetic input methods because of a language model almost embedded in these input methods. For this reason, the design of SPAT tree is tried to efficiently retrieve all the homonym character strings for any given Mandarin syllable sequence. A simple example for SPAT tree construction and access is given below for demonstration:

C1~C7: 大 力 士 打 大 理 石
S1~ S7: <Da Li Shi Da Da Li Shi>
Hercules (大力士) beats the marble.

Assume the character string (C1~C7) is to be constructed as a SPAT tree. First of all, each character in the string will be paired with a corresponding base syllable (S1~S7) respectively. An encoding scheme is then adopted to make the syllable-character string as a bit stream by hashing each syllable into a unique two-byte code and each Chinese character into its corresponding BIG5 code (Fig.2(a)). For each syllable-character pair in the encoded bit stream its beginning position will be pointed by a certain suffix node to represent an occurrence of a particular suffix pattern (SP) as shown in Fig.2(b). The construction process for SPAT tree is similar to that of original PAT tree, except that the indexing unit is changed as the syllable-character pair. This indexing scheme makes it possible to find out all the homonym character strings in a SPAT tree for any given base syllable sequence. Fig.2(c) shows the whole tree structure, in which each node represents a unique suffix pattern and associated with a quadruple of information including “comparison bit”, “frequency”, “accumulated frequency” and “node number”. The “comparison bit” is used to indicate the bit number needs to compare and decide the left or right branch to go when traversing at this node. The “frequency count” is the number of total frequencies of the indexed suffix pattern occurring in the SPAT tree while “accumulated frequency” stands for the sum of frequency counts of total nodes in the sub-trees. At last, the node number records the starting position of the indexed point in the data stream.

As for searching all the homonym character strings for a base-syllable query sequence <Da Li Shi> for

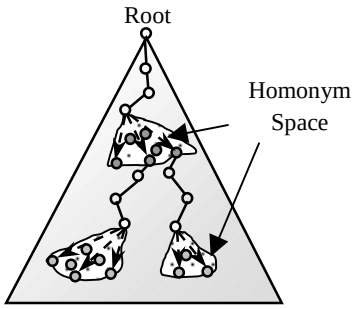

Fig. 3 : An abstract diagram showing possible conceptually searching space in SPAT tree

example, the query sequence Q is assigned a bit stream with question marks inserted between each base syllable as the gray blocks in Fig. 2(d). The purpose of the searching can be, thus, figured as that to find out all of replaceable characters in the SPAT tree to fill in these question marks. As shown in Fig.2(c), the searching process will start from node B instead of root A because A is a dummy node of the tree. If the comparison bit of the query syllable sequence is 0 then it will choose the left branch to go and otherwise right branch. Repeat going in this way until it bumps the node with the comparison bit indicating to question-mark regions of E_q (Fig.2(d)). In this example, it will stop at node D for a while since the comparison bit of D is 24, which is between 16 and 32. The node D is conceptually the root of the homonym space (Fig. 3). All of its decedent nodes in which have a sub-sequence of syllables in common but associated with different character strings. For clearness, Fig.3 shows the whole traverse can be divided into two alternating stages. The object of the first stage is to find the next root of homonym space, in which all the nodes with comparison bit within character region (gray cells) in E_q are included. The comparison bits of the nodes traversed in first stage are sure to be within the syllable region (white cells) in E_q. At stage two, the object is to produce all the possible homonyms by exhaustive search in homonym space and it must keep track of some nodes at the fringe of this space so as to use them as the seed nodes to start in next loop of stage one. Stage one and two alternates until it reaches at the end of E_q or all the homonyms are checked. Return to the example, nodes D,E and A are nodes in homonym space and we check each of them by the suffix patterns pointed by node number on each node. Finally we find that SP0 and SP4 pointed separately by node E and A are the answers to the Q. By SP0 and SP4 we retrieve <大力士> and <大理石> as the corresponding homonyms of <Da Li Shi>. Furthermore, its right context can also be extracted by the way of just checking the parts of byte offset larger than 12 in SP0 and SP4.

3. THE PROPOSED TEXT ERIFICATION APPROACH

3.1 Procedures for verification

The proposed text verifier is designed based on the above SPAT tree with other useful techniques. The goal and procedures to perform the verification process are defined and introduced below.

3.1.1 Goal

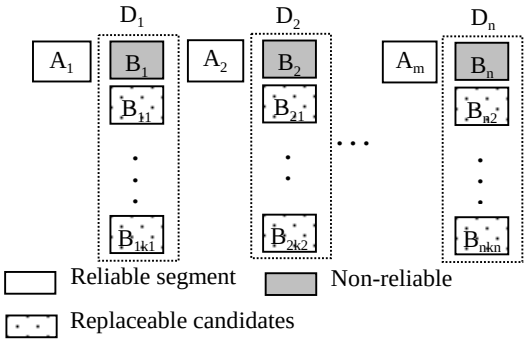

Fig. 4 : The segmentation for error detection and correction

Given a character string S resulted from speech recognition and a SPAT tree T_s, the goal of the verification process is to re-estimate the reliability score **R**(S) to see if there exists another S* in T_s, which is phonologically similar to S and has the maximum **R**(S*) to replace S as the output.

3.1.2 Steps

(1) **Generating N-gram Spectrum**

The first step in the processing is to generate a N-gram spectrum for each input S. Assume S have N characters, then there are N*(N-1)/2 variable length sub-patterns in S. For each of the patterns with the length greater than 4 (an empirical number), it will be transformed from character to corresponding base syllable sequence and used for retrieving all possible homonym character strings in SPAT tree. All of the obtained homonym character strings conceptually constitute a N-gram spectrum as that shown in Fig.4. For reducing the computation complexity, it will remove the strings occurring only once in the SPAT tree, and that without any overlapped bi-gram or tri-grams in the spectrum.

(2) Reliable and Non-reliable Segments Detection

For each character in S, starting from left to right, the second step is to check in the N-gram spectrum to see if there are more reliable characters to replace. If it is yes, the character will be marked. Then merging all of the marked characters in nearby positions it will form a bigger segment. Moreover, the process will further divide S into two groups: reliable class A (consists of all unmarked segments) and non-reliable class B (consists of all marked segments), supposing there are m reliable segments in A and n non-reliable segments in B. Besides, the non-reliable segments which are replaceable to the same sub-string in S will also belong to class D (Fig.4).

(3) Error Detection and Correction

The non-reliable segments produced in step 2 are prone to be the error segments. To judge the reliability of each non-reliable segment, there are two contextual association measurement functions f_1 and f_2 presented and defined below.

Let $A=\{A_1,A_2,\ldots A_m\}$, $B=\{B_1,B_2,\ldots,B_n\}$, $D=\{D_1,D_2,\ldots D_n\}$, and $D_i=\{B_i,B_{i1},\ldots,B_{iki}\}$, where $B_{i1},\ldots,B_{iki}$ are replaceable candidates for B_i found in step 2. Also, let the context of any segment X be $C(X)=LC(X)\cup RC(X)$, where $LC(X)$ and $RC(X)$ each stands for the left context and right context of X respectively. Note that the context information can be easily retrieved

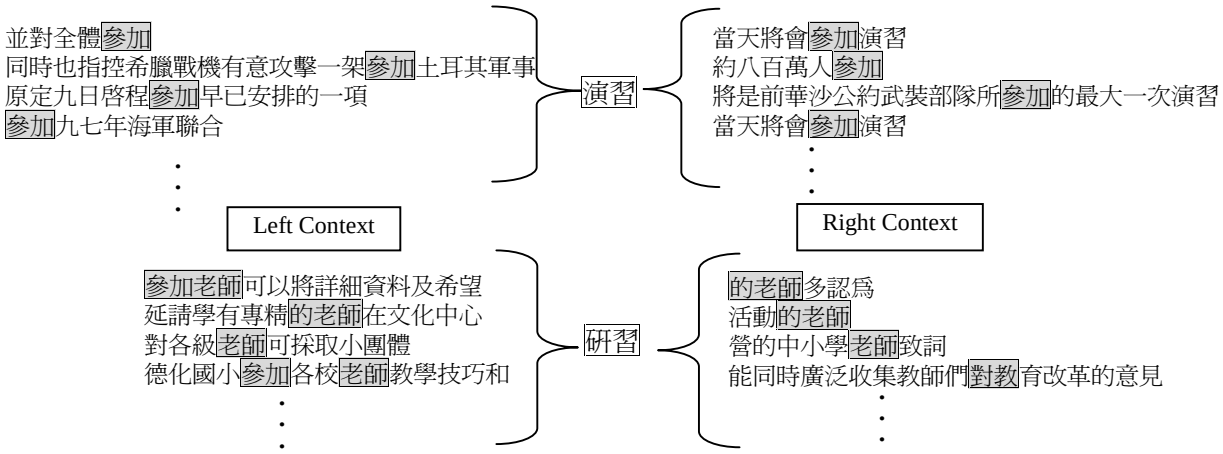

Fig. 6 : The context of "演習" and "研習"

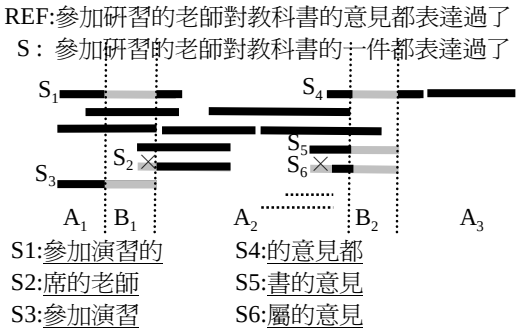

Fig. 5 : N-gram spectrum of hypothesis output S

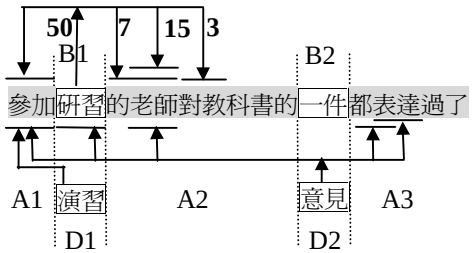

Fig. 7 : The competition of contextual association

in SPAT tree. Besides, let $P_x \in A$, $P_y \in D_i$ and P_z be a sub-pattern in P_x. The definitions of the contextual association measurement function f_1 and f_2 are as follows:

$$\begin{cases} f_1(P_x, P_y) = 1 \text{ if there exists a } P_z \in C(P_y) \\ f_1(P_x, P_y) = 0 \text{ if no } P_z \in C(P_y) \end{cases}$$

$$f_2(P_x, P_y) = \sum_{\forall Pz \in Px} \frac{N(P_z \subset C(P_y))}{2N(P_y)}$$

, where $N(X)$ denotes the number of frequency of X .With the above functions, the procedures for error detection and correction are listed as follows:

Repeat

{

find $B'_1 = \arg\max_{Py \in D1} \left(\sum_{1 \le j \le m} f_1(A_j, P_y) \right)$, calculate the first order measure function for all the candidates in D_1 and choose the candidate with the highest scores as B'_1.

if ($B'_1 = B_1$), we say B_1 passed the verification and now the non-reliable segment B_1 is sure to become reliable.

else

{

if (B'_1 has only one element), replace B_1 with B'_1.

else

$B''_1 = \arg\max_{Py \in B'1} \left(\sum_{1 \le j \le m} f_2(A_j, P_y) \right)$, Replace B_1 with B''_1

}

Merge B_1 with its right-side and left-side reliable segments to form a new reliable segment A_1.

m:=m-1; n:=n-1; and re-indexing class A, B, D

} **Until** only one reliable segment S* left.

3.2 Example

An example is illustrated here to clarify the details of the verification process. Given a speech recognition output S, in step one, it will convert S into base syllables and a number of homonym character strings found in SPAT tree to form a N-gram spectrum. In Fig. 5, the black lines stand for the extracted character segments with the same characters to those in S, while the gray lines contrarily mean different characters (Fig. 5). All noisy segments have been deleted, like S_2 and S_6. Then, in step 2, it is also shown in Fig. 4 that there are three reliable segments (A_1,A_2,A_3) and two non-reliable segments (B_1,B_2) obtained. Two patterns "演習" and "意見" are then formed the replaceable candidate for B_1 and B_2 respectively. At step 3, the reliability of the replaceable candidates is examined according to the contextual association measurement function f_1 and f_2. If the obtained reliability is more robust than that in non-reliable segment, a replacement operation will occur. Some of the contextual information of the replaceable candidate "演習" and "研習", which are obtained in our experimental SPAT tree, are illustrated for comparison in Fig. 6. The value of function f_1 is calculated by the numbers of distinct sub-patterns of the reliable segments appearing in the context of a certain replaceable candidate. Here only the sub-patterns with length over 1 are taken into calculation. In Fig. 7, we can clearly see that "演習" has only one such a distinct sub-pattern "參加" in the first reliable segment while "研習" has four. The replaceable candidate "演習" cannot replace "研習" because of its weak contextual association. On the contrary, "意見" is more reliable than "一件" here hence it's an error detected and the replacement will occur. Using f_1 to measure the reliability of replaceable

candidates in D may be not robust and satisfactory enough. If f_1 fails to, we use f_2 to precisely estimate the statistical information in long distance contextual dependency. Take for an example, the numbers near the link in Fig. 7 are assumed to be the number of co-occurrences of contextual association. The f_2 value of A_2 and B_1 is calculated as follows:

$$
\begin{aligned}
& f_2(A_2, B_1) \\
&= f_2(\text{的老師對教科書的}, \text{研習}) \\
&= \frac{N(\text{的老師} \in C(\text{研習}))}{N(\text{研習})} + \frac{N(\text{老師} \in C(\text{研習}))}{N(\text{研習})} + \frac{N(\text{對教} \in C(\text{研習}))}{N(\text{研習})} \\
&= \frac{7}{2 \times 50} + \frac{15}{2 \times 50} + \frac{3}{2 \times 50} = \frac{25}{100} = \frac{1}{4}
\end{aligned}
$$

4. EXPERIMENTAL RESULTS

The 1998 Golden Mandarin speech recognizer [10,11] in NTU speech lab is used for our preliminary speech output verification experiments. Ten testing documents, which are arbitrarily selected from the editorials of newspapers in 1997, were input by microphone by one speaker with Speaker Independent (SI) model. This testing set includes 469 Chinese sentence and a total of 5,394 Chinese characters. The recognition accuracy for each document is listed in column 5 of Table 1 and the obtained average character accuracy is 85.61%.

The SPAT trees were constructed by CNA (Central News Agency) on-line news with size ranging from 23MB to 107MB. The obtained statistics listed in columns “Error Detection” and “Error Correction” are the detailed results of each document using 107MB corpus to construct SPAT tree. As the corpus size increasing, the obtained precision rates for error detection can be from 70.44% to 78.76% and recall rates drop from 27.18% to 22.53% (Table 2). When measuring the system performance by the average precision and recall rates, APR=(Precision+Recall)/2, it can be found that the APR value of error detection can be improved from 48.81% to 50.65% and that of error correction improved from 43.94% to 44.21%. In addition, from Table 3, it also can be observed that 12.66% errors can be corrected automatically by the system.

5. DISCUSSION AND CONCLUSION

All the errors corrected by proposed approach are analyzed and classified into 3 reason categories: (1) homonyms resolution (2) tone tolerence (3) OOV determination. Catogory 1 has two sub-categories, (1a) longer N-gram constraints and (1b) contextual dependency. Some of the examples are shown in Table 4. In comparison with the western alphabetic orthograpy, the Chinese character is an ideograph, which not only directly combines sounds and meanings into one character but also carries the specific meaning when collocated with neighboring characters. The SPAT tree reflects the truth of advantages of using longer N-grams and contextual dependency to recover errors. By constructing SPAT tree with base syllable sequences, we release the constraints of correct syllables with wrongly recognized tones to include more replaceable candidates for ambiguation resolutions. Besides, SPAT tree also provides another solution for OOV determination with the help of abundant Internet resources.

The quality of verification of speech recognition output highly depends on the quality of speech recognition output. The circumstances of insertions, deletions and continuous wrong syllables are not discussed here because it’s more complicated for error detection, let alone correction. The syllable PAT tree shows its effectiveness in verifying Mandarin speech recognition output. The proposed text verification approach contains two additional features. The first one is that the performed verification process is speech-recognizer independent. It can be easily extended to verify texts input by conventional phonetic input methods. And, the second is the proposed approach is easily to utilize abundant resources from Internet. In our planning, it is expected that the proposed Syllable PAT tree language models can be tightly integrated with speech recognition system to achieve more accurate recognition results. The primary problem to be dealt with is the compression of SPAT tree.

REFERENCES

[1] Chase Lin, "Blame Assignment for Errors Made by Large Vocabulary Speech Recognizer", EuroSpeech’97, Vol. 2, pp. 815 - 818,1997.

[2] S. Kaki, E. Sumita and H. Iida, "A Method for Correcting Errors in Speech Recognition Using the Statistical Features of Character Co-occurrence", COLING’98, pp.653-657, 1998.

[3] E.K. Ringger and J.F. Allen, "Robust Error Correction of Continuous Speech Recognition", Proceedings of the ESCA-NATO Workshop on Robust speech Recognition for Unknown Communication Channels, 1997.

[4] E.K. Ringger and J.F. Allen, "A Fertility Channel Model for Post-Correction of Continuous Speech Recognition", ICSLP'96, Vol.2 pp.524-527, 1996.

[5] L.F Chien et al., "Internet Chinese Information Retrieval Using Unconstrained Mandarin Speech Queries Based on A Client-Server Architecture and A PAT-tree-based Language Model", ICASSP’97,Vol. 2, pp. 1155-1158, 1997.

[6] George Demetriou et al., "Large Scale Lexical Semantics for Speech Recognition Support", EuroSpeech’97, Vol. 5, pp. 2755 - 2758,1997.

[7] A.F. Lochovsky and K.H. Chung, "Homonym Resolution for Chinese Phonetic Input", Communications of COLIPS, Vol. 7, No. 1, pp.5-15, JUN 1997.

[8] Gaston H. Gonnet, Ricardo A. Baeza-yates and Tim Snider, "New Indices for Text: PAT Trees and PAT Arrays", Information Retrieval Data Structures & Algorithms, Prentice Hall, NY, pp. 66-82, 1992.

[9] C.L. Chen, B.R. Bai, L.F Chien and L.S. Lee, "CPAT-Tree-Based Language Models with an Application for Text Verification in Chinese", ROCLING XI, pp. 189-203, 1998.

[10] Lin-shan Lee, "Voice Dictation of Mandarin Chinese", IEEE Signal Processing Magazine, Vol.14, No.4, pp.63-101, July 1997.

[11] T.H. Ho, K.C. Yang, K.H. Huang, L.S. Lee, "Improved Search Strategy for large Vocabulary Continuous Mandarin Speech Recognition", ICASSP’98, pp.825-828, 1998.

Doc #	Char Num (A)	Gm4SI Model			Error Detection				Error Correction			
		Correct (B)	Error (C)	Accuracy (D=B/A)	Correct (E)	False Alarm (F)	Recall (G=E/C)	Precision (H=E/(E+F))	Correct (I)	False Alarm (J)	Recall (K=I/C)	Precision (L=I/(I+J))
860606-1	601	515	86	85.69%	24	5	27.91%	82.76%	8	3	9.30%	72.73%
860609	529	395	134	74.67%	31	10	23.13%	75.61%	15	5	11.19%	75.00%
860611	664	574	90	86.45%	17	4	18.89%	80.95%	9	3	10.00%	75.00%
860612	756	673	83	89.02%	9	2	10.84%	81.82%	6	2	7.23%	75.00%
860613	1133	952	181	84.02%	34	9	18.78%	79.07%	22	6	12.15%	78.57%
860620	373	325	48	87.13%	10	3	20.83%	76.92%	6	2	12.50%	75.00%
860908-1	319	267	52	83.70%	16	7	30.77%	69.57%	7	3	13.46%	70.00%
860908-2	403	374	29	92.80%	14	3	48.28%	82.35%	11	4	37.93%	73.33%
860908-3	260	217	43	83.46%	13	3	30.23%	81.25%	8	2	18.60%	80.00%
860921	453	409	44	90.29%	10	2	22.73%	83.33%	8	2	18.18%	80.00%
Avg/Total	5491	4701	790	85.61%	178	48	22.53%	78.76%	100	32	12.66%	75.76%

Table 1: The obtained results for the 10 testing documents with the SPAT trees trained by 107MB corpus

Error Detection					
Corpus size	23M	44M	66M	88M	107M
Recall	27.18%	26.80%	25.12%	23.77%	22.53%
Precision	70.44%	73.01%	75.22%	76.89%	78.76%
APR	48.81%	49.91%	50.17%	50.33%	50.65%

Table 2: The obtained recall and precision values for error detection

Error Correction					
Corpus size	23M	44M	66M	88M	107M
Recall	9.40%	10.82%	11.55%	12.14%	12.66%
Precision	77.57%	76.70%	76.12%	75.90%	75.76%
APR	43.49%	43.76%	43.84%	44.02%	44.21%

Table 3: The obtained recall and precision values for error correction

(1) Homonym Resolution	
(1a): Longer N-gram Constrains	(1b): Contextual Dependency
(YES)李登輝總統已應邀出席這項會議 (NO) 李登輝總統以應邀出席這項會議	(YES)推薦表中也沒有政黨和省籍這項資料 (NO) 推薦表中也沒有政黨和省及這項資料
(YES)李總統的同步翻譯機出了點小狀況 (NO) 李總統的同步翻譯擊出二點小狀況	(YES)政治人物應該接受公評 (NO) 政治人物應該接受公平
(YES)主張修憲者歡欣鼓舞之餘 (NO) 主張修憲的歡心股五十餘	(YES)權力的分配 (NO) 全力的分配
(2) Tone Tolerence	(3) OOV Determination
(YES)刻意營造立法行政兩院的氣氛 (NO) 可以營造立法行政兩院的氣氛	(YES)李總統的孫女李坤儀 (NO) 李總統的孫女李坤一
(YES)立法院有倒閣權 (NO) 立法院有道格權	(YES)蕭內閣的議題 (NO) 小內閣的議題
(YES)並不只是出於私人的感情而已 (NO) 並不只是處於私人的感情而已	(YES)台中市籍省議員盧秀燕在省政質詢中 (NO) 台中市籍省議員盧秀岩在省政質詢中

Table 4: Classification for the errors remedied