SPEECH RECOGNITION USING NEURAL NETWORKS WITH FORWARD-BACKWARD PROBABILITY GENERATED TARGETS

Yonghong Yan Mark Fanty Ron Cole

Center for Spoken Language Understanding
Oregon Graduate Institute of Science and Technology
P.O. Box 91000, Portland, OR 97291-1000
{yan, fanty, cole}@cse.ogi.edu

ABSTRACT

Neural network training targets for speech recognition are estimated using a novel method. Rather than use zero and one, continuous targets are generated using forward-backward probabilities. Each training pattern has more than one class active. Experiments showed that the new method effectively decreased the error rate by 15% in a continuous digits recognition task.

1. Introduction

A new training method for hybrid speech recognition systems is presented in this paper. Hybrid approaches for speech recognition are motived by exploiting the advantages of Hidden Markov Models (HMMs) and Neural Networks (NNs). HMMs have been used extensively in speech recognition for their ability to model the temporal information in speech, while NNs have been shown to be a powerful tool for static classification tasks due to their discriminant nature. A variety of NN-based approaches have been reported (e.g. [14, 4, 10, 6, 2, 13, 12, 7, 9, 8, 5, 11]).

In most of these approaches, the NNs were used as a posterior probability estimator. The outputs of the NNs were used as the state emission probabilities within a Morkov chain. Typically, a single network is used to estimate probabilities for all classes (phones or sub-phonetic units). To train the network, each training pattern is assigned a "desired" output vector, i.e. a desired value for each of the classes. In general, a Viterbi-based forced alignment method is employed (when phonetically transcripted data are not available) and the subword units in the best path found by Viterbi decoding are used to generate the training targets. For each feature vector, only a single class is assigned a target of 1.0 and the rest are assigned a target of 0.0.

Although it is very simple computationally, this method does not accurately reflect the nature of speech in a training pattern. For features based on short-time analysis, the distinction between contiguous phones (or other modeling units) at the transition region is ambiguous; the speech vector represents both classes. It is not appropriate to force a hard decision. A reasonable set of targets should represent this; a soft (probabilistic) decision is more appropriate. Attempts along this direction have been made recently, in [11] a frame work called REMAP was proposed and applied to a transition-based model to estimate the posterior probabilities, and in [10] a VQ-like technique was used to estimate the fuzzy likelihoods.

This paper describes a method for estimating continuous targets for training patterns of NNs based on the conventional forward-backward algorithm. The targets used to train the neural network are derived from the posterior state occupation probabilities.

This approach was evaluated on a continuous digit telephone speech database. For comparison, evaluations were also done for a hybrid system using our previous work (trained with Viterbi forced alignment with only 1 fixed non-zero target per training pattern) ([1]) and for a continuous HMM-based recognizer, using the same training, development and testing sets. The results showed more than a 15% error reduction for the hybrid system using the new training method.

The rest of this paper is organized as: Section 2 describes how neural network targets are derived using the forward-backward algorithm. Section 3 describes our hybrid system. Section 4 describes the comparative experiments and results. Concluding remarks are given in Section 5.

2. Target Estimation Using the Forward-backward Probabilities

The common practice for generating targets of training patterns is based on Viterbi forced alignments on the training utterances. Each input speech vector (training pattern) is assigned to one class with a non-zero target, and the targets for the rest of the classes are set to zero. In theory, a NN trained using discrete targets of 0 and 1 will produce posterior probabilities as outputs (such outputs minimize the training error). However, this theoretical result relies on unlimited training data and network resources. In fact, networks are severely limited in the amount of training data which can be used. Unlike HMMs, not every frame of the training data is in the training set. These data must be subsampled to achieve reasonable training times. Examination of the behavior of NN outputs in ambiguous regions has shown that they often behave unreasonably, making extreme decisions which are often wrong. We hypothesize that a more reasonable target set which represents the true probabilities will simplify learning, and will increase the generalizationability of the trained network given the same amount of data (especially for outputs with sparse training data, which are common for context-dependent modeling).

In HMM training, the forward-backward algorithm achieves model optimization by estimating the posterior probability of being each state and the posterior probability of state transitions for each observation. These posterior probabilities are estimated based on the forward and backward probability calculations. In our proposed work, by viewing the outputs of an initialized neural net on the training utterances as emission probabilities of Markov states, the targets for input training patterns are reestimated within the framework of the forward-backward algorithm.

In a HMM system, the probability of the observation sequence O is defined as $P_r(O|S,\pi)$:

$$P_r(O|S,\pi) = b_{s_1}(O_1)b_{s_2}(O_2)\cdots b_{s_T}(O_T) \tag{1}$$

where S is the state sequence of Markov chain, O_i is the ith observation and π is the model set.

The probability of the corresponding state sequence is defined as:

$$P_r(S|\pi) = a_{s_1 s_2}a_{s_2 s_3}\cdots a_{s_{T-1}s_T} \tag{2}$$

where s_i is the ith state in the state sequence. The likelihood $P_r(O|\pi)$ is given as:

$$\begin{aligned} P_r(O|\pi) &= \sum_{all\ S} P_r(O|S,\pi)P_r(S|\pi) && (3)\\ &= \sum_{all S}\prod_{t=1}^{T} a_{s_{t-1}s_t}b_{s_t}(O_t) && (4)\end{aligned}$$

The forward probability is defined as:

$$\begin{aligned} \alpha_t(i) &= P_r(O_1,O_2,\cdots,O_t,S_t=i|\pi) && (5)\\ &= \sum_i \alpha_{t-1}(i)a_{ij}b_j(O_t) && (6)\end{aligned}$$

The backward probability is defined as:

$$\begin{aligned} \beta_t(i) &= P_r(O_{t+1},O_{t+2},\cdots,O_T|S_t=i,\pi) && (7)\\ &= \sum_i a_{ji}b_i(O_{t+1})\beta_{t+1}(i) && (8)\end{aligned}$$

So the posterior probability of transitions λ_{ij}, from state i to state j given the observation and model can be computed as:

$$\begin{aligned} \lambda_{ij}(t) &= P_r(s_t=i,s_{t+1}=j|O,\pi) && (9)\\ &= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{P_r(O|\pi)} && (10)\\ &= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\sum_{k\in S}\alpha_T(k)} && (11)\end{aligned}$$

The posterior probability of being in state i at time t can be computed as:

$$\begin{aligned} \lambda_i(t) &= P_r(s_t=i|O,\pi) && (12)\\ &= \frac{\alpha_t(i)\beta_t(i)}{\sum_{k\in S}\alpha_T(k)} && (13)\end{aligned}$$

And the transitions between states of a model can be estimated as:

$$\begin{aligned} \bar{a}_{ij}(t) &= \frac{\sum_{t=1}^{T-1}\lambda_{ij}(t)}{\sum_{t=1}^{T-1}\sum_k \lambda_{ik}(t)} && (14)\\ &= \frac{\sum_{t=1}^{T-1}\lambda_{ij}(t)}{\sum_{t=1}^{T-1}\lambda_i(t)} && (15)\end{aligned}$$

During training, an initialized NN is needed. The outputs of the initialization network are used as the emission probabilities of the Markov states, and the initialization network is retrained using the generated targets (in (12)).

3. The Hybrid System

The hybrid system used for this study is very similar to our previous approach described in [1], except within-model state transitions are implemented in this new system.

Like most of the other hybrid systems, the NN in our system is used as a state emission probability estimator. A three-layer fully-connected NN was used in this study. The modeling units are phones. Each phone has one to three states, and each state corresponds to an output node of the NN.

The relation between context-dependent phone models and the output nodes is illustrated in Figure 1. In the figure, L-PH-R denotes phone *PH* in the context of phone *L* (left) and *R* (right). As shown, the context-dependent phones from the same monophone shared the middle state. The left (right) state for each model only depends on the left (right) context. Thus in Figure 1, both the middle states and the end states of the the w-ay-f and l-ay-f /ay/ models use the same NN output.

Unlike most of the existing hybrid systems which do not model the within phone model transitions, our new hybrid does model the speech process as a double stochastic process. As an analogy to the naming of discrete, semi-continuous, and continuous HMMs, our new hybrid system can be called NN/HMMs (since the emission probability is modeled by a NN).

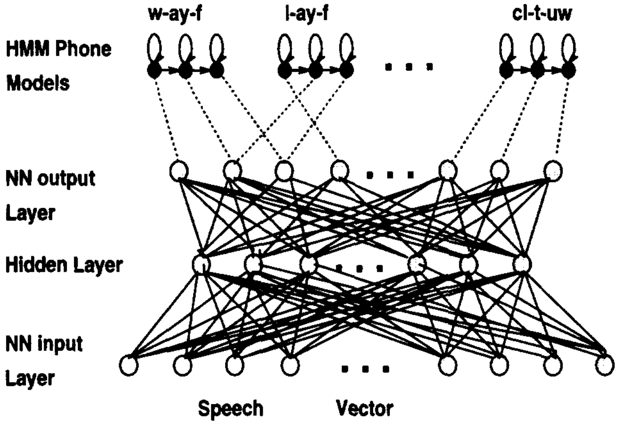

Figure 1: Relation between NN output nodes and the phone models

During training, an initialization NN was trained first using our previous Viterbi forced alignment-based method. The within-phone transition probabilities were initialized with constant numbers. Then we used the proposed forward-backward re-estimation algorithm to regenerate the targets for the training utterances. The forward-backward re-estimation is implemented in an embedded form, which concatenates the phone models in the input utterance into a "big" model and reestimates the parameters based on the whole input utterance. The NNs were trained using the standard stochastic backpropagation algorithm with mean square error as the cost function.

4. Comparative Experiments

4.1. Database and Task

The speech corpus used in this experiment consists of digit sequences taken from the public domain numbers corpus (telephone speech), collected by the Center for Spoken Language Understanding (CSLU) at OGI [3]. Each utterance contains 1 to 6 continuously pronounced digit strings. Since callers were recruited through public advertisements, and were instructed to call the data collection phone number at any time or place, the database is close to a real-world application environment. False starts, pauses, repetition, and background noise are common in the database. The vocabulary consists of: zero, oh, one, two, three, four, five, six, seven, eight and nine. The data were randomly divided into three sets, with 2090 utterances in the training set, 500 utterances in the development test set and 1600 utterances in the final evaluation set. [1]

4.2. The HMM System

A continuous phone-based HMM recognizer was implemented using HTK ([15]) for comparison purposes. Each phone was represented as a 3-state left-to-right model with a 4-element Gaussian mixture using diagonal covariances. The speech signal was parameterized every 12.8 ms with a 25.6 Hamming window. The feature vector is 26-dimensional, with 12 LPC-Cepstra plus normalized energy plus their deltas. Cepstral mean subtraction was used. There are 77 context-dependent phone models total.

4.3. The Hybrid Systems

Two hybrid systems were built for evaluating target generating methods. One is based on our previous work, which only has 0-1 as targets, and for each input speech vector only one class is assigned to a non-zero target. Within-phone state transition probabilities were not used in the baseline system.

The second system was retrained using forward-backward targets. Within-phone state transition probabilities computed with the forward-backward algorithm were used in this system.

4.3..1 The Baseline

The baseline system was based on our previous work ([1]). The NN used was a three-layer fully-connected feed-forward net. It had 56 input nodes, 200 hidden nodes and 209 output nodes. PLP analysis was carried out every 6 ms with a 10 ms window. For each frame, the resulting feature is a 8-dimensional vector (7 PLP coefficients plus energy). Seven contiguous frames were used as one NN input vector (hence the NN has 56 input nodes). The use of multi-frame features alleviates the problem caused by the HMM independence assumption about contiguous acoustic observation.

4.3..2 The New System

The network trained for the baseline system was used as the initialization net in this experiment. The new system used the same feature set as the baseline system, and has the same architecture as the initialization network.

We first ran the initialization network on the training set, and got the emission probabilities for each input vector. The within-phone transition probabilities were initialized as (0.6, 0.4) for each state (0.6 as the probability to stay and 0.4 as the probability to escape to the next state). The words in each utterance were instantiated to phone strings using the pronunciation dictionary. The embedded forward-backward algorithm was run on each utterance and generated new targets for the speech vectors. Once all the new targets for the training set were re-estimated, a new NN was trained.

We iterated the above process twice (the second time using the network trained in the first iteration as the initial network), and used the final trained net in our new system.

4.4. Experiments and results

The three systems are evaluated on the development set and the final test set. Results are summarized in Table 4.4..

[1] More information, including how to download a free copy of CSLU toolkit, is available via http://www.cse.ogi.edu/CSLU/

Data Set	Unit	HMM	Baseline	New
Dev. set	Word	4.1%	4.1%	3.1%
	String	13.2%	13.0%	11.4%
Test Set	Word	5.7%	6.0%	4.9%
	String	18.9%	19.7%	16.7%

Table 1: Word and String (Sentence) Error Rates for the three systems

Also, we compared the impact of the within-model state-transition modeling to the system performance. The results showed that the modeling has little effect on word correctness but decreased the insertion error by 30%, thereby increasing the sentence accuracy by 5%.

5. Concluding Remarks

This paper reported our first attempt to improve the posterior probability estimation using NNs. Encouraging results were achieved on a digit task. It effectively decreased the error rate of our baseline approach by more than 15%. Currently we are extending this work to large vocabulary recognition tasks.

6. Acknowledgement

This work was supported by the National Science Foundation, the Office of Naval Research, DARPA, and the member companies of CSLU.

7. References

[1] E. Barnard, R. Cole, M. Fanty, and P. Vermeulen. Real-world speech recognition with neural networks. In *Proceedings of the International Symposium on Aerospace/Defense Sensing & Control and Dual-Use Photonics*, April 1995.

[2] U. Bodenhausen and S. Manke. Connectionist architectural learning for high performance character and speech recognition. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages I-625–I-628, 1993.

[3] R.A. Cole, M. Noel, T. Lander, and T. Durham. New telephone speech corpora at cslu. In *Proceedings of Eurospeech'95*, pages 821–824, Madrid, Spain, 1995.

[4] M. Fanty and R. A. Cole. Spoken letter recognition. In R. P. Lippman, J. Moody, and D. S. Touretzky, editors, *Advances in Neural Information Processing Systems 3*. San Mateo, CA: Morgan Kaufmann, 1991.

[5] V. Fontaine, C. Ris, H. Leich, J. Vantieghen, S. Accaino, and D. Compernolle. Comparison between two hybrid hmm/mlp approaches in speech recognition. In *Proceedings 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 3362–3365, Atlanta, USA, May 1996.

[6] H. Hild and A. Waibel. Connected letter recognition with a multi-state time delay neural network. In J. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems 5*, pages 1059 – 1068. Morgan Kaufmann Publishers, Inc., 1993.

[7] H. Hutter. Comparison of a new hybrid connectionist-schmm approach with other hybrid approaches for speech recognition. In *Proceedings 1995 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 3311–3314, May 1995.

[8] K. Kasper, H. Reininger, and H. Wust. Strategies for reducing the complexity of a rnn based speech recognizer. In *Proceedings 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 3354 – 3357, Atlanta, USA, May 1996.

[9] D. Kershaw, T. Robinson, and M. Hochberg. Context-dependent classes in a hybrid recurrent network-hmm speech recognition system. In D.S. Touretzky, M.C. Mozer, and M.E. Hasselmo, editors, *Advances in Neural Information Processing Systems 8*, pages 750 – 756. The MIT Press, 1996.

[10] Y. Komori. A neural fuzzy training approach for continuous speech recognition improvemen t. In *Proceedings 1992 IEEE International Conference on Acoustics, Speech, and Signal Processi ng*, pages 405–408, March 1992.

[11] Y. Konig, H. Bourlard, and N. Morgan. Remap-experiments with speech recognition. In *Proceedings 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 3350– 3353, Atlanta, USA, May 1996.

[12] S. Renals, M. Hochberg, and T. Robinson. Learning temporal depdendencies in continuous speech recognition. In J. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems 6*, pages 1051 – 1058. Morgan Kaufmann Publishers, Inc., 1994.

[13] J. Tebelskis. Performance through consistency: connectionist large vocabulary continuous speech recognition. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages II-259–II-262, 1993.

[14] A. Waibel, T. Hanazawa, G. Hinton, K. Shiano, and K. Lang. Phoneme recognition using time-delay neural networks. In *IEEE Trans. on Acoust., Speech, and Signal Processing*, volume 37(3), pages 328–339, 1989.

[15] S.J. Young. Htk: Hidden markov model toolkit v.14. In *Cambridge University Engineering Depart ment*, 1992.