AN MRNN-BASED METHOD FOR CONTINUOUS MANDARIN SPEECH RECOGNITION

Yuan-Fu Liao and Sin-Horng Chen

Department of Communication Engineering,
National Chiao Tung University, Hsinchu, Taiwan, R.O.C.
Tel: +886-3-5731822, Fax: +886-3-5710116, Email: schen@cc.nctu.edu.tw

ABSTRACT

A new MRNN-based method for continuous Mandarin speech recognition is proposed. The system uses five RNNs to accomplish many subtasks separately and then combine them to integrally solve the problem. They include two RNNs for the discriminations of the two sub-syllable groups of 100 RFD *initial*s and 39 CI *final*s, two RNNs for the generations of dynamic weighting functions for sub-syllable's integration, and one RNN for syllable boundary detection. All RNN modules are combined using a delay-decision Viterbi search. The method differs from the ANN/HMM hybrid approach on using ANNs to perform not only sub-syllables discrimination but also temporal structure modeling of speech signal. The system is trained using a three-stage training method embedding with the MCE/GPD algorithms. Besides, fast recognition method using multi-level pruning is also proposed. Experimental results showed that it outperforms the HMM method on both the recognition accuracy and the computational complexity.

1. INTRODUCTION

Recently, hybrid ANN/HMM speech recognition have become an attractive research topic because it integrates the advantages of artificial neural networks and hidden Markov models. One popular approach is to replace the mixture Gaussian observation probability functions of conventional HMM models with non-parametric ANN pattern classifiers to take advantage of high discrimination capability of ANN resulted from competitive training. The modeling of temporal structure of speech signal is still performed implicitly under the HMM framework [1,2].

Although this approach is effective, there is still some room for performance improvement. In this approach, ANN acts simply as a non-parametric approximator of state emission probability function and usually trained with a criterion to maximize the classification accuracy of the phoneme classes. But, this does not consistently comply with the final goal of speech recognition to maximize the word or sentence recognition accuracy [3,4]. Moreover, the ANN outputs are often not directly used in HMMs as likelihood functions. They need to be scaled according to *a priori* probabilities [1,2]. But, this scaling operation has no appropriate interpretations to explain the role it takes in the final goal of minimizing the word/sentence error rate. So, it is potentially risky and may lead to a reduction of the word/sentence recognition rate. Another major weakness of the ANN/HMM hybrid method is that the ability of ANN is not fully utilized. For instances, it does not use a priori domain knowledge in ANN's architecture design to improve the word/sentence recognition accuracy. It also does not use ANN in the modeling of the temporal structure of speech signal to provide useful cues for further improving the recognition performance.

In this paper, a modular recurrent neural network (MRNN) method for the recognition of continuous Mandarin speech is proposed. It uses an MRNN containing several RNN modules to conquer subtasks separately and then combine them to integrally solve the complicated task of continuous Mandarin syllable recognition. It differs from the conventional ANN/HMM hybrid approach on adopting a pure ANN approach to perform not only the task of phonemes or sub-syllables discrimination but also the task of temporal structure modeling of speech signal. Besides, it incorporates a novel multi-level pruning method into the Viterbi search to speed up the recognition process. So it is an effective and efficient method.

The organization of this paper is stated as follows. Section 2 presents the proposed method. A three-stage training method, embedding with sub-syllable-, syllable-, and string-level MCE/GPD algorithms, and the multi-level pruning method are also discussed. Experiments for examining the performance of the proposed method are described in Section 3. Some conclusions are given in the last section.

2. THE PROPOSED MRNN SYSTEM

2.1 The MRNN Architecture

Figure 1 shows a block diagram of the proposed MRNN recognizer. It adopts the "divide and conquer" principle to decompose the complicated task of continuous Mandarin syllable recognition, in both spatial and temporal domains, into several subtasks. Specifically, the task is first divided into two subtasks involving speech segmentation and base-syllable discrimination. The former is to segment the input speech by classifying each input frame into two classes of syllable boundary and non-syllable boundary and is accomplished by a segmentation RNN. The latter is to discriminate 411 base-syllables and is further decomposed into four subtasks including two to discriminate the two sets of 100 right-*final*-dependent (RFD) *initial*s and 39 context-independent (CI) *final*s, and two to generate appropriate dynamic weighting functions. Each of these four subtasks is accomplished by an RNN. A delay-decision, frame-synchronized Viterbi search algorithm is lastly used to integrate all subtasks to find out the best recognized base-syllable sequence. In the following, we discuss

This work was supported by National Science Council, Taiwan, ROC. under contract NSC87-2213-E009-027.

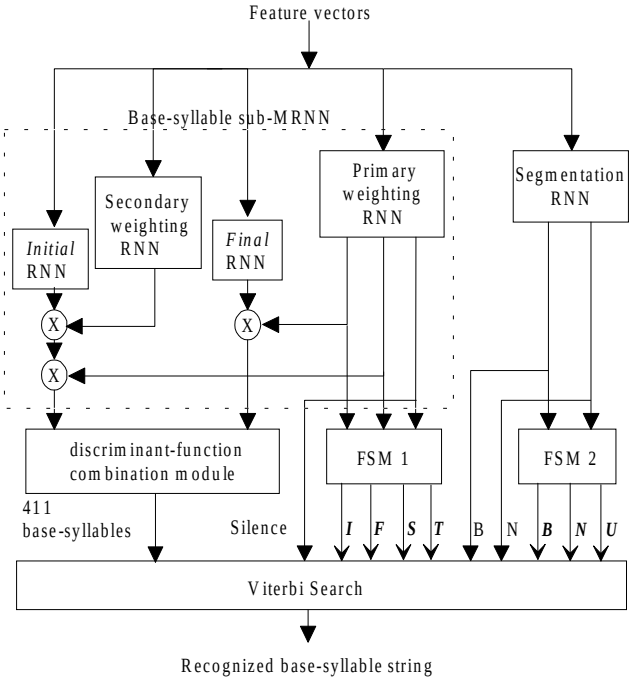

Figure 1. The block diagram of the proposed system.

the functions of the final RNN modules and the Viterbi search in more detail.

The base-syllable discrimination sub-MRNN is composed of 4 RNNs: *initial* RNN, *final* RNN, primary weighting RNN and secondary weighting RNN. The functions of *initial* and *final* RNNs are to generate discriminant functions respectively for the two sub-syllable groups of 100 RFD *initial*s and 39 CI *final*s. The primary weighting RNN generates 3 primary dynamic weighting functions respectively for silence and the two groups of 100 RFD *initial*s and 39 CI *final*s. The secondary weighting RNN generates 9 additional secondary dynamic weighting functions for the 9 *initial* sub-groups which partition the set of 100 RFD *initial*s according to the manner of articulation. All these dynamic weighting functions are used to combine the discriminant functions generated by the *initial* and *final* RNNs to form discriminant functions for the entire set of 411 base-syllables. We note that the silence discriminant function is generated directly by the primary weighting RNN.

The segmentation RNN explicitly models the temporal structure of speech signal by detecting all possible syllable boundaries directly from acoustic features. It generates two dynamic transition weighting functions for syllable-boundary and non-syllable-boundary. These dynamic transition weighting functions are used to combine the base-syllable and silence discriminant-functions to form complete discriminant-functions for base-syllable sequence hypotheses $g(X,S;\Lambda)$. Here X is the input utterance, S is a possible base-syllable sequence, and Λ is the model parameters. The final goal is to find out the best base-syllable sequence $\tilde{S}$ among all possible ones. The decision rule is defined as

$$\tilde{S} = \arg\max_{S} g(X,S;\Lambda) = \sum_{t=0}^{L-1}\left[syllable_{q_{syl}(t)}(t) + transition_{q_{trans}(t)}(t)\right]$$

where

$$syllable_{q_{syl}(t)}(t) = \begin{cases} W_I(t)\cdot W^g_{q_g(t)}(t)\cdot O^I_{q_I(t)}(t) + W_F(t)\cdot O^F_{q_F(t)}(t) & \\ & \text{if } q_{syl}(t) \text{ is a syllable} \\ W_S(t) & \text{if } q_{syl}(t) \text{ is a silence} \end{cases}$$

is the base-syllable discriminant function;

$$transition_{q_{trans}(t)}(t) = \begin{cases} O^{Seg}_B(t) & \text{if } q_{trans}(t) \text{ is a syllable-boundary} \\ O^{Seg}_N(t) & \text{if } q_{trans}(t) \text{ is a non-syllable-boundary} \end{cases}$$

is the dynamic transition weighting function generated by the segmentation RNN; L is the length of the input utterance; $q_{syl}(t)$ and $q_{trans}(t)$ are the best syllable and transition path corresponding to S; $W_I(t)$, $W_F(t)$, and $W_S(t)$ are the three outputs of the primary weighting RNN; $W^g_{q_g(t)}(t)$, $O^I_{q_I(t)}(t)$, and $O^F_{q_F(t)}(t)$ are, respectively, the outputs of the secondary weighting RNN, *the initial* RNN, and the *final* RNN; $q_g(t)$, $q_I(t)$ and $q_F(t)$ are, respectively, the classes of the *initial* sub-group, *initial*, and *final* corresponding to $q_{syl}(t)$. We note that the two transition weighting functions, $O^{Seg}_B(t)$ and $O^{Seg}_N(t)$, generated by the segmentation RNN, are dynamic functions so that they can provide precise syllable boundary cues to help the modeling of temporal structure of speech signal. They are hence completely different from the static state transition probability a_{ij} used in the HMM method.

In the recognition, a delay-decision Viterbi search is used to find out the best base-syllable sequence. It represents each base-syllable by a single state, and retains several delayed cumulative scores for it in the recognition search. The delay-decision is introduced to conquer the effect of finite pulse duration of $O^{Seg}_B(t)$ and $O^{Seg}_N(t)$ when using it in the calculation of $g(X,S;\Lambda)$.

2.2 The Three-Stage MCE/GPD Training method

To efficiently train the MRNN speech recognizer, a special three-stage training method embedding with sub-syllable-, syllable- and string-level MCE/GPD algorithms [5] is used. It first separately trains the *initial* RNN, the *final* RNN using the sub-syllable-level MCE/GPD algorithm, and trains the primary and secondary weighting RNNs using the error back-propagation (EBP) algorithm in the first training stage. These 4 RNNs are then combined together in the second training stage to form the base-syllable sub-MRNN and fine-tuned by the syllable-level MCE/GPD algorithm. In the third training stage, the segmentation RNN is first independently trained and then integrated with the base-syllable discrimination sub-MRNN to form the MRNN recognizer. The whole MRNN system is then further fine-tuned by

the string-level MCE/GPD algorithm. In the following, the three training stages are discussed in more detail.

Before the training, all training utterances are segmented into *initial*, *final*, and silence segments in advance. Segmentation is realized by using an HMM-based method to find out all *initial*/*final*/silence boundaries. Then, each *initial*/*final* boundary is relaxed a little bit to let the two neighboring *initial* and *final* segments overlap by several frames. The *initial* and *final* RNNs are then separately trained using the corresponding *initial* and *final* segments by the sub-syllable-level MCE/GPD algorithm. And, the two weighting RNNs are trained independently by the EBP algorithm to generate appropriate dynamic weighting functions for the two groups of *initial* and *final*, silence, and 9 *initial* sub-groups. Here, "$0-1$" output target functions determined according to the segmentation results are used for all 12 dynamic weighting functions. It is worth noting that the intra-syllable coarticulation effect can be partially compensated by using overlapping *initial* and *final* segments to train the four RNNs.

In the second training stage, the four pre-trained RNNs are combined to form a base-syllable discrimination sub-MRNN and then fine-tuned using a "bootstrapping" procedure embedding with a syllable-level MCE/GPD algorithm. In the bootstrap fine-tuning procedure, the four RNNs of the base-syllable discrimination MRNN are divided into 2 parts and retrained part-by-part.

In the third training stage, the segmentation RNN is first trained independently by an EBP algorithm to generate appropriate dynamic transition weighting functions and then combined with the base-syllable discrimination sub-MRNN. The whole MRNN is then fine-tuned by a string-level MCE/GPD algorithm to adjust all constituent RNNs at the same time. In the string-level MCE/GPD training algorithm , the misclassification measure [5] is defined as

$$d(X;\Lambda) = -g(X, S_0;\Lambda) + \ln\left(\frac{1}{r_m - 1}\sum_{r=1}^{r_m}\exp\left[g(X, S_r;\Lambda)\cdot\eta\right]\right)^{\frac{1}{\eta}}$$

where S_0 is the desired base-syllable sequence, r_m is the total number of competing base-syllable sequences other than S_0. The loss function is defined as

$$l(X;\Lambda) = \mathrm{sigmod}\left(d(X;\Lambda)\right).$$

We note that the "$0-1$" bounds of secondary dynamic weighting functions, set as learning targets in the first-stage training, are relaxed and freed of any manual control in the second- and third-stage trainings. The ultimate level that a secondary dynamic weighting function can reach is determined automatically by the syllable- and string-level MCE/GPD algorithms. We also note that the "$0-1$" bounds of primary dynamic weighting functions and dynamic transition weighting functions are also relaxed in the third-stage training. The ultimate levels that these dynamic functions can reach are automatically adjusted by the string-level MCE/GPD algorithm. The discrimination capacity of the base-syllable sub-MRNN can hence be increased via placing special emphases on the most distinguishing parts of the input test utterance for each candidate syllable.

2.3 The Multi-Level Pruning Method

To speed up the recognition process, a novel multi-level pruning method to be incorporated into the Viterbi search is proposed. It uses some useful acoustic cues provided by the final RNN modules to prune many unnecessary path searches with very small extra efforts. It is a combination of the following three pruning schemes: syllable deactivation, pre-classification based pruning and pre-segmentation based pruning. The syllable deactivation scheme uses an idea similar to the phone deactivation method to eliminate the recognition searches for all paths containing unlikely base-syllables with very low base-syllable discrimination function scores. The pre-classification based pruning scheme uses the idea of setting more restrict searching constraints for the stable parts of the input speech signal to eliminate unnecessary path searches. The pre-segmentation based pruning scheme uses a similar idea to set more restrict syllable transition/non-transition constraints for syllable-boundary and non-syllable-boundary parts of the input speech to eliminate unnecessary syllable transition/non-transition tests in the recognition search. We discuss the latter two schemes in more detail as follows.

As mentioned previously, the primary weighting RNN generates three outputs to discriminate each input frame among the three classes of silence, *initial*, and *final*. The pre-classification based pruning scheme uses these three outputs to drive a pre-classification FSM to classify and label the frame into four stable states of silence (***S***), *initial* (***I***), *medial* (***M***) and *final* (***F***), and one transient (***T***) state. Specifically, the pre-classification FSM compares the three outputs of the primary weighting RNN with two threshold values, TH_L and TH_H. While one (*initial*, *final* , or silence) output is higher than TH_H and the other two are all lower than TH_L, the FSM moves into the corresponding stable (***I, F,*** or ***S***) state if it is a legal one. When both *initial* and *final* outputs are higher than TH_H and the silence output is lower than TH_L, the FSM moves into the ***M*** state. Otherwise, it goes to the ***T*** state. Similarly, the pre-segmentation based pruning scheme uses a pre-segmentation FSM driven by the two outputs of the segmentation RNN to classify and label each input frames into two certain states of syllable-boundary (***B***) and non-syllable-boundary (***N***) and one uncertain (***U***) state.

We then use the output labels of these two FSMs to explicitly model the temporal structure of the input speech signal and prune unnecessary search paths so as to speed up the recognition process. Specifically, when an input frame is labeled as an ***I*** or ***M*** (***M*** or ***F***) state, the delay-decision Viterbi search only allows the frame to stay in the beginning (ending) states of all base-syllables. As an input frame is labeled as a ***S*** state, we let it to stay in silence. When an input frame is labeled as a ***B*** (***N***) state, the search is allowed (not allowed) to jump into or leave a base-syllable state. For ***T*** and ***U*** states, a full search is performed. It is worth to note that, from the viewpoint of the DP search, the resulting path constraining scheme is a partial-hard-decision-and-partial-soft-decision one [6].

	Ins.	Del.	Sub.	Syl. Acc.	Str. Acc	Para.
HMM/Mix5	33	91	1304	79.7%	11.2%	104,800
HMM/Mix8	35	85	1227	80.9%	12.7%	161,850
HMM/Mix10	39	83	1232	80.8%	12.9%	195,600
MRNN/100	44	83	1118	84.1%	18.0%	88,853
MRNN/150	44	88	1066	84.8%	20.2%	133,403
MRNN/200	34	105	875	85.6%	19.5%	187,953

Table 1. The recognition results of the HMM and MRNN methods.

	Active syllable states	Active syllable transition paths	Syllable accuracy rate	String accuracy rate
Baseline	x	x	85.6%	19.5%
deactivation	98.4%	98.4%	85.5%	19.5%
Pre-classification	79.7%	68.8%	85.6%	19.5%
Pre-segmentation	x	36.1%	85.6%	19.5%
Multi-level	82.0%	35.6%	85.6%	19.7%

Table 2. The experimental results of the multi-level pruning method.

3. SIMULATIONS

Effectiveness of the proposed method was examined by simulations using a continuous Mandarin speech database uttered by a single male speaker. The database contains in total 35,231 syllables including 28,197 training syllables (1933 sentences) and 7034 testing syllables (544 sentences). All speech signals were A/D converted at a rate of 10 kHz and then pre-emphasized with a digital filter, $1-0.95z^{-1}$. It is then analyzed to extract recognition features for each 20-ms Hamming-windowed frame with 10-ms frame shift. The recognition features include 12 MFCC, 12 delta MFCC, and a delta-energy. All RNNs used in our simulation have the same 3-layer, simple recurrent structure with all outputs of the hidden layer being fed back to itself as additional inputs. The length of input window is 7 frames for the segmentation RNN and is 5 frames for the other four RNNs. All output-layer nodes in each RNN use linear output functions rather than the more commonly sigmod functions.

We now examine the performance of the proposed MRNN recognizer. All five RNNs were first trained separately and then combined to be retrained. 10 iterations were performed in each of the second and third retraining stages. In the third-stage string-level MCE/GPD training, top 20 best base-syllable sequences were used. The experimental results are listed in Table 1. The best base-syllable and string accuracy rates are 85.6% and 20.2%, respectively. It is noted that the low string recognition rate is owing to the use of no language model in the recognition search.

For performance comparison, the conventional HMM method using the same basic recognition units was also tested. It employed 100 3-state RFD *initial* HMM models, 39 5-state CI *final* HMM models to form 411 8-state base-syllable HMM models and used a single state HMM model for silence. The number of mixture Gaussian components in each state of these sub-syllable HMM models varies from 1 to *M* depending on the number of training data. Three values of *M* were examined. They are 5, 8, and 10. Experimental results are also listed in Table 1 The best base-syllable and string accuracy rates are 80.9% and 12.9%, respectively. They are all inferior to those obtained by the proposed MRNN method.

A rough comparison of the computational efficiencies of the HMM and MRNN methods based on the total number of parameters used was then checked. The experimental results are shown in the last column of Table 1. It can be seen from Table 1 that the best MRNN method uses slightly fewer parameters than the best HMM method. If the computational complexities of the recognition searches are also considered, the proposed MRNN method is more efficient than the conventional HMM method.

Lastly, the efficiency of the multi-level pruning method is examined. In this test, all threshold values of the two FSMs are empirically determined to meet the condition of keeping the recognition accuracy almost the same as the baseline system. The experimental results are listed in Table 2. It can be seen from the table that only 82.0% of active syllable states and 35.6% of active syllable transition paths are needed to be searched with no pay on the degradation of recognition accuracy. It is a dramatic saving.

4. CONCLUSIONS

A novel MRNN-based method for continuous Mandarin speech recognition has been discussed. It differs from the ANN/HMM hybrid approach on adopting a pure ANN approach to perform not only the task of phoneme (sub-syllable) classes classification but also the task of temporal structure modeling. Experimental results have confirmed that it outperforms the conventional HMM method on both the recognition accuracy and the computational complexity. So it is a promising method for continuous Mandarin speech recognition. Further studies to incorporate it with a tone recognizer and a language model are worth doing in the future.

5. REFERENCES

[1] H. Bourlard and N. Morgan, "Connectionist speech recognition - A hybrid approach", *Kluwer Academic Publishers*, 1994.

[2] A. J. Robinson, "An application of recurrent nets to phone probability estimation", *IEEE Trans. on Neural Networks*, Vol. 5, No. 2, pp. 298-305, March 1994.

[3] Xin Tu, Yonghong Yan and Ron Cole, "Matching Training And Testing Criteria in Hybrid Speech Recognition Systems", *Eurospeech97*. pp. 1943-1946, 1997.

[4] Mike Schuster, "Incorporation of HMM Output Constraints in Hybrid NN/HMM Systems during Training", *Eurospeech97*. pp. 2843-2846, 1997.

[5] B. H. Juang, W. Chou, and C. H. Lee, "Minimum Classification Error Rate Methods for Speech Recognition", *IEEE Trans. on Speech and Audio Processing*, Vol. 5, No. 3, pp. 257-265, May, 1997.

[6] S. H. Chen, Y. F. Liao, S. M. Chiang, and S. Chang, "An RNN-Based Pre-classification Method for Fast Continuous Mandarin Speech Recognition", to appear in *IEEE Trans. on Speech and Audio Processing*.