Design of Vocabulary-Independent Mandarin Keyword Spotters

Chi-Min Liu, Chin-Chih Chiu, and Hung-Yuan Chang

***Abstract*—This paper considers the design of a vocabulary-independent keyword spotter for Mandarin speech according to the framework by Huang *et al.* [1]. This paper considers three varieties of filler model structures for the framework based on subsyllabic grammar of Mandarin speech. On the basis of the three structures, we infer the problems of this framework through three arguments and presents two methods to modify the spotting vehicle.**

***Index Terms*—Keyword spotting, speech recognition.**

I. Introduction

The objective of keyword spotters is to spot keywords embedded in extraneous speech. Since keyword spotting techniques can provide user-friendly interfaces for speech recognition systems, the research on spotting techniques is receiving wide attentions in recent years [1]–[7]. Among these techniques, the hidden Markov modeling (HMM) is most widely adopted. A keyword spotting system in general consists of two processing modules: the keyword spotter and the keyword verification. The keyword spotter spots the most likely words from an utterance while the keyword verification verifies whether or not the spotted word can be accepted. This paper considers the design of keyword spotters based on the HMM modeling.

Designing suitable filler models to represent the extraneous speech is the critical issue in keyword spotting techniques. After establishing filler models, speech recognition techniques are then applied to determine the sequence of keyword and filler models that represent speech utterances. Therefore the recognition technique and the design of filler models are two fundamental topics for keyword spotting. The dimension of the above two topics depends strongly on the vocabulary size of keywords and the allowable extraneous speech. This paper develops keyword-spotting techniques based on the assumption that the contents and size of keywords and extraneous speech are not limited. We test various vocabulary size ranging from 500 to 25 000 and assume all the Mandarin speech other than keywords are the extraneous speech.

Keyword spotters based on the speaker-independent Mandarin polysyllabic word recognition system [9], [10] are studied in this paper. This Mandarin recognition system modeled polysyllabic words through subsyllabic units. Those units were trained from 74 speakers with ages ranging from 20 to 40. The tree-trellis search algorithm [8], which has been considered an efficient search algorithm for large vocabulary Mandarin recognition, was applied to efficiently search for the most probable word. Our previous experiments have demonstrated that the time spent on searching slightly increased when vocabulary size grew from 500 to 25 000. A framework as exhibited in Fig. 1 has been developed by Huang *et al.* [1] to extend tree-trellis efficient search algorithm for keyword spotting. This framework models the extraneous speech via the filler models positioned in the front and tail of keywords. Since the phoneme units of a language can constitute the filler units, that framework fulfills the assumption that all Mandarin speech other than keywords can be accounted to be extraneous speech. This paper discusses the design of the spotting system based on the framework.

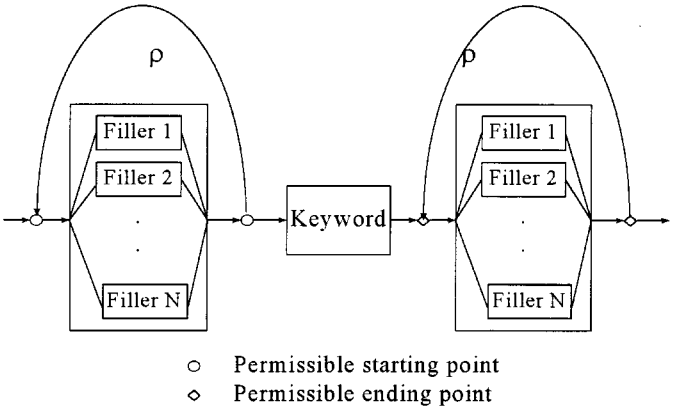

Fig. 1. Framework of the spotting system.

Section II presents three structures of fillers which are modeled based on the subsyllabic grammar of Mandarin speech. On the basis of the three filler structures, we infer the problems of this framework through three arguments. The first argument challenges the precision of the modeling units. The traditional concept that better precise modeling units leads to better spotting rate is not always true from the experiments on the framework in Fig. 1. The second argument sets up the factor which degrades the merits of the modeling precision. The third argument demonstrates the tradeoff between the degradation and merits from the modeling precision. To avoid the degradation in the second argument, we introduce an antitrust factor to improve the performance in Section III. Furthermore, Section IV presents an inhomogeneous modeling method to keep a good balance for the tradeoff in the third argument. The above two schemes are tested through extensive experiments to demonstrate that the recognition system based on the framework by Huang *et al.* can be enhanced individually by 5.8%, 9.3%, and 8.7% for 500-, 5000-, and 25 000-words systems. For comparing and setting up the design rule of filler modeling, Section V gives concluding remarks.

II. Mandarin Spotting Systems

A. Baseline System

Mandarin speech is a syllabic language, where each character is pronounced as a spoken syllable. Each word is composed of several connected characters and is pronounced continuously through the associated syllables. In the tree-trellis algorithm [2], all polysyllabic words are arranged in a tree structure, where each arc is associated with one syllable. The algorithm consists of two main processes: the forward time-synchronous heuristic scoring and the backward time-asynchronous A* searching. In the forward process, the Viterbi decoding with looser grammar constraints was applied to prepare the heuristic score for each node of the lexicon tree at each time instant. The backward process utilized these heuristic scores to find the N-best candidates sequentially along the lexicon tree.

Fig. 2 illustrates the network diagram of forward heuristic scoring process, where the string length is chosen to be three syllables for illustration. In Fig. 2, the filler network is located in front of the keyword syllable network. All the keywords are sorted according to the syllable sequences and the sequence are sorted with orders and put into different levels of the network. Fig. 3 illustrates the network diagram of the backward A* process. Following the method in [1], the tree-trellis

Manuscript received December 31, 1997; revised July 14, 1999. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Richard C. Rose.

The authors are with the Department and Institute of Computer Science and Information Engineering, National Chiao-Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: cmliu@csie.nctu.edu.tw).

Publisher Item Identifier S 1063-6676(00)05177-4.

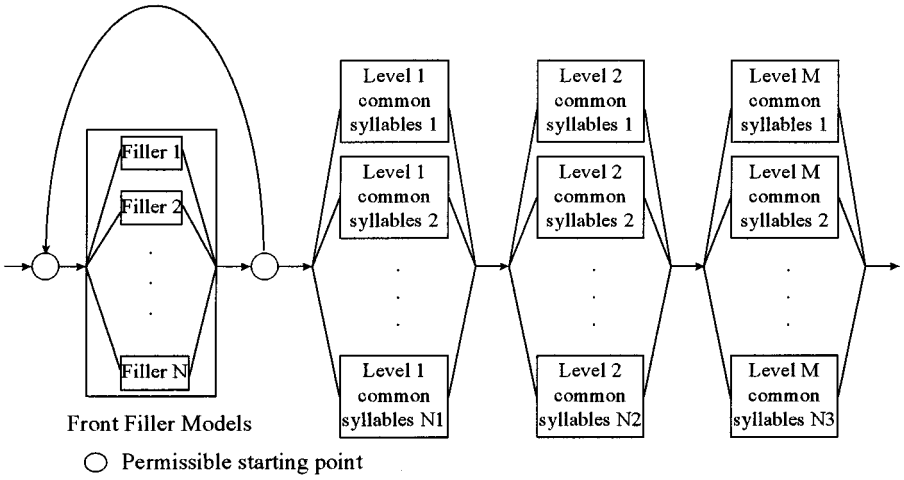

Fig. 2. Network of forward heuristic scoring process, where the syllable length is assumed to be three for illustration.

search algorithm can then be applied to achieve the efficient search for keyword spotting.

B. Three Filler Structures and Experiments

We examine three filler structures according to the characteristic of Mandarin speech mentioned in the previous section. The first structure is a syllabic network where each filler model in Fig. 1 was a syllable. Mandarin speech consists of 408 syllables; hence, 408 syllable models have been adopted for the first filler model. The second one is an INITIAL/FINAL (I/F) structure where each filler unit is either an INITIAL or a FINAL. Each Mandarin syllable is composed of one INITIAL concatenated by one FINAL, where each INITIAL is uttered as consonants and each FINAL consists of at least one vowel. These INITIAL's and FINAL's are generally modeled and trained as either context-dependent (CI) or right-context-independent (RCD) models. We have adopted 27 INITIAL models and 38 FINAL models for CI case, and 109 INITIAL models and 38 FINAL models for RCD one. Among the 27 INITIAL's, six INITIAL's are trained for those syllables without INITIAL's. The last filler structure is an INITIAL-FINAL (I- F)-constrained network where each INITIAL must be sandwiched between two FINAL's and each FINAL is required to be parenthesized into two INITIAL's as illustrated in Fig. 4.

The baseline system was trained through the use of 7400 utterances spoken by 74 male speakers with ages ranging from 20 to 40. Each utterance was composed of one to several syllabic units. It implies that the utterance could be a monosyllabic word, a polysyllabic word, a phrase, or a sentence. The training method is the standard forward-backward method usually used for hidden Markov models.

Testing data were collected from five male speakers. Each speaker offered 200 different utterances containing a single keyword embedded in extraneous speech. The extraneous speech has a syllable length ranging from zero to five. The tested keywords contain 500, 5000, and 25 000 Chinese names, respectively.

Two types of HMM models, CI (context independent) and RCD (right context dependent) ones mentioned above were employed in the experiments. Each model contained five states, and each state contained four mixtures. All speech data were sampled at 16 kHz and pre-emphasized using a first order filter with a coefficient of 0.95. The frame size was 320 digitized samples that were windowed by a Hamming window with shifted length 160 samples. Twelve cepstrum coefficients were derived from a mel-frequency analysis with 18 filter banks. A feature vector composed of the 12 cepstrum and 12 delta cepstrum was utilized to represent the features of each frame.

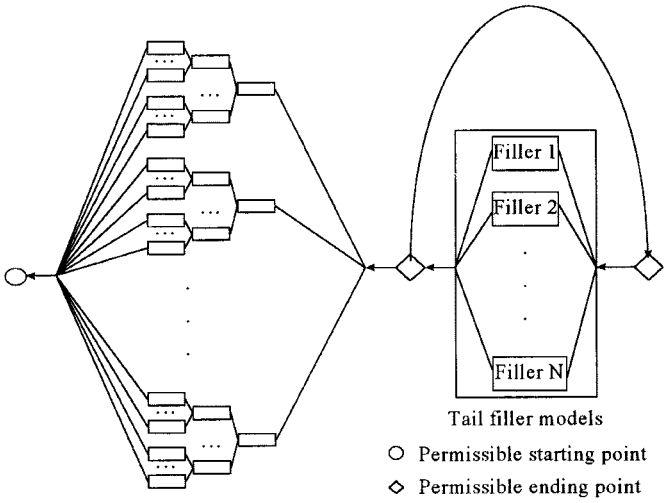

Fig. 3. Network of backward A* search process, where the syllable length is assumed to be three for illustration.

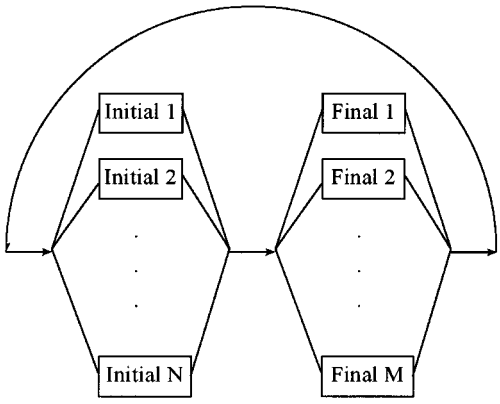

Fig. 4. Network of the I–F-constrained filler models.

Table I shows the recognition rates for the three filler structures based on either CI or RCD models. The TOP5 column shows the inclusion of five best recognition results in counting the recognition rate.

C. Remarks

From the experiments, we discovered the following phenomenon.

Argument 1: The traditional concepts that more precise modeling units lead to higher recognition rate in speech recognition systems is not always applicable to the framework in Fig. 1.

Considering I/F and I-F-constrained fillers for 500 word vocabulary in Table I, we found that the fillers based on RCD model units do not always offer better recognition rates than CI models. But both our previous experiments on the same HMM models for speaker independent recognition system [9] and those in other researches [11] illustrated that the recognition systems based on the RCD units provided higher recognition rate than that based on the CI units.

Taking the modeling units and the structure of fillers into account may offer a reasonable interpretation for this phenomenon. All of the above three filler structures try to model all Mandarin speech as the extraneous speech. In Mandarin speech, there are 27 INITIAL's and 38 FINAL's. But there are only 408 syllables instead of $(27 \times 38 = 1026)$ ones. For fillers networks constituted by INITIAL/FINAL (I/F) structure and INITIAL–FINAL(I–F)-constrained structure, these networks represent 1026 syllables instead of the 408 syllables in Mandarin. All the paths of the network associated with the syllables other than the 408 syllables are the illegal paths. When we utilize RCD models under I/F or I–F-constrained filler structure, those illegal acoustic paths will reduce the spotting rate.

The advantage of the approach in Fig. 1 is that any Mandarin speech can be modeled through the fillers and hence there is no need to have the priori knowledge on the extraneous speech. However, these filler structures have the following problem.

Argument 2: The fillers based on phoneme models provide too many acoustic paths for extraneous speech such that the speech frames are unsuitably segmented or trapped to fillers.

This mis-segmentation degrades the recognition rate of keywords and may provide the reason why the I/F and I–F-constrained fillers based on RCD's have a worse performance. This mis-segmentation as well indicates the following tradeoff.

Argument 3: There is a tradeoff between the modeling accuracy and the acoustic paths that the fillers can provide.

RCD units model Mandarin speech through a larger number of units than CI units. Meanwhile, the large number of model units also indicates the larger acoustic paths or a higher possibility for mis-segmentation. However, this tradeoff also suggests that the recognition rate can be improved if undesirable segmentation can be controlled.

To describe the effects of mis-segmentation, we concatenate the filler models which represent the extraneous speech of each test utterance. Then the testing is performed again to acquire the recognition rate. In other word, the testing assumes that the extraneous speech is determinate and well modeled. The results provide the optimum recognition rate for the filler design in Fig. 1. The recognition results are displayed in Table II. We could conclude by comparing Tables I and II that the optimum system provides about 9%, 20%, and 22% improvement potentials for 500-, 5000-, and 25 000-words systems respectively. These numbers are utilized as the technical reference for developing new techniques. The next two sections present two solutions to control the mis-segmentation.

TABLE I
RECOGNITION RATE (%) OF THREE FILLER MODELS, WHERE SY, IF, AND IFC DENOTE SYLLABIC, I/F, AND I–F-CONSTRAINED FILLER MODELS

	TOP 1			TOP 5		
Vocabulary	**500**	**5k**	**25k**	**500**	**5k**	**25k**
Sy-CI	84.9	63.5	46.0	94.3	80.8	66.7
Sy-RCD	86.2	68.8	52.8	94.3	82.7	71.2
I/F-CI	80.5	56.2	39.1	94.3	78.7	63.7
I/F-RCD	78.7	56.7	41.6	90.2	80.0	66.4
IFC-CI	83.1	59.9	44.0	92.8	76.1	59.9
IFC-RCD	83.5	62.1	46.7	91.3	76.5	59.9

TABLE II
THE RECOGNITION SYSTEM WITH KNOWN EXTRANEOUS SPEECH

	TOP 1			TOP 5		
Vocabulary	**500**	**5k**	**25k**	**500**	**5k**	**25k**
CI	95.4	85.8	69.6	98.0	95.3	88.1
RCD	95.5	88.4	74.8	98.0	95.7	90.0

III. ANTITRUST FACTOR FOR FILLERS

If the fillers span a large space for extraneous speech as mentioned in ***Argument 2,*** the illegal acoustic paths should somehow be restricted or reduced to improve the recognition rate. We introduce an antitrust factor to reduce the trapping effects. To illustrate the method, we can consider Viterbi search of the recognition system. Let F be the set of filler, and F^i denotes the set of filler sequences constructed by concatenating i filler units from F. The closure of F, denoted F^* is the set $F^* \equiv \bigcup_{i=0}^{\infty} F^i$. Given an utterance sequence $(O_1 \sim O_T)$ with frame length T, the keyword spotter finds the most likely keyword by the following rule:

$$\lambda' = \arg \max_{\lambda} \max_{s_1^T} P(O_1^T, s_1^T | F^* \lambda F^*). \tag{1}$$

In this paper, an observation sequence $(O_j \sim O_k)$ is denoted as O_j^k, and the associated state sequence in the HMM network in Fig. 1 is denoted as s_j^k. The joint probability can be represented as the multiplication of three probability functions

$$P(O_1^T, s_1^T | F^* \lambda F^*) = \max_{0 \le t_1 \le t_2 \le T} P(O_1^{t_1}, s_1^{t_1} | F^*) \cdot P(O_{t_1+1}^{t_2} s_{t_1+1}^{t_2} | \lambda) P(O_{t_2+1}^T, s_{t_2+1}^T | F^*). \tag{2}$$

For (2), ***Argument 2*** implies that F^* provides a large number of illegal acoustic paths and most frames of an utterance sequence will be segmented as fillers to have high probability values from the two probability functions $P(O_1^{t_1}, s_1^{t_1} | F^*)$ and $P(O_{t_2+1}^T, s_{t_2+1}^T | F^*)$. Hence, most speech frames are easily trapped by filler models. To overcome the drawback, we introduce the antitrust factor γ into (2)

$$P'(O_1^T, s_1^T | F^* \lambda F^*) = \max_{0 \le t_1 \le t_2 \le T} \left[P\left(O_1^{t_1}, s_1^{t_1} | F^*\right) \right]^{\gamma} P \cdot \left(O_{t_1+1}^{t_2}, s_{t_1+1}^{t_2} | \lambda\right) \cdot \left[P\left(O_{t_2+1}^T, s_{t_2+1}^T | F^*\right) \right]^{\gamma} \tag{3}$$

where γ is positive and is greater than 1. Since the density function is mostly less than 1, the factor will reduce the weights of the two likelihood functions $P\left(O_1^{t_1}, s_1^{t_1} | F^*\right)$ and $P\left(O_{t_2+1}^T, s_{t_2+1}^T | F^*\right)$ in Viterbi search process. Furthermore, since we have implemented Viterbi search through logarithmic scoring and because

$$\log P'(O_1^T, s_1^T | F^* \lambda F^*) = \gamma \log P\left(O_1^{t_1}, s_1^{t_1} | F^*\right) + \log P\left(O_{t_1+1}^{t_2}, s_{t_1+1}^{t_2} | \lambda\right) + \gamma \log P\left(O_{t_2+1}^T, s_{t_2+1}^T | F^*\right) \tag{4}$$

factor γ can be easily included in Viterbi search of the A* algorithm illustrated in Section II with little overhead. For implementation, the logarithmic function has been always realized by memory look-up tables to speed up the computation time. In the situation, the logarithmic function and the multiplication factor γ can be jointly implemented by memory-look-up tables without overhead. Boulard *et al.* [6] have incorporated a weighting factor for the transitions between filler models and those between fillers and keywords. A garbage transition penalty was introduced to inhibit the frames trapping from fillers and give score compensation for keyword models. To compare with the penalty method, we derive the associated joint probability. As indicated in Fig. 1, the penalty method gives a penalty score $\rho < 1$ in the transition between fillers. The joint probability is then

$$\begin{aligned} P'(O_1^T,\, s_1^T|F^*\lambda F^*) = & \max_{0\le t_1\le t_2\le T}\left[P\left(O_1^{t_1},\, s_1^{t_1}|F^*\right)\rho^{i_1-1}\right] \\ & \cdot P\left(O_{t_1+1}^{t_2},\, s_{t_1+1}^{t_2}|\lambda\right) \\ & \cdot\left[P\left(O_{t_2+1}^{T},\, s_{t_2+1}^{T}|F^*\right)\rho^{i_2-1}\right] \end{aligned} \tag{5}$$

where i_1 and i_2 are the number of filler units associated with the path $s_1^{t_1}$ and $s_{t_2+1}^T$. The probability in (5) can be computed through the logarithmic form as follows:

$$\begin{aligned} \log\, P'(O_1^T,\, s_1^T|F^*\lambda F^*) = & \log\, P\left(O_1^{t_1},\, s_1^{t_1}|F^*\right) \\ & + \log\, P\left(O_{t_1+1}^{t_2},\, s_{t_1+1}^{t_2}|\lambda\right) \\ & + \log\, P\left(O_{t_2+1}^{T},\, s_{t_2+1}^{T}|F^*\right) \\ & + (i_1+i_2-2)\log\,\rho. \end{aligned} \tag{6}$$

Comparing (6) and (4), we know that the penalty method avoids the trapping effects through the bias related with the number of filler units $(i_1 + i_2 - 2)$ while the antitrust method adjusts each probability through the multiplication term γ. The trapping effects come from the two probabilities $P\left(O_1^{t_1}s_1^{t_1}|F^*\right)$ and $P\left(O_{t_2+1}^T,\, s_{t_2+1}^T|F^*\right)$. Since the two joint probabilities are multiplication of the probabilities from individual observation frame O_i, i.e., $P\left(O_1^{t_1},\, s_1^{t_1}|F^*\right) = \prod_{k=1}^{t_1} P(O_k|s_k,\, F^*)P(s_k|s_{k-1},\, F^*)$ and $P\left(O_{t_2}^T,\, s_{t_2}^T|F^*\right) = \prod_{k=t_2}^{T} P(O_k|s_k,\, F^*)P(s_k|s_{k-1},\, F^*)$. The trapping effects relate with the two factors: the number of the multiplication terms, t_1 and $(T - t_2)$, and the two probability terms: $P(O_k|s_k,\, F^*)P(s_k|s_{k-1},\, F^*)$ and $P(O_k|s_k,\, F^*)P(s_k|s_{k-1},\, F^*)$. Although the inclusion of the bias in (5) can ease the trapping effects, the concept that a larger number of filler units leads to a higher penalty is in general not true. It is known that a large number of frames may correspond to just one filler unit (or phoneme unit) due to different utterance manner. From the two probability terms $P(O_k|s_k,\, F^*)P(s_k|s_{k-1},\, F^*)$ and $P(O_k|s_k,\, F^*)P(s_k|s_{k-1},\, F^*)$, the large number of frames implies the higher trapping effect. Although we can increase the value of penalty factor ρ to cover the trapping, the reliance on the number of utterance frames leads to value uncertainty and some vehicles need to handle the problem. For comparison, the formula (4) associated with antitrust method is rewritten as

$$\begin{aligned} \log\, P'(O_1^T,\, s_1^T|F^*\lambda F^*) = & \log\, P\left(O_1^{t_1},\, s_1^{t_1}|F^*\right) \\ & + \log\, P\left(O_{t_1+1}^{t_2},\, s_{t_1+1}^{t_2}|\lambda\right) \\ & + \log\, P\left(O_{t_2+1}^{T},\, s_{t_2+1}^{T}|F^*\right) \\ & + (\gamma-1)\log\, P\left(O_1^{t_1},\, s_1^{t_1}|F^*\right) \\ & + (\gamma-1)\log\, P\left(O_{t_2+1}^{T},\, s_{t_2+1}^{T}|F^*\right). \end{aligned} \tag{7}$$

Obviously, the two methods are equivalent when

$$\begin{aligned} & (\gamma-1)\log\, P\left(O_1^{t_1},\, s_1^{t_1}|F^*\right) + (\gamma-1)\log\, P \\ & \quad\cdot\left(O_{t_2+1}^{T},\, s_{t_2+1}^{T}|F^*\right) \\ & = (i_1+i_2-2)\log\,\rho. \end{aligned} \tag{8}$$

That is, for every value of γ, there will be a corresponding value of ρ covering the trapping effects. The problem is the value determination of factor ρ. For the antitrust method, the last term of (7) can naturally consider the number of trapping frames through a factor γ. Also from (3), the method with the factor in exponents can give different punishing extent for different probability values. Since a higher likelihood probability of a frame provides higher fidelity, the probability should be punished less to reflect the fidelity that is the feature of the exponent weights. Following the above discussion, the antitrust factor provides an ease to control the trapping phenomenon with consideration to frame length, probability fidelity, and simple complexity. Table III illustrates the experimental results with various values of antitrust factor. We can give the remarks for the experiment results as follows.

- All the results illustrate improvement over those in Table II, which indirectly certificates ***Argument 2***.
- The experiments also demonstrate that the value of the antitrust factor can not be increased without limit because a large value of the factor violates the role of filler for modeling extraneous speech. In other words, a large value leads to an over-segmentation of speech frames for keywords, which leads to recognition degradation although can avoid the trapping effects from fillers.
- The value of the antitrust factor depends on the filler space and keyword space. The factor can be suitably trained or tuned if the extraneous speech and keyword of applications are determined.
- Comparing the performance of the fillers based on CI units with that on RCD units, we found again that RCD fillers do not have a better performance than the correspondent CI fillers under I-F-constrained and I/F filler structures, which also hints that the tradeoff mentioned in ***Argument 3*** has not been excluded through the antitrust factor.
- The TOP 5 inclusion rate in Table III is greater than the TOP 1 inclusion rate in Table II, which indicates that the method has also greatly enhanced the potential of the system. Next section will further improve the system based on the technique.

IV. Inhomogeneous Modeling Units for Keywords and Fillers

The antitrust factor is considered from ***Argument 2*** and has been proved to have improved the recognition rate of the baseline system. As mentioned in ***Argument 3***, there is a tradeoff between the modeling accuracy and the acoustic paths the fillers can provide. We can speculate that the tradeoff comes from the need for the modeling accuracy of keyword and the extraordinarily large paths of fillers. To give a good balance between the two factors, this section considers an inhomogeneous modeling method, which modeled keywords through RCD units and fillers through the CI units.

Table IV illustrates the experiment results combining the inhomogeneous method and the antitrust factor. The experiment results are summarized as follows:

- By combining the inhomogeneous modeling units with the antitrust factors, we can enhance the recognition rate by 3%–9% depending on the vocabulary size. The results also illustrate that the inhomogeneous modeling unit can control well the tradeoff mentioned in ***Argument 3***.

TABLE III
RECOGNITION RATE (%) OF THE THREE FILLER MODELS WITH VARIOUS VALUES OF ANTITRUST FACTOR γ

	TOP 1			TOP 5		
Vocabulary	**500**	**5k**	**25k**	**500**	**5k**	**25k**
Sy-CI (γ=1.05)	87.2	67.8	50.9	96.0	84.7	56.4
Sy-CI (γ=1.1)	87.7	69.4	**53.3**	96.1	86.2	**57.6**
Sy-CI (γ=1.15)	**88.5**	**69.7**	52.8	**96.6**	**86.5**	57.1
Sy-CI (γ=1.2)	86.8	67.8	51.1	96.4	85.5	56.8
Sy-RCD (γ=1.05)	89.0	72.1	56.4	95.0	86.3	75.2
Sy-RCD(γ=1.1)	**89.4**	73.6	**57.6**	95.5	87.1	**76.5**
Sy-RCD(γ=1.15)	89.0	**74.3**	57.1	**96.0**	**87.9**	76.4
Sy-RCD(γ=1.2)	88.3	74.2	56.8	95.8	87.1	76.6
I/F-CI(γ=1.1)	85.2	63.8	47.1	95.0	82.7	68.7
I/F-CI(γ=1.15)	**85.9**	**65.0**	**48.3**	95.6	83.8	**69.8**
I/F-CI(γ=1.2)	84.6	64.3	48.1	**96.0**	**84.0**	69.4
I/F-CI(γ=1.25)	83.9	64.0	47.0	95.8	83.6	68.6
I/F-RCD(γ=1.1)	83.0	64.3	48.6	94.0	82.2	68.7
I/F-RCD(γ=1.15)	85.3	66.2	50.3	**94.5**	82.9	71.2
I/F-RCD(γ=1.2)	**85.7**	**68.6**	51.3	94.3	83.5	**71.9**
I/F-RCD(γ=1.25)	85.4	67.9	**51.6**	94.1	**83.6**	71.4
IFC-CI(γ=1.1)	87.4	69.1	53.7	97.0	86.8	73.5
IFC-CI(γ=1.15)	88.1	70.3	54.5	**97.1**	86.9	74.8
IFC-CI(γ=1.2)	**88.2**	**70.7**	**54.1**	96.8	**87.1**	**75.3**
IFC-CI(γ=1.25)	87.8	70.2	52.7	96.4	86.6	74.3
IFC-RCD(γ=1.1)	85.8	65.7	47.4	95.4	84.5	70.1
IFC-RCD(γ=1.15)	**86.2**	67.0	49.5	96.0	85.5	71.4
IFC-RCD(γ=1.2)	85.4	66.9	49.0	**96.3**	**85.5**	**72.0**
IFC-RCD(γ=1.25)	85.4	**67.7**	**49.7**	96.0	84.8	72.4

TABLE IV
RECOGNITION RATE (%) OF SPOTTING SYSTEMS, WHERE CI UNITS ARE USED TO MODEL FILLERS WHILE RCD UNITS TO KEYWORDS

	TOP 1			TOP 5		
Vocabulary	**500**	**5k**	**25k**	**500**	**5k**	**25k**
Sy (γ=1.0)	89.8	72.0	53.9	96.6	86.9	75.2
Sy (γ=1.05)	91.5	75.0	56.4	96.6	88.3	77.9
Sy (γ=1.1)	**91.6**	**76.5**	**57.7**	**97.1**	89.6	**79.4**
Sy (γ=1.15)	91.4	76.4	57.7	97.0	**90.2**	78.9
Sy (γ=1.2)	86.9	64.9	46.7	96.9	83.6	70.0
I/F(γ=1.0)	86.9	67.9	49.0	94.7	83.8	71.4
I/F(γ=1.1)	**90.2**	72.9	54.9	96.1	**87.9**	75.9
I/F(γ=1.15)	89.9	**73.8**	**55.3**	**96.8**	87.6	**77.4**
I/F(γ=1.2)	89.6	74.2	54.3	96.8	88.0	77.3
I/F(γ=1.25)	83.8	61.5	42.3	96.1	83.9	66.3
IFC(γ=1.0)	88.3	70.5	52.5	96.2	86.4	77.3
IFC(γ=1.1)	**92.0**	77.8	61.2	97.5	89.6	81.6
IFC(γ=1.15)	91.8	**78.1**	**61.5**	**97.5**	**90.6**	**82.1**
IFC(γ=1.2)	91.2	77.6	60.6	97.4	90.9	81.4
IFC(γ=1.25)	90.6	76.9	60.0	97.3	90.6	80.9

- The noticeable condition is that the INITIAL–FINAL-constrained (IFC) fillers can achieve better performance than the syllabic fillers. Since the number of modeling units for the syllabic fillers is much larger than the IFC ones, the complexity of the search will be lower for IFC than that for syllabic units. Hence this condition will emphasize the merits of the IFC fillers.
- There is more than 5% rate difference between the TOP 1 rate and the TOP 5 rate, which indicates that there is still large space for finding the better methods.

V. CONCLUDING REMARKS

This paper has considered the design of vocabulary-independent keyword spotters for Mandarin speech. We established keyword spotting systems based on the framework in Fig. 1. The framework can model all speech other than the keywords as extraneous speech and well integrated with the tree-trellis search algorithm to achieve efficient search in large vocabulary systems. We have designed three filler structures for the framework, named syllabic fillers, I/F (INITIAL/FINAL) fillers, and I–F-constrained fillers. Also, the three structures can be constructed either by the context-dependent units or the context-independent ones. Section II has shown the performance of the three fillers. The results lead to three important arguments, which motivates the study in Sections III–V of the paper. For reducing the filler space mentioned in ***Argument 2***, Section III has presented an antitrust factor to control the space. To achieve a good tradeoff in modeling accuracy and filler space mentioned in ***Argument 3***, Section IV shows an inhomogeneous modeling method. From Tables I and IV, the inhomogeneous model jointed with antitrust factor can improve the original system by 5.8%, 9.3%, and 8.7% individually for 500-, 5000-, and 25 000-words systems. Also, if we compare the results between Tables II and IV, there is still large gap between the presented methods and the ideal spotters, which indicates that there is still large space for finding the better methods.

REFERENCES

[1] E. F. Huang, H. C. Wang, and F. K. Soong, "A fast algorithm for large vocabulary keyword spotting application," *IEEE Trans. Speech Audio Processing*, vol. 2, pp. 449–452, July 1994.

[2] J. C. Junqua and J. P. Haton, "Robustness in automatic speech recognition," in *Word-Spotting and Rejection*. Norwell, MA: Kluwer, 1996, ch. 10.

[3] C. J. Lee and E. F. Huang, "A large vocabulary keyword spotter for automated Mandarin phone directory assistant services," in *Proc. ISMIP*, Dec. 1996, pp. 577–582.

[4] R. Rose and D. Paul, "A hidden Markov model based keyword recognition system," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, 1990, pp. 129–132.

[5] A. Manos and V. Zue, "A segment-based wordspotter using phonetic filler models," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, 1997, pp. 899–902.

[6] H. Boulard, B. D'joore, and J. Boite, "Optimizing recognition and rejection performance in word spotting systems," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, vol. I, 1994, pp. 373–375.

[7] R. Rose, "Automatic speech and speaker recognition," in *Word Spotting-Extracting Partial Information from Continuous Utterances*. Norwell, MA: Kluwer, 1996, ch. 13.

[8] E. F. Huang, F. K. Soong, and H. C. Wang, "The use of tree- trellis search for large-vocabulary Mandarin polysyllabic word speech recognition," *Comput. Speech Lang.*, vol. 8, pp. 39–58, 1994.

[9] C. M. Liu, H. Y. Chang, and B. Chen, "The use of tree-trellis search for speaker independent Mandarin polysyllabic word recognition," in *Proc. Int. Conf. Computer System Technology Industrial Applications*, 1996.

[10] H. Y. Chang, B. Chen, C. S. Chou, and C. M. Liu, "Speaker-independent Mandarin polysyllabic word recognition," in *4th Int. Symp. Signal Processing Applications*, 1996.

[11] H. M. Wang *et al.*, "Complete recognition of continuous Mandarin speech for Chinese language with very large vocabulary using limited training data," *IEEE Trans. Speech Audio Processing*, vol. 5, pp. 195–200, Mar. 1997.