LARGE VOCABULARY CONTINUOUS SPEECH RECOGNITION USING HTK

P.C. Woodland, J.J. Odell, V. Valtchev & S.J. Young

Cambridge University Engineering Department,
Trumpington Street, Cambridge, CB2 1PZ, England.

ABSTRACT

HTK is a portable software toolkit for building speech recognition systems using continuous density hidden Markov models developed by the Cambridge University Speech Group. One particularly successful type of system uses mixture density tied-state triphones. Recently we have used this technique for the 5k/20k word ARPA Wall Street Journal (WSJ) task. We have extended our approach from using word-internal gender independent modelling to use decision tree based state clustering, cross-word triphones and gender dependent models. Our current systems can be run with either bigram or trigram language models using a single pass dynamic network decoder. Systems based on these techniques were included in the November 1993 ARPA WSJ evaluation, and gave the lowest error rate reported on the 5k word bigram, 5k word trigram and 20k word bigram "hub" tests and the second lowest error rate on the 20k word trigram "hub" test.

1. INTRODUCTION

This paper describes the large vocabulary continuous speech recognisers that we have built using HTK (HMM toolkit). HTK is a software toolkit for building and manipulating systems that use continuous density hidden Markov models that has been developed by the Speech Group at Cambridge University Engineering Department over the last four and a half years. HTK is designed to be flexible enough to support both research and development of HMM systems and also to provide a platform for benchmark evaluations. It can be used to perform a wide range of tasks, including isolated or connected speech recognition using models based on whole word or sub-word units. HTK includes a software library as well as a number of tools (programs) that perform tasks such as coding data, various styles of HMM training including embedded Baum-Welch re-estimation, Viterbi decoding, results analysis and editing of HMM definitions. The current commercially available version of HTK, V1.5, is now in use at many sites worldwide. Full details of HTK V1.5 are given in [6].

A number of features of HTK make it especially suitable for performing large vocabulary continuous speech recognition. In particular, HTK has a unique generalised parameter tying (sharing) mechanism [5] that allows HMM systems to be constructed that are balanced between acoustic modelling detail (model complexity) and parameter estimation accuracy for a given training corpus. It has been found that tied-state mixture density triphone systems are particularly effective [7]. A tied-state word-internal triphone system was included in the final September 1992 evaluation for the 1000 word DARPA Resource Management task, and gave comparable performance to the main DARPA sites [3, 4]. A full description and all the necessary tools to build such a system are included in the HTK V1.5 distribution.

Recently, we have been working with the HTK tied-state recogniser on the 5k/20k word ARPA Wall Street Journal (WSJ) task. While working on this task we have significantly extended the capabilities of our system. We have added the ability to train and test cross-word gender dependent triphone models, and use bigram and trigram language models all with a single pass dynamic network decoder. We have also extended the state-tying approach outlined in [7] to use decision-tree based clustering which allows the synthesis of HMM models which do not occur in the training data.

This paper first describes the methods by which our mixture density tied-state triphone systems are constructed and outlines the decision tree based clustering procedure. The decoder strategies that we have used are then described and it is shown that this approach gives state-of-the-art performance on the Wall Street Journal task.

2. TIED-STATE SYSTEM

In any large vocabulary recognition system using triphone acoustic models there will be some triphone contexts for which there is very limited or even no training data i.e. "unseen" triphones. Indeed, if cross-word triphones are used the majority of triphones needed for recognition will not be observed in the training data. We have also previously found that it is important to use multiple component Gaussian mixtures to accurately characterise the data in speaker independent systems [3], however this requires a reasonable amount of data to be available for each mixture distribution estimated.

Previously we have examined a bottom-up state-clustering technique [3, 7] that groups together the corresponding states in different triphones of the same base phone if the output distributions are acoustically similar while also ensuring that there is enough data associated with each distribution to form robust estimates of parameters of mixture distributions. The parameters of these "tied-states" are then re-estimated and the complexity of the output distribution increased. State-clustering is more efficient than model clustering because it allows a far greater number of models to be formed with the same number of distributions, and in an experimental comparison on the Wall Street Journal database we found that a tied-state system had an 18% lower error rate than a model-tied system.

The bottom-up clustering approach to grouping states is appropriate when there are few unseen triphones—however when unseen triphones are common it doesn't provide a direct way of specificying the correct set of tied-states to be used to synthesise an unseen model. To tackle the problem of unseen triphones we have used a decision tree based state-clustering procedure. This approach has similarities to that adopted in [1]. A separate decision tree is constructed for each state of each base phone with the goal of grouping triphone states into a number of equivalence classes by use of a number of linguistic questions concerning the identity of the base phone and the triphone context.

When the tree-growing process starts, all the states for the current tree are associated with the single root node. At each stage of tree building a node is selected and the states associated with that node split into two parts by an appropriate question. The question to ask at each stage along with the node to split is determined so as to maximise the increase in log likelihood of the data described by the state distributions. This log likelihood can be easily calculated, since the tree clustering procedure uses unimodal Gaussian distributions each characterised by mean and variance vectors and the state occupation count computed during training. It is therefore possible to calculate the likelihood of the data used to train all of these states when a number of them are merged and characterised by a new mean and variance. The tree growing process terminates when the increase in log likelihood falls below a threshold, and a constraint is imposed that all leaf nodes have a minimum total occupation count for all states clustered together. The leaf nodes corresponding to states of unseen triphones can be determined by examining the questions at each node of the decision tree to find the appropriate equivalence class for that triphone.

Therefore the basic steps involved in building the mixture density tied-state system are as follows:–

1. Create a set of unimodal Gaussian monophone models and train using transcriptions derived from the pronunciation dictionary and the sentence orthography.
2. Create all triphones (either word-internal or cross-word) that occur in the training corpus and copy the monophone models for each required triphone context and re-train. During re-training, retain the state-occupation counts.
3. For each group of triphones that share the same base phone, compute a decision tree to group the states into equivalence classes ensuring that enough data to train mixture distributions will be associated with each class. Tie the distributions of all the states in each equivalence class. Compute the state sequence required for all unseen triphones that may be needed during recognition. Tie the transition matrices across all the triphones of each base phone.
4. Merge any triphones that are now identical. This reduces the total number of models improving decoder efficiency.
5. Re-train the state-clustered triphones and then successively increment the number of mixture components in each state by a mixture-splitting procedure.

Further details of the mixture splitting procedure are given in [7]. It should be noted that the tied-states and transition matrices produced are virtually transparent to the HTK training and recognition tools due to HTK's generalised tying mechanism.

3. DECODING STRATEGIES

The standard HTK Viterbi recogniser uses a pre-compiled static recognition network. This approach can be efficiently applied to systems that use word-internal triphones, but becomes much more complex for cross-word triphones. A bigram language model can be used with such a recogniser and an extended version of the standard HTK recogniser (with a bigram cache) was used for the bigram experiments with word-internal triphones. A second problem with a static network is that it does not readily support long-span language models such as a trigram. Although the identity of all previous words is known at each point, the search will not be admissible unless multiple histories are held in each model instance.

To solve the above problems and integrate cross-word triphones as well as trigram language models directly into the search we have developed a new single-pass dynamic network decoder architecture. Firstly to reduce the search requirements a tree-structured representation of the lexicon is used. In a large vocabulary system many words will share the first phones in a number of words and so tree-structuring reduces the number of model instances. Furthermore the search effort is concentrated in the early parts of words [2] and so a much greater reduction in computation than storage is obtained, although using context dependent models reduces this effect since fewer words will share the same initial models. Tree-structuring also ensures that the computational requirements rise more slowly than linearly with increasing vocabulary size.

Using cross-word triphones with this architecture requires multiple instances of the word final phones but not multiple instances of the first phone of the next word. At every time frame, only a few words have their final states within the pruning beam and therefore cross-word triphones have a low overhead. Bigram and trigrams can also be incorporated directly into this structure by using tree copies. In our implementation we used bigram and trigram caches.

The potential size of this network is huge (of the order of 10^{12} nodes for a 20k trigram task), and so to keep the search manageable the network is *dynamically* created when model instances are in the beam and the current path needs to be extended. A conventional beam search is used as well as an upper limit on the total number of model instances. This single pass dynamic network decoder was used for all the experiments other than those with word-internal triphones reported in this paper.

We are also developing a multiple pass recogniser. On the first (no-grammar) pass a backwards structured tree lexicon is used and list of potential starting words and scores is recorded at each time frame. The first pass uses either monophones or word-internal triphones. The second pass then uses more detailed acoustic models and applies either a bigram or trigram language model. We hope that this multiple pass approach will allow greater beamwidths to be used on the second pass than with the single pass dynamic decoder or give shorter recognition times.

4. EXPERIMENTS AND RESULTS

We have used our tied-state approach with word-internal triphones, cross-word triphones and gender dependent models to build recognisers for the ARPA Wall Street Journal (WSJ) task and a number of these systems were evaluated as part of the November 1993 ARPA WSJ evaluation.

4.1. Experimental Setup

The WSJ database is in two distinct parts—WSJ0 and WSJ1. We have built systems using either just the SI-84 training material from WSJ0 (7,193 utterances used), or the SI-284 data formed by combining training data from both

Training	Model Type	Grammar	Nov'92	si_dt_s6	si_dt_05.odd	Nov'93
SI84	wint/gi	bg	8.11	10.39	12.40	12.83†
SI84	xwrd/gi	bg	6.86	9.52	10.48	8.65
SI84	xwrd/gd	bg	6.58	9.13	9.67	8.83†
SI284	xwrd/gd	bg	5.14	6.63	7.58	6.91
SI284	xwrd/gd	tg	3.19	5.27	6.09	4.99†

Table 1: % word error rates for different model types, grammars and acoustic training data for a number of 5k WSJ test-sets. † denotes systems used for the ARPA November 1993 WSJ evaluation.

WSJ0 and WSJ1 (36,515 utterances). Silences at the start and end of each training sentence were reduced to 200ms based on Viterbi generated word alignments which meant that there remained about 14 hours of training speech in the SI-84 set and 66 hours of speech in the SI-284 set. All experiments used only the data from the Sennheiser microphone channel. The acoustic feature vector contained 12 Mel frequency cepstral coefficients and log energy, plus the first and second differentials of these coefficients giving a 39 dimension observation vector. The cepstral features were normalised for each sentence by subtraction of the cepstral mean calculated over the sentence.

We have worked on both the 5k (4,986) word closed vocabulary and 20k (19,979) word open vocabulary non-verbalised pronunciation WSJ tasks. The data used for the 20k open tests can in fact contain any of 64,000 words (64k data), and hence there are a number of words that occur in the test data that are unknown to the recogniser and hence cause errors. We have run experiments using the standard bigram and trigram grammars.

The pronunciation information came from the Dragon Wall Street Journal Pronunciation Lexicon Version 2.0 with some locally generated additions and corrections. This dictionary contains multiple pronunciations, and the most likely pronunciation for each token in the training data was found by Viterbi alignment using models built in an early version of our WSJ system. The Dragon dictionary also contains markings for stressed vowels, but no stress distinctions were made in our system, resulting in 44 base phones plus silence and optional inter-word silence models. All speech phone models had three emitting states, and a strictly left-to-right topology.

4.2. SI-84 Word-Internal System

Initially we built a system using word-internal triphones and gender independent models. This system is essentially a fairly straightforward extension to the systems that we used for the Resource Management task [3], except that the states were clustered using the decision tree method and a bigram grammar was used in recognition. The speech is first labelled using Viterbi alignment to identify the correct pronunciation of each word and any inter-word silences and then HMMs are estimated for each of the the 8,087 word-internal triphones (24,261 states) in the SI-84 training set. The states are then clustered as described in Sec. 2. to 3,701 tied-states and the tied-state representation for all the 14,344 word-internal triphones in our 29,623 word dictionary synthesised using the appropriate set of tied-states. This resulted in 8,453 distinct triphones. Eight component mixture distributions were then estimated using embedded Baum-Welch re-estimation for each of the speech states producing a system with approximately 2.3 million parameters. Recognition experiments with this gender-independent (`gi`) word-internal (`wint`) system were performed for the 5k task and used the static network decoder with a bigram (`bg`) grammar.

The results for the `wint/gi` system are shown in the first line of Table 1 for a number of different test-sets: `Nov'92` was the non-verbalised 5k closed test set used in the November 1992 ARPA WSJ evaluation (330 sentences from 8 speakers); `si_dt_s6` is the Sennheiser data from the WSJ1 spoke 6 development test data (202 sentences from 8 speakers); `si_dt_05.odd` is a subset of the WSJ1 5k development test data formed by deleting sentences with out-of-vocabulary (OOV) words and choosing every other sentence that remained (248 sentences from 10 speakers); and finally `Nov'93` is the Hub 2 5k test data from the ARPA November 1993 WSJ evaluation (215 sentences from 10 speakers). The first three sets were used as development test data and the final set used as evaluation test. It should be noted that there is quite a range of word error rates over the different test sets, and also that the Nov'93 results correspond to the preliminary results released by NIST on December 8th 1993. We expect the final error rates for all Nov'93 systems reported here to be 0.1-0.2% lower due to corrections in the reference transcriptions.

4.3. SI-84 Cross-Word System

To try to improve the `wint/gi` system we investigated the use of position independent cross-word (`xwrd`) triphones. In this case all word-boundary information is ignored for the purpose of identifying triphone contexts. The SI-84 training set contains 18,512 cross-word triphones (55,530 states). Model building proceeds in a similar fashion to the `wint` system—the tree clustering reduced the number of tied-states to 3,820, and then models for all the possible cross-word triphones that can occur in the 20k recognition vocabulary (54,456 triphones) were systhesised resulting in 15,303 distinct triphones. Eight component mixture distributions were then trained to produce a system with about 2.4 million parameters. This `xwrd/gi` system was tested using the single pass dynamic network decoder for the 5k bigram task. The results are given in the second line of Table 1. It can be seen that performance is between 8% and 15% better than the `wint/gi` system on the development test data and gave a 33% improvement on the evaluation test. It appears that the use of cross-word triphones is of greater benefit for faster speaking rates and speech that is less clearly articulated: the variations in speech style between different speakers accounts for the fairly substantial variations in error rate reductions observed.

4.4. SI-84 Gender-Dependent System

The `xwrd/gi` system was cloned, and the means and mixture weights of the two model-sets re-trained (one iteration only)—one set on the data from the female talkers (3635 sentences) and the other set on the data from the male talkers (3558 sentences) to form a gender dependent (`gd`) system. The variance vectors are not retrained, and were shared using HTK's generalised tying mechanism between the model sets to save memory. Therefore in total the system has about 3.6 million parameters. The two model sets were then decoded in parallel using the single-pass dynamic network recogniser with the constraint that all active paths could be extended only by using models of the same gender. The results in Table 1 show that the **xwrd/gd** system gave small (between 4% and 8%) reductions in word error for the development test sets. However, the Nov'93 evaluation test data showed a small increase (2%) in error when gender dependent models were used. However, the 8.83% error given by this SI-84 **xwrd/gd** system was still the lowest error rate from any site for the H2-C1 test in the November 1993 evaluation (WSJ0 training data only, standard bigram grammar).

4.5. SI-284 System

The same techniques used to build the SI-84 cross-word gender dependent system were again used to build a model set using the complete 66 hours of speech data in the SI-284 training set. Since the size of the training corpus is nearly a factor of five larger, models with more parameters can be trained. In the SI-284 data set there are 22,699 cross-word triphones (68,097 states) and these were tree-clustered to 7,558 tied states. After all 54,457 triphones potentially needed for recognition had been synthesised there were 22,978 distinct triphones. Ten component mixture distributions were estimated for each of the tied-states, and then the system was cloned and the models retrained using gender-specific training data (18127 sentences from male talkers, 18,388 sentences from females). Again the variances were not re-estimated and were tied between the two model sets. In total this system has about 8.9 million parameters.

When this system is used with the 5k bigram grammar, reductions in error rate of between 22% and 27% are obtained relative to the SI-84 **xwrd/gd** system. This system was used with a 5k trigram grammar for the November 1993 evaluation and the error rate of 4.99% on the evaluation data was the lowest error rate for the H2-P0 test (any acoustic training data, any grammar). Over all the test sets, it can be seen from Table 1 that use of the trigram produces improvements of between 20% and 38% relative to the bigram error rate due to an approximate halving in test-set perplexity. When used with the 5k trigram, the single-pass dynamic network decoder typically requires 5 minutes per sentence on an HP735 computer.

4.6. 20k Results

The SI-284 system described in Sec. 4.5. was also used with the full 64k test sets. However in this case the standard 20k open bigram and trigram grammars were used. Therefore there are a number of out-of-vocabulary (OOV) words in each set which cause errors. Three 64k test sets have been used: `Nov'92`–November 1992 20k open NVP set (333 sentences from 8 speakers, 1.9% OOV words); `si_dt_20.odd`–formed by taking every other sentence from WSJ1 Hub 1 development test data (252 sentences, 10 speakers, 1.8% OOV words); and `Nov'93` which consists of the November 1993 H1 evaluation test data (213 sentences, 10 speakers, 1.8% OOV words). The results on these test-sets using both bigram and trigram grammars are shown in Table 2: 14.45% error rate for the November 1993 evaluation with the standard 20k bigram grammar was the lowest error rate reported for H1-C2 (WSJ0 & WSJ1 training, 20k bigram grammar); and the 12.74% word error rate was the second lowest for H1-C1 (WSJ0 & WSJ1 training, 20k trigram grammar). The single pass dynamic network decoder typically takes about 10 minutes to recognise a sentence using the 20k trigram on an HP735 computer.

Grammar	Nov'92	si_dt_20.odd	Nov'93
bg	11.08	16.17	14.45†
tg	9.46	13.71	12.74†

Table 2: % word error rates for SI-284 xrwd/gd 20k system for different 64k WSJ test-sets. † denotes systems used for the ARPA November 1993 WSJ evaluation.

It can be seen from Table 2 that there is between 12% and 15% reduction in error rate obtained by using the 20k trigram relative to the bigram. This is a rather smaller relative decrease that for the 5k case and is caused by a smaller reduction in test-set perplexity and by the effects of OOV words. On average each OOV word in the test set causes about 1.6 word-errors, and hence the OOV words make a significant contribution to the word error rate.

5. CONCLUSION

This paper described our approach to speaker independent large vocabulary continuous speech recognition using HTK. The system uses tied-state, mixture density, cross-word gender dependent triphones and recognition is performed using a single pass dynamic network decoder directly incorporating either bigram or trigram language models. The results show that this approach yields state-of-the-art performance on both the 5k and 20k word Wall Street Journal tasks.

ACKNOWLEDGMENTS

The WSJ pronunciations were provided by Dragon Systems Inc., the WSJ speech data by the LDC, and the language models by Lincoln Labs. This work is in part supported by an SERC/MOD research grant, number GR/J10204.

REFERENCES

[1] Hwang M-Y., Huang X. & Alleva F. (1993). Predicting Unseen Triphones with Senones. *Proc. ICASSP'93*, Vol II, pp. 311-314, Minneapolis.

[2] Ney H., Haeb-Umbach R, Tran B-H. & Oerder M. (1992). Improvements in Beam Search for 10000-Word Continuous Speech Recognition. *Proc. ICASSP'92*, Vol I, pp. 9-12. San Francisco.

[3] Woodland P.C. & Young S.J. (1992) Benchmark DARPA RM Results with the HTK Portable HMM Toolkit. *Proc. DARPA Continuous Speech Recognition Workshop*, September 1992, Stanford.

[4] Woodland P.C. & Young S.J. (1993) The HTK Tied-State Continuous Speech Recogniser. *Proc. Eurospeech'93*, Berlin.

[5] Young S.J. (1992). The General Use of Tying in Phoneme-Based HMM Speech Recognisers. *Proc. ICASSP'92*, Vol I, pp. 569-572. San Francisco.

[6] Young, S.J., Woodland P.C. & Byrne W.J. (1993). HTK Version 1.5: User, Reference & Programmer Manual. Cambridge University Engineering Department & Entropic Research Laboratories Inc., September 1993.

[7] Young S.J. & Woodland P.C. (1993). The Use of State Tying in Continuous Speech Recognition. *Proc. Eurospeech'93*, Berlin.