

WORD-SELECTIVE TRAINING FOR SPEECH RECOGNITION

Teresa M. Kamm and Gerard G. L. Meyer

Center for Language and Speech Processing
Department of Electrical and Computer Engineering
The Johns Hopkins University
Baltimore, Maryland 21218, USA
tkamm@jhu.edu, gglmeyer@jhu.edu

ABSTRACT

We previously proposed [1, 2] a two-pronged approach to improve system performance by selective use of training data. We demonstrated a sentence-selective algorithm that, first, made effective use of the available humanly transcribed training data and, second, focused future human transcription effort on data that was more likely to improve system performance. We now extend that algorithm to focus on word selection, and demonstrate that we can reduce the error rate from 10.3% to 9.3% on a simple, 36-word corpus, by selecting 30% (15 hours) of the 50 hours of training data available in this corpus, without knowledge of the true transcription. We also discuss application of our word selection algorithm to the Wall Street Journal 5K word task. Preliminary results show that we can select up to 60% (48 hours) of the training data, with minimal knowledge of the true transcription, and match or beat the error rate of a system built using the same amount of randomly selected training data.

1. INTRODUCTION

According to conventional wisdom, incorporating more training data is the surest way to reduce the error rate of a speech recognition system. The obvious way to increase the amount of training data is to acquire more speech data and have it transcribed, but this is expensive due to the high cost of annotating data.

A second way to acquire more training data, without incurring annotation cost, is to incorporate automatically transcribed data into the training set. An existing recognition system is used to automatically transcribe a large amount of available, but untranscribed, speech data. A selection criterion is applied to determine a subset of the automatically transcribed material to be combined with the training data to create a larger training set. A new system

is trained and its error rate is observed. It has been demonstrated that it requires twice as much automatically transcribed data to observe the same reduction in error rate as that realized by humanly transcribed data [3-5].

As an alternative, we proposed in [1, 2] a two-pronged approach to improve system performance by manipulation of the training set. First, we make effective use of the available humanly transcribed data, and second, we focus future human effort on data that is more likely to improve performance.

This approach utilizes an *active selection learning* method that allows the system/learner to exert control over what new data is introduced into training. In particular, the learner seeks new examples that it anticipates will have a positive impact when incorporated into training. This approach has the potential to select a set of data from which to build a recognition system that outperforms a system built on larger, but randomly selected, data.

We showed in [2] a successful active selection algorithm for sentences. By sentence, we mean a complete utterance, as given in the distributed corpus, without further segmentation. On a simple acoustic corpus, our method was shown to reduce the error rate from 10.3% to 9.5% under supervised conditions by selecting 35% (17.5 hours) of the training data, focusing on sentences with high error and low confidence. Under unsupervised conditions, our method was able to select 25% (12.5 hours) of the training data, without knowledge of the true transcription, which when transcribed resulted in a system matching the error rate of one built using the entire 50 hours of data. In that case, our active selection method focused on low confidence sentences. Active selection was successful on this corpus because the sentences were well behaved; that is, each was roughly 3-5 seconds long and the selection criterion did not have to factor in sentence length.

Here we focus on active word selection methods, with the goal of applying this to more realistic speech problems. We start by modifying our selection method to focus on words, and demonstrate a reduction in error rate from 10.3% to 9.3% on our simple, 36-word corpus, by

selecting 30% of the training data available, without knowledge of the true transcription.

We then turn our attention to the more difficult 5K word Wall Street Journal (WSJ) task. We investigate the relation between training size and error rate and establish baseline error rates across a range of training sizes. We then propose a confidence measure suitable for use in our active selection method on this corpus. Preliminary results show that we can select up to 60% (48 hours) of the data, with minimal knowledge of the true transcription, and match or beat the error rate of a system using the same amount of randomly selected data.

The organization of this paper is as follows. Section 2 introduces notation and the active learning paradigm. Section 3 discusses the simple acoustic corpus and results when applying the selective training algorithms. Section 4 discusses the 5K word WSJ task and results when applying the word-selective training algorithm. Finally, a summary and discussion are presented in section 5. The training and testing of the systems described generally follow the procedures in the HTK documentation [6].

2. ACTIVE LEARNING PARADIGM

In this section, we present a reformulation of the maximum likelihood training criterion to accommodate selection of training data, and present the active selection criterion used.

2.1. Selective Training

We start with a sequence of acoustic observations, $O = o_1 \cdots o_i \cdots o_T$, over time $t = 1, \dots, T$. The acoustic model, λ , is determined by the maximum likelihood training criterion:

$$\lambda = \arg \max_{\lambda'} P(O|W; \lambda') \quad (1)$$

where W are the words associated with the observations O . This formulation assumes one string of observations.

Now, suppose we partition the observation string O into utterances, $U = u_1 \cdots u_N$, where N is clearly less than or equal to T , and partition the associated word labels as well, $W = W_{u_1} \cdots W_{u_N}$, such that W_{u_i} are the words associated with the utterance u_i , for $i = 1 \dots N$. We can now rewrite the maximum likelihood training criterion in terms of a set of utterances:

$$\lambda = \arg \max_{\lambda'} \prod_{i=1}^N P(u_i | W_{u_i}; \lambda'). \quad (2)$$

Finally, by applying a selection method to determine a subset U' of U :

$$U' = \{u: u \in U, \text{ and } u \text{ obeys some constraint}\} \quad (3)$$

Given a set of candidate training data U ,

1. Select an initial training set U_0 from U ; and let $i = 0$.
2. Obtain labels for the training set U_i .
3. Train a model, $\lambda_i = \arg \max_{\lambda} \prod_{u \in U_i} P(u|W_u; \lambda)$.
4. Accumulate additional training data,

$$U_{i+1} = U_i \cup \{U': U' \subset (U - U_i)\}.$$

5. Go to step 2.

Figure 1: Generalized Iterative Selective Training Paradigm

we define a generalized iterative selective training procedure. This procedure starts with a subset U_0 of the candidate training set U , deems it the training set, and then iteratively selects, accumulates and utilizes additional training from the remaining candidate set, as given in Figure 1.

2.2. Active Selection Criterion for Acoustic Training

What constraints the subset U' should meet are essential to achieve our goal of reducing error rate. In particular, we focus on active selection criteria that allow the learner to exert some control over what new data is used.

In the case of our prior sentence selection algorithm, we started with a sequence of observations partitioned into sentences and applied the current model λ_i to hypothesize the most likely words \hat{W} for a given sentence S :

$$\hat{W} = \arg \max_w P(W)P(S|W; \lambda_i). \quad (4)$$

Then, for each hypothesized word w_h in \hat{W} and associated acoustic observation (i.e. sentence segmentation) S_{w_h} , we generated the word posterior probability:

$$P(w_h | S_{w_h}; \lambda_i) = \frac{P(S_{w_h} | w_h; \lambda_i)}{\sum_{\text{all } w} P(S_{w_h} | w; \lambda_i)} \quad (5)$$

Next, we determined sentence level confidence by taking the average of the hypothesized word posterior probabilities for each sentence. Finally, we applied a selection criterion choosing the set of sentences with lowest confidence that cumulatively represented a fixed number of acoustic observations (roughly 3-5% of the original training set).

We now adapt this method to focus on words. For each hypothesized word w_h in \hat{W} and the associated sentence segmentation S_{w_h} , we compute the posterior probability as given in (5), and apply a corpus-dependent normalization. We then select the set of words with lowest confidence (i.e. normalized posterior probabilities) that cumulatively represent a fixed number of acoustic observations. Following the iterative training procedure in Figure 1, step 4, we now must acquire the true transcription as well as the *true word segmentation* in the region of our selected words. We then extract the acoustic observations associated with the true word that most closely aligns with the selected, low confidence, hypothesized word.

3. THE OGI ALPHADIGIT CORPUS

The OGI alphasdigit corpus [7], comprised of nearly 75 hours, is a collection of over 3000 subjects speaking strings of 6 alphasdigits over the telephone. The alphasdigits are the English letters "A" through "Z" and the digits "0" through "9". The speakers were prompted to speak either 19 or 29 sets of 6 alphasdigits. Researchers at ISIP [8] have defined a standard train/test partition of this corpus, and have reported error rates achieved on this corpus [9].

From the ISIP training partition of 51545 sentences, 46730 sentences (approximately 50 hours) were selected such that the transcription matches one of the prompts given in the prompt list [7]. This was an inexpensive way to remove possible transcription substitution errors from the training set. In this paper, this training set is denoted as U . We selected our test set (3112 sentences, 3.26 hours) from the ISIP evaluation test partition of 3329 sentences by using the prompt list in the same way.

3.1. System Description and Baseline Error Rates

Our system consists of a word model for each alphasdigit, plus silence and short pause. The silence/short pause models are built according to the procedure in the HTK documentation [6]. Each word model uses a standard left-right topology including a re-entrant transition, with the number of states based on one-half the mean duration of the word.

The word durations are determined by first training a system with each word model having 10 states. Then a forced alignment of the models to the training data is generated and the word duration statistics are computed from this forced word alignment. All systems discussed in this section have a total of 825 states.

The features used are 12 mel-frequency cepstral coefficients plus energy, the deltas, and the double deltas to make a feature vector of length 39. In all systems discussed in this section, 12 gaussians are estimated per feature per state.

Table 1. Baseline error rates for selected training set sizes of the OGI alphasdigit corpus.

| Training Size (hours) | Error Rate (%) |
|-----------------------|----------------|
| 2.5 | 14.3% |
| 5 | 11.8% |
| 12.5 | 11.0% |
| 25 | 10.4% |
| 50 | 10.3% |

An equal probability word network is used to drive recognition. This network is defined as: optional silence, followed by one or more alphasdigits with optional inter-alphasdigit silence, followed by optional silence.

Baseline error rates are given in Table 1 for various sizes of balanced sets of training data (see [1] for further details).

3.2. Sentence-Selective Criterion and Results

Here we review the results achieved in [1, 2]. Following the generalized iterative training algorithm, as in Figure 1, we started with a system trained with 2.5 hours of the candidate set U . We applied that system to the remaining candidate data to generate a hypothesized word string for each sentence and a sentence confidence (the average per word posterior probability of the hypothesized words).

We then selected additional data, equal in size to the 2.5-hour initial set, in one of two ways. In the supervised case, we observed the number of errors in each sentence and we selected the set of sentences that had the most error. In the case of a tie, where multiple sentences had the same error rate, we chose the sentences with the lowest confidence. In the unsupervised case, we selected the set of sentences with the lowest confidence.

We joined the newly selected data with the initial set, built a new system, and observed the error rate of the new system. The selective training algorithm was iterated until all of the candidate data was exhausted. For the supervised case, our sentence-selective training method was shown to reduce the error rate from 10.3% to 9.5% by selecting 35% (17.5 hours) of the 50-hour candidate set. Under unsupervised conditions, our method was able to select 25% (12.5 hours) of the 50-hour candidate set, without knowledge of the true transcription, which, when used, resulted in a system that matched the error rate of one using all of the training data.

3.3. Word-Selective Criterion and Results

Following the generalized iterative training algorithm, given in Figure 1, we start with a system trained with 2.5-hours of the candidate set U . We apply that system to the remaining candidate data, generating a hypothesized word string for each sentence. For each hypothesized word, we

generate a confidence score (the normalized word posterior probability). It is normalized by observing the cumulative probability distribution of posterior probabilities for each possible word and computing the probability of observing the computed posterior probability.

We select additional training, approximately 3% of the total training, which is the set of hypothesized words with the lowest confidence. Under normal unsupervised circumstances, we would have a human correct the word hypothesis and designate the word boundaries. Instead, we use the true transcription to simulate this step by mapping the hypothesized words onto the true transcription by a simple one-to-one mapping based on word order. Then we determine the word boundaries and discover between word silences by a forced alignment of the true transcription and the current model.

Continuing, we extract the observations associated with each of the selected words, thereby generating new training utterances along with their transcription. In the case where two or more adjacent words are selected, these are joined into one utterance, and the words are separated either by silence or short pause, as determined by the forced alignment.

We join the newly selected data with the initial set, build a new system, and observe the error rate. We iteratively apply our word-selective training algorithm until the candidate data is exhausted and observe where the error rate is minimized. Our method reduces the error rate from 10.3% to 9.3% by selecting 30% (15 hours) of the 50-hour candidate set without knowledge of the true transcription. This is also an improvement over both supervised (9.6% error at 35% selection) and unsupervised (10.3% error at 25% selection) sentence-selective methods.

Results of supervised word selection, which focus first on words that are in error and second on low confidence, are the same as those achieved with the unsupervised selection algorithm. Since the error rate of the system used in selection is relatively low, the words in error are exhausted in the early selection steps, and selection becomes dominated by the confidence measure.

4. THE WALL STREET JOURNAL CORPUS

We focus on the 5K word Wall Street Journal (WSJ) task, originally studied in [10]. Our training data is comprised of 36,515 utterances of the Sennheiser microphone data from WSJ0 and WSJ1, totaling about 80 hours.

Our test data comes from the 5K word closed vocabulary task. In particular, our development test is the `si_dt_05.odd` set defined in [10], which is a subset of the WSJ1 5K development test data formed by deleting sentences with out-of-vocabulary words and choosing every other sentence, leaving 248 sentences from 10 speakers.

4.1. System Description and Baseline Error Rates

Our recognition system is a tied-state, cross-word, gender-independent triphone system, similar to that in [10], following the training procedures in [6]. The features are 12 mel-frequency cepstral coefficients plus energy, the deltas, and the double deltas for a feature vector of length 39. The features are normalized by per utterance cepstral mean subtraction. In all systems discussed in this section, 8 gaussians are estimated per feature per state.

During recognition, we use a 5K bigram language model. Our baseline system, trained using all 80 hours of the training data, achieves an error rate of 8.7% on the `si_dt_05.odd` development test set. This is comparable to the results reported in [10].

Several additional systems were built to demonstrate baseline performance for varying amounts of training data. Initially, the training set was partitioned into twenty 4-hour subsets, using a procedure that sought to select 5% of the males speakers and 5% of the female speakers, such that the amount of training data selected was 5% of the total available training data. By joining two or more of the twenty subsets and following rules that intend to ensure diversity, we created larger sets. Finally, when training systems of varying sizes, we followed a training procedure identical to that used with all of the training data, modifying only the input training list. The baseline error rates for various training sizes are plotted in Figure 2. The minimum and maximum error rates are plotted with a line, to highlight the wide range of error rates achieved with random samples of the training data.

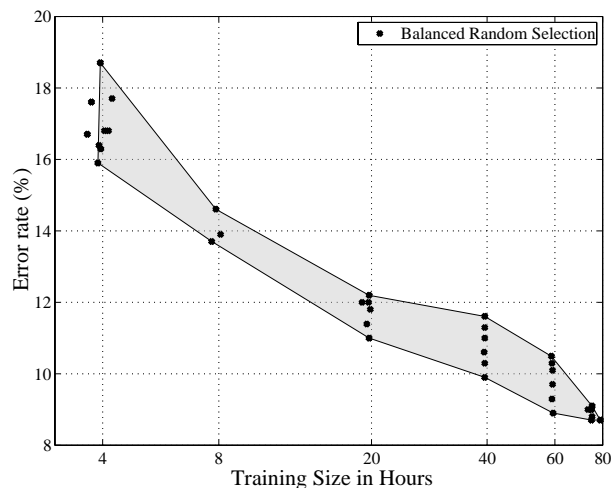


Figure 2: Relationship between error rate and training size for various balanced subsets of the full training available for the WSJ corpus. The range of minimum and maximum error rate is highlighted in gray.

Table 2. Word permutations for generation of the N-best list used to compute the entropy-normalized confidence score. $(n)_{15K}$ contains words with n-syllables drawn from the 15,000 word list represented in the training set. $(m,n)_{5K}$ means all permutations of n-syllable and m-syllable words drawn from the 5,000 word dictionary targeted for the recognition system. Likewise for $(m,n,p)_{5K}$.

| Number of Syllables known to be in word segment | Word Permutation sets |
|---|---------------------------------------|
| 1 | $(1)_{15K}$ |
| 2 | $(2)_{15K}, (1,1)_{5K}$ |
| 3 | $(3)_{15K}, (1,2)_{5K}$ |
| 4 | $(4)_{15K}, (2,2)_{5K}$ |
| 5 | $(5)_{15K}, (2,3)_{5K}$ |
| 6 | $(6)_{15K}, (2,2,2)_{5K}, (3,3)_{5K}$ |
| 7 | $(7)_{15K}, (3,4)_{5K}, (2,2,3)_{5K}$ |

4.2. Word-Selective Criterion and Results

Following the generalized iterative training algorithm, given in Figure 1, we start with a system trained with 5% of the candidate set U . We then apply the model to the remaining candidate data to generate word confidence in the following way. We cheat and use the true transcription to get a word segmentation, as well as knowledge about how many syllables are present in each word segment. Then, with only the knowledge of the start and end points of the word as well as the number of syllables, we compute a word confidence.

The word confidence is computed by first generating an N-best list by competing a select set of words and word combinations. In particular, all words that have the same number of syllables are included. Combinations of words with lessor syllables are also included, as long as the total number of syllables in the combination is equal to the number of syllables known to be in the segment. For higher numbers of syllables, not all possible combinations are used in order to reduce the search space. The particular combinations used in this study are listed in Table 2.

Then, from the N-best list, we obtain the full state alignment and the associated phone labels for the top five unique competitors. (In some cases, combinations of words can achieve the same underlying phone labels as other hypotheses, so only one is carried forward.) From this state alignment, we compute the posterior probability $P(\text{Best}_1^f | O_f; \lambda)$ of each frame f of the 1-best phone hypothesis, denoted Best_1^f , compared to the set of 5-best phone hypotheses, denoted Best_5^f :

$$P(\text{Best}_1^f | O_f; \lambda) = \frac{P(O_f | \text{Best}_1^f; \lambda)}{\sum_{p \in \text{Best}_5^f} P(O_f | p; \lambda)} \quad (6)$$

Table 3: Intermediate values used to compute the entropy-normalized confidence for a particular 5-best list. (The true word is "the"). For each frame, $f = 1 \dots 11$, we observe the label from the full state alignment (denoted $\text{label}_{\text{state}}$, for example "dh₃") for each hypothesis in the 5-best list (columns labeled 1 through 5). The state numbers, 1 through 3, indicate the state chosen from the 3-state HMM used to model the phoneme. $H(O_f)$ is the entropy computed for each frame and the last column is the posterior probability as given in (6).

| f | Phoneme label from the N-best state hypothesis | | | | | $H(O_f)$ | $P(\text{Best}_1^f O_f; \lambda)$ |
|-----|--|-----------------|-----------------|-----------------|-----------------|----------|-------------------------------------|
| | 1 | 2 | 3 | 4 | 5 | | |
| 1 | dh ₁ | dh ₁ | dh ₁ | dh ₁ | t ₁ | 0.7219 | 0.9727 |
| 2 | dh ₁ | dh ₁ | dh ₁ | dh ₁ | t ₂ | 0.7219 | 0.8944 |
| 3 | dh ₁ | dh ₁ | dh ₁ | dh ₁ | t ₃ | 0.7219 | 1.0000 |
| 4 | dh ₂ | dh ₂ | dh ₂ | dh ₂ | ax ₁ | 0.7219 | 0.9844 |
| 5 | dh ₂ | dh ₂ | dh ₂ | dh ₂ | ax ₁ | 0.7219 | 0.9844 |
| 6 | dh ₃ | dh ₃ | dh ₃ | dh ₃ | ax ₁ | 0.7219 | 1.0000 |
| 7 | ax ₁ | dh ₃ | dh ₃ | dh ₃ | ax ₁ | 0.9719 | 0 |
| 8 | ax ₂ | ah ₁ | iy ₁ | ey ₁ | ax ₂ | 1.9219 | 0.9136 |
| 9 | ax ₂ | ah ₂ | iy ₂ | ey ₁ | ax ₂ | 1.9219 | 0.9189 |
| 10 | ax ₃ | ah ₃ | iy ₃ | ey ₂ | ax ₃ | 1.9219 | 0.9053 |
| 11 | ax ₃ | ah ₃ | iy ₃ | ey ₃ | ax ₃ | 1.9219 | 0.7973 |

We then compute an *entropy-normalized confidence*, by weighting these posterior probabilities by the corresponding entropy of the phoneme label observed in each frame across the 5-best hypotheses, denoted $H(O_f)$. Finally, the word confidence is computed by summing, across frames, the entropy-weighted posterior probabilities and dividing by the sum of the per frame entropy:

$$C = \frac{\sum_f P(\text{Best}_1^f | O_f; \lambda) H(O_f)}{\sum_f H(O_f)} \quad (7)$$

For example, Table 3 shows the state alignment for the 5-best list for one one-syllable word segment. For frame 1, O_1 , state 1 of the phone dh , denoted dh_1 , is hypothesized in the first four hypotheses of the 5-best list, and state 1 of the phone t , denoted t_1 , is hypothesized in the last case. The set Best_1^f is $\{dh_1\}$ and the set Best_5^f is $\{dh_1, t_1\}$ for frame $f = 1$. The posterior probability of choosing state 1 of the phone dh in frame 1, denoted $P(dh_1 | O_1; \lambda)$, is therefore computed as,

$$P(dh_1 | O_1; \lambda) = \frac{P(O_1 | dh_1; \lambda)}{P(O_1 | dh_1; \lambda) + P(O_1 | t_1; \lambda)} \quad (8)$$

To compute the entropy for frame 1, we note that the phone label dh occurred with probability 0.8, and the

phone label t occurred with probability 0.2, and apply the standard formula for entropy:

$$H(O_1) = \sum_p p \log_2 p = 0.8 \log_2 0.8 + 0.2 \log_2 0.2 = 0.72 \quad (9)$$

Finally, by summing across frames and dividing by the sum of the per frame entropy, we arrive at a confidence score of 0.84.

We select additional training, equal in size to roughly 3% of the total training, comprised of the set of hypothesized words with the lowest entropy-normalized confidence. Under normal unsupervised circumstances, we would have a human correct the word hypothesis and designate the word boundaries. Instead, since we started with a word segmentation derived from the true transcription, we have a direct mapping to the true label and word boundaries. We extract the observations associated with each of the selected words, thereby generating new training utterances along with the appropriate transcription. Contrary to what we did on the simple acoustic corpus, we do not join adjacent selected words to create longer utterances.

We join the newly selected data with the initial set, build a new system, and observe the error rate. We iteratively apply our word selection algorithm until the candidate set is exhausted. Preliminary results (Figure 3) show that we can select up to 60% (48 hours) of the data, with minimal knowledge of the true transcription, and meet or beat the error rate predicted by random selection of a similar training size.

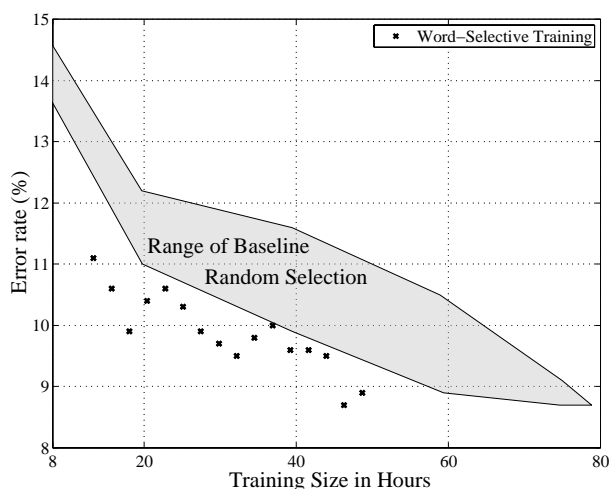


Figure 3: Comparison between the error rate achieved by the word-selective training algorithm and the range of baseline error rates determined by random selection of balanced subsets of the full training available for the WSJ corpus. The word selective algorithm is able to meet or beat the error rate predicted by random selection of a similar training size.

5. SUMMARY AND DISCUSSION

We have presented a method for word-selective training of speech recognition systems meant for realistic speech problems. We started by modifying our sentence-selective method to focus on word selection, and demonstrated that we can reduce the error rate from 10.3% to 9.3% on a simple acoustic corpus, by selecting 30% (15 hours) of the training data available, without knowledge of the true transcription. We then demonstrated successful application of our word-selective training method on a more difficult corpus. We introduced an entropy-normalized word confidence measure and showed that we can select up to 60% (48 hours) of the training data, with minimal knowledge of the true transcription, and match or beat the error rate of a system built using the same amount of randomly selected data.

Our ability to select additional data to continually lower the system error rate is closely tied to the richness of our candidate training set. When we exhaust our candidate set of low confidence examples, then our method requires a fresh source of data. These preliminary results show great potential for our word-selective algorithm in conjunction with the acquisition and transcription of new training data.

6. REFERENCES

- [1] T. M. Kamm and G. G. L. Meyer, "Automatic selection of transcribed training material," In *Proc. IEEE Workshop Automatic Speech Recognition and Understanding*, 2001
- [2] _____, "Selective sampling of training data for speech recognition," In *Proc. Human Language Technology*, 2002
- [3] G. Zavaliagos, et al, "Using untranscribed training data to improve performance," In *Proc. ICSLP*, 1998
- [4] L. Lamel, J.-L. Gauvain, and G. Adda, "Lightly supervised and unsupervised acoustic model training," *Computer Speech and Language*, 16:115-129, 2002.
- [5] T. Kemp and A. Waibel, "Unsupervised training of a speech recognizer: recent experiments," In *Proc. EUROSPEECH*, 1999, pp. 2725-2728.
- [6] S. Young, et al, *The HTK Book, Version 3.2*: Cambridge University Engineering Department, 2002.
- [7] M. Noel, (1997), "Alphadigits," [Online] Available: <http://www.cse.ogi.edu/CSLU/corpora/alphadigit>
- [8] J. Hamaker, A. Ganapathiraju, and J. Picone, (1997), "A proposal for a standard partitioning of the OGI AlphaDigit corpus," [Online] Available: http://www.isip.msstate.edu/projects/speech/software/asr/research/syllable/alphadigits/data/ogi_alphadigits/eval_trans.text
- [9] J. Hamaker, et al, "Advances in alpha digit recognition using syllables," In *Proc. ICASSP*, 1998, pp. 421-424.
- [10] P. C. Woodland, et al, "Large vocabulary continuous speech recognition using HTK," In *Proc. ICASSP*, 1994, pp. 125-8.