CONTEXT DEPENDENT PHONE MODELS FOR LSTM RNN ACOUSTIC MODELLING

Andrew Senior, Haşim Sak, Izhak Shafran

Google Inc.,
New York
{andrewsenior,hasim,izhak}@google.com

ABSTRACT

Long Short Term Memory Recurrent Neural Networks (LSTM RNNs), combined with hidden Markov models (HMMs), have recently been show to outperform other acoustic models such as Gaussian mixture models (GMMs) and deep neural networks (DNNs) for large scale speech recognition. We argue that using multi-state HMMs with LSTM RNN acoustic models is an unnecessary vestige of GMM-HMM and DNN-HMM modelling since LSTM RNNs are able to predict output distributions through continuous, instead of piece-wise stationary, modelling of the acoustic trajectory. We demonstrate equivalent results for context independent whole-phone or 3-state models and show that minimum-duration modelling can lead to improved results. We go on to show that context dependent whole-phone models can perform as well as context dependent states, given a minimum duration model.

Index Terms— Hybrid neural networks, hidden Markov models, Long Short-Term Memory Recurrent Neural Networks, context dependent phone models.

1. INTRODUCTION

Deep neural networks (DNNs) have been successful for acoustic modeling in large vocabulary speech recognition [1]. More recently, Long Short-Term Memory (LSTM) recurrent neural networks (RNNs) have been shown to beat state-of-the-art DNN systems [2, 3, 4]. LSTMs [5, 6] are a type of recurrent neural network, which contain special units called *memory blocks* in the recurrent hidden layer, and which are often easier to train than standard RNNs. The memory blocks contain memory cells with self-connections storing the temporal state of the network. In addition, they have multiplicative units called gates to control the flow of information into the memory cell and from the cell to the rest of the network.

Both DNNs and LSTMs are commonly used as probability estimators and in speech recognition, the probabilities are used to compute the likelihood of some acoustic data, given word sequences, in a hidden Markov model. This is a so-called "hybrid" use of neural networks. By searching through a weighted search graph of word sequences, implemented as a finite state automaton, the maximum likelihood word sequence can be found. Typically the probabilities are estimated for a set of acoustic units which correspond to the states of the HMM. These acoustic units are produced by a clustering based on the context — the phonemes preceding and following the units.

In this paper we reexamine how these acoustic units are chosen, and show that we can achieve comparable results with a simpler HMM model, provided that we introduce a simple duration model. Section 2 describes the Hybrid HMM-LSTM models we use and describes context dependency (Section 2.2) and duration modelling (Section 2.3). Section 3.2 describes our system architecture and data, while Section 4 describes initial experiments with context-independent (CI) models and futher experiments with duration modelling and CD phone models. The final section summarizes the experiments and describes future work.

2. HYBRID LSTM-HMM ACOUSTIC MODELS

DNNs and LSTM RNNs for acoustic modeling have commonly used the hybrid approach [7], where the neural networks estimate the posterior probabilities $p(s_i|x_1, \ldots, x_i)$ of acoustic states s_i given part of a sequence of T feature vectors $X = x_1, \ldots, x_T$. A hidden Markov model decoder finds the most likely sequence of states through a search graph by combining the scaled posteriors $p(s_i|x_1, \ldots, x_i)/p(s_i)$ for individual frames with the language model probability $p(s_1, \ldots, s_N)$.

These hybrid neural network models use a softmax output layer which converges to estimate class posteriors when trained with a cross-entropy loss. They are generally trained with targets from an alignment. Alignments can be obtained by forced alignment of the supervised transcript with the acoustic sequence using any existing model, including one that has been"flat-started' [8] without alignment information.

2.1. HMM States

The simplest form of acoustic model uses a single HMM state per phone, as show in Figure 1(c). Because of the temporal variation within a phone, it is common to split a phone into more than one state, usually three, as shown in Figure 1(b), whose probabilities are modelled separately by the acoustic model.

Transitions in the hidden Markov model are restricted to allow only left-to-right transitions in the model, effectively dividing the phonetic unit into a set of states which must be traversed in sequence, with optional repetitions, each state having a stationary probability distribution. While there has been previous work in which the HMM topology or number of states is varied, the majority of recent work, particularly that using deep neural networks, uses the 3-state left-to-right models shown in Figure 1(a). The three-state, piecewise-stationary model is a parsimonious and effective simplification that was hard to beat with more complex models of the non-stationarity of the acoustic frames from a phone. We have previously used the same HMM topology with LSTM acoustic models [4]. Throughout this work we use HMM states with self-loops and transitions to the next state.

The independent processing of acoustic frames in GMMs and DNNs means that the distribution for each acoustic state is the same for all frames in that state — the posterior is conditioned only on

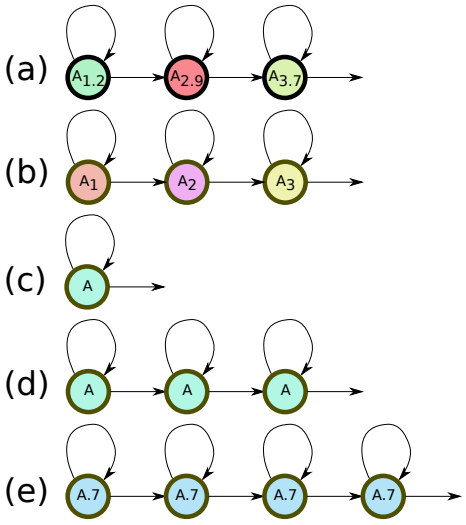

Fig. 1: Simple left-to-right HMM topologies. (a) A conventional 3-state CD HMM. (b) A 3-state CI HMM (c) A one-state phone HMM (d) A tied-state CI phone model with minimum duration of 3. (e) A tied-state CD phone model with minimum duration of 4.

the current feature vector. However, in a recurrent network the distribution for each frame of a state is different, being dependent on the internal state of the RNN, so we would argue that there is no need for the modelling of three distinct output distributions for each phoneme. Temporal variation of the distribution can already be captured by the RNN dynamics, so a single output label per phone should be sufficient.

2.2. Context dependent state tying

It has long been known that, because of coarticulation effects, the acoustic realisations of phonetic units depend on their context and in particular the phonemes that precede and follow them. To achieve greater modelling power, *context-dependent* units were proposed, in which states in different contexts are modelled separately [9]. Because of the large number of possible contexts (N^2 contexts for triphone units with N phones, leading to $3 \times N^3$ possible units for 3-state HMMs), context dependent modelling is only feasible by clustering similar contexts and treating them identically, resulting in context dependent state tying.

Young *et al.* [9] described one way to cluster similar context-dependent states. This algorithm takes force-aligned feature vectors, collecting together all those vectors aligned to a particular CI state, and computing sufficient statistics on each subset with a particular phonetic context. Now, for each CI state a decision tree is built by binary divisive clustering. At each node of the tree a set of binary phonetic questions is posed about the state's neighboring phones. Each such question partitions the data in two, and from the sufficient statistics a Gaussian model can be estimated for each partition. The tree is extended by choosing the question which leads to the greatest likelihood gain. Tree building terminates when the likelihood gains are below a threshold, or when the leaves have too few observations.

By terminating the tree-building earlier, we can derive smaller inventories of context dependent states. Throughout our experiments, for CD state clustering and model training, we use the same training set and state boundaries given by a CD-DNN model. Since the state inventories from truncated tree-building are nested, a simple many-to-one mapping can be applied to the original alignment labels to train with these smaller inventories.

In this work, we modify the algorithm in three ways. First, instead of clustering with one tree per CI state, we build one tree per phone. Second, since we wish our acoustic states to model whole-phone *trajectories* of acoustic features rather than piecewise stationary periods of acoustic features, instead of clustering all the frames assigned to each phone we make a single representative feature vector for each example of the phone in the training set. A simple feature vector is constructed by concatenating the central frame from each state of the 3-state frame alignment (the alignment in this case comes from a previously-trained 3-state CI DNN system.) Third, following our previous work [8], we investigate changing the speech representation used for clustering. Previously we compared conventional (PLP) acoustic features with temporal differences; filterbanks with and without temporal differences; and the activations of a DNN ASR model's penultimate layer.

In this work we begin by clustering with DNN activation features which results in 8367 CD phone models. The baseline model obtained by clustering CI states using PLP features has 13522 CD states. The static CLG FSTs that result from the different C transducers are approximately the same size.

2.2.1. *Clustering on LSTM state*

Since we argue that the LSTM is modelling the acoustic trajectory throughout each phone, then it also seems natural that the LSTM state should be a good representation of that trajectory. Thus we repeat the same clustering algorithm using vectors of LSTM state from a previously trained two-layer LSTM model. Each phoneme in the training set is represented by the second LSTM layer's state for the final frame of that phoneme, which is 800 dimensional. Clustering in this case results in 8491 CD phones.

2.2.2. *Clustering right context*

We note that using the connectionist temporal classification (CTC) algorithm [10] with *bidirectional* LSTMs has shown good results on whole-phone models [11, 12] without the need for context-dependent modelling. We argue that bidirectionality provides the model with evidence for the acoustic context and thus the LSTM model itself is modelling the distribution given the context, in the same way that choosing a context-dependent unit on the basis of the search graph conditions the distribution on the context. Since we have a unidirectional model which is aware of the left, but not right, acoustic context, we investigated the effect of phone clustering based only on the right phonetic context. For these experiments, we again used the LSTM state features, and clustering resulted in 1120 CD phone units. Note that the maximum number of right-context dependent CD phone units when silence is not made context dependent is 1641 for 41 phones.

2.3. Duration modelling

Several previous researchers have investigated the use of duration models for speech recognition [13]. Typically these are used to assign transition probabilities to the HMM to match the empirical distribution of durations observed in the training set. Our baseline model had no duration model — using probabilities of $\frac{1}{2}$ for both arcs leaving each state. Nevertheless the three-state left-to-right topology imposes a minimum duration of 3 frames per phone since it requires that at least one frame is emitted by each state.

With a single state per phone, the minimum phone duration becomes a single frame, which allows poorly-matching phones to be

passed through with minimal cost. To prevent this, we apply a simple minimum duration. In our system this is simple to do by making a multi-state HMM but tying the distributions of these states, as shown in Figure 1(d). For direct comparison we initially use 3-state HMMs but observe that it is simple to make the minimum duration phone-dependent by varying the number of state replicas. With context dependent phone-modelling we can make the duration dependent on the particular CD phone, as shown in Figure 1(e).

Duration histograms can be computed from the training set. Figure 2(top) shows the cumulative histograms for the phone models. Every phone instance has a duration of 3 or more states because the alignment was done with a 3-state CD HMM, but it can be seen that the observed distribution is quite different for different phones. Thresholding the cumulative probability (we found a threshold of 10% of the probability mass to give the best results in initial tests), we arrive at a minimum duration for each phone or CD-phone (shown in Figure 1(e)), though for the special case of silence we continue to use a 3-state minimum duration. Figure 2(top) shows that there is considerable variation in the observed durations of different context dependent variants of a single phone.

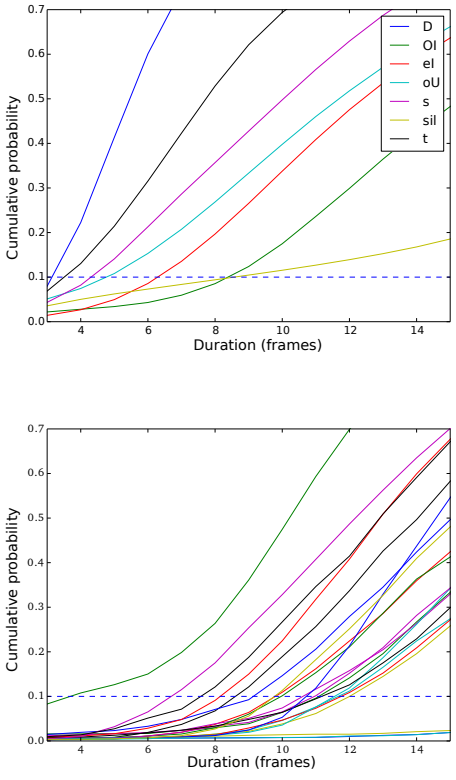

Fig. 2: Cumulative duration histograms for selected models measured on the training set alignments. The dashed line shows the 10% threshold used here to determine a minimum duration threshold. Top: whole-phone durations. Bottom: The 21 different CD variants of the "OI" phone from the 8491-CD inventory show considerable diversity.

3. EXPERIMENTAL SET-UP

3.1. LSTM RNN

The LSTM network used in this paper is adopted from our previous work [4] and uses the conventional hyperbolic tangent activation (tanh) for the cell input units and cell output units, and logistic sigmoid for the input, output and forget gate units. A final output layer has a softmax activation function. We use a two layer deep LSTM RNN, where each LSTM layer has 800 memory cells and a dimensionality-reducing linear recurrent projection layer of 512 linear units. The network has a total of 13 million parameters. We use a unidirectional model because of the low-latency requirement of our interactive task. With a unidirectional model we can stream computation and decoding results while a query is being uttered.

The input to the LSTM at each time step is a single frame of 40-dimensional log-mel filterbank features. Since information from future frames helps making better decisions for the current frame (similar to having a right context window in DNNs), we delay the output HMM state label by 5 frames.

The LSTM networks are trained with a cross-entropy loss, using asynchronous stochastic gradient descent (ASGD) [4] using distributed training with 300 tasks scheduled on different machines, each working through a partition of the randomly shuffled training utterances. Each task processes four utterances at a time, using the back propagation through time algorithm to forward propagate and then backpropagate for 20 consecutive frames. Each task thus computes a parameter gradient update for a minibatch of 4×20 frames. More details of LSTMs and training with ASGD can be found in an earlier work [4].

3.2. Training & Evaluation

All the networks are trained on a 3 million utterance (about 1700 hours) dataset consisting of anonymized and hand-transcribed 8kHz US English Google voice search and dictation traffic. The dataset is represented with 25ms frames of 40-dimensional log-filterbank energy features computed every 10ms. The 40-dimensional features are input to the network with no stacking of frames. The utterances are force-aligned with an 85 million parameter DNN to generate fixed labels for training. The weights of all layers are randomly initialized prior to training. We try to set the learning rate specific to a network architecture and its configuration to the largest value that results in a stable convergence. The learning rates are initially held constant and then decayed exponentially during training. A small amount of ℓ_2 regularization was used throughout training.

The trained models are evaluated in a large vocabulary speech recognition system on a test set of 22,500 hand-transcribed utterances and the word error rates (WERs) are reported. The language model used in the first pass of decoding is a 5-gram language model heavily pruned to 23 million n-grams with a 2.2 million word vocabulary. In a second pass, the word lattices output from the first pass are rescored with a 5-gram language model having 15 billion n-grams.

4. EXPERIMENTS

Our first experiment investigates the need for dividing each phone into 3 states modelled separately by the LSTM using context independent models.

We trained two LSTM acoustic models using the same alignments given by forced-alignments with a 14,000 CD state DNN. The

first LSTM has 126 softmax outputs corresponding to the context independent states of an HMM with 3 states per phone (mapping the CD labels of the alignment to the corresponding CI state). The second LSTM has 42 output states, one per phone, after mapping the alignments to the corresponding phone. These models are used for decoding with a simple HMM that has one state per phone (Figure 1(b) and (c) respectively). Results are shown in Table 1. We first observe that this phone model performs worse than the CI model. However, in changing the granularity of the acoustic model we have also changed the number of states per phone. This means that the minimum number of frames that must be expended in each phone has changed from 3 to 1, which by itself impacts recognition accuracy. By representing each phone with a 3-state HMM with tied distributions (Figure 1(d)), we can use the phone acoustic model but retain the minimum duration constraint, and achieve a similar WER, as shown on the last line of Table 1. The remaining experiments investigate the use of context dependency and were trained with a different phone set and different alignments to the context-independent experiments.

Table 1: Word Error Rates of context-independent models. A 14000 state context dependent model trained with the same alignments achieves 10.7% WER.

Model	WER (%)
126-state CI model	16.5
42-state phone model	20.0
42-state phone model with minimum duration 3	16.4

4.1. Duration modelling

Table 2 shows the effect of different duration models when testing the 8397-state CD phone LSTM acoustic model. It can again be seen that imposing a minimum duration is essential for good performance, with the best performance for a fixed minimum duration when every model has duration 4. Setting a minimum duration per phone gives better results, but the best performance is found when the minimum duration is chosen separately for each CD-phone model.

4.2. Context dependent clustering

We note that the performance of the context independent model is significantly worse than the performance of a context dependent model (16.5% WER vs 10.7%), so we next investigate whether we can make context-dependent whole-phone models in the same way as we use context dependent HMM states in our baseline model.

Table 3 compares the three different features from Section 2.2 for clustering CD-phones (based on DNN-activation, LSTM state or LSTM-state with right-context only) with the standard CD-state inventory. In each case we compare different sizes of state inventory by early-termination of the clustering.

We first observe that the whole-phone CD models with 8367/8491 states perform as well as the conventional CD state model with 13522 states (and better than the CD state model with 8000 states). For smaller state inventories the performance is roughly comparable. We observe that the two feature types we have used for CD phone tree building result in similar performance. It appears not to be sufficient to only cluster on the right context, although the state inventory is small (1120) it represents 70% of the possible right-clustered diphone units possible.

Table 2: Word error rates for CD phone models with different minimum-duration models, using the 8397-CD-phone LSTM acoustic model.

Duration	WER
1 state	12.3
3 state	10.4
4 state	10.2
5 state	10.3
Per-phone	10.1
Per-CD phone	10.0

Table 3: Clustering on LSTM state vs DNN features. In each CD phone evaluation we use the per-CD-phone minimum duration model.

States	CD phones			CD states
	DNN Activations	LSTM State	Right only	
500	12.0	12.4	12.1	12.1
1120	–	–	11.7	–
2000	11.6	11.3	–	11.0
~8300	**10.0**	**10.0**	–	10.5
13522	–	–	–	10.1

5. CONCLUSIONS AND FURTHER WORK

In this paper we have shown that the conventional multi-state phone model used with GMMs and DNNs is not necessary with LSTM acoustic models. We have shown that, with a simple duration model, a context-dependent triphone model can equal the performance of a 3-state context-dependent triphone model. This reduces the number of states that must be modelled and consequently the number of parameters and acoustic model computational burden.

Since we have shown the importance of even a simple minimum duration model, we plan to investigate stronger modelling of the CD-phone duration distributions. The models described here were all trained on the same DNN-based alignment. One or more iterations of realignment and retraining of both the LSTM and the duration model may result in a more consistent and thus more accurate model.

Further, we have recently [14] shown improvements in WER (around 10% relative) from sequence training [15] of LSTM acoustic models. We have still to investigate whether such gains can also be demonstrated for these CD phone models. Finally, we plan to investigate whether these context dependent models can be used in conjunction with the CTC algorithm [10] that has hitherto only been used with context independent whole-phone models, but which can nevertheless achieve word error rates close to those from conventional CD state models [12].

6. REFERENCES

[1] G. Hinton, L. Deng, D. Yu, G.E. Dahl, Mohamed A., N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T.N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Processing Magazine*, vol. 29, pp. 82–97, November 2012.

[2] A. Graves, N. Jaitly, and A.-R. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *Automatic Speech Recognition and Understanding (ASRU), IEEE Workshop on*. IEEE, 2013, pp. 273–278.

[3] H. Sak, A. Senior, and F. Beaufays, "Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition," *ArXiv e-prints*, Feb. 2014.

[4] H. Sak, A. Senior, and F. Beaufays, "Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition," in *Interspeech*, 2014.

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[6] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, 2000.

[7] H. Bourlard and N. Morgan, *Connectionist speech recognition*, Kluwer Academic Publishers, 1994.

[8] A. Senior, G. Heigold, M. Bacchiani, and H. Liao, "GMM-free DNN training," in *Proc. ICASSP*, 2014.

[9] S. Young, J. Odell, and P. Woodland, "Tree-based state tying for high accuracy acoustic modelling," in *Proc. ARPA Human Language Technology Workshop*, 1994.

[10] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, Springer, 2012.

[11] A. Graves, A. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. ICASSP*, 2013.

[12] H. Sak, A. Senior, K. Rao, O. Irsoy, A. Graves, F. Beaufays, and J. Schalkwyk, "Learning acoustic frame labeling for speech recognition with recurrent neural networks," in *submitted to ICASSP*, 2015.

[13] M. Lehr and I. Shafran, "Learning a discriminative weighted finite-state transducer for speech recognition," *IEEE Transaction on Audio, Speech and Language Processing*, vol. 19, no. 5, pp. 1360–1367, July 2011.

[14] H. Sak, O. Vinyals, G. Heigold, A. Senior, E. McDermott, R. Monga, and M. Mao, "Sequence discriminative distributed training of long short-term memory recurrent neural networks," in *Interspeech*, 2014.

[15] B. Kingsbury, "Lattice-based optimization of sequence classification criteria for neural-network acoustic modeling," in *Proc. ICASSP*, Taipei, Taiwan, Apr. 2009, pp. 3761–3764.