AUTOMATIC SELECTION OF TRANSCRIBED TRAINING MATERIAL

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, MD 21218, USA
tkamm@clsp.jhu.edu, gglmeyer@jhu.edu

ABSTRACT

Conventional wisdom says that incorporating more training data is the surest way to reduce the error rate of a speech recognition system. This, in turn, guarantees that speech recognition systems are expensive to train, because of the high cost of annotating training data. In this paper, we propose an iterative training algorithm that seeks to improve the error rate of a speech recognizer without incurring additional transcription cost, by selecting a subset of the already available transcribed training data. We apply the proposed algorithm to an alphadigit recognition problem and reduce the error rate from 10.3% to 9.4% on a particular test set.

1 INTRODUCTION

Conventional wisdom says that incorporating more training data is the surest way to reduce the error rate of a speech recognition system. This, in turn, guarantees that speech recognition systems are expensive to train, because of the high cost of annotating training data.

Several authors [1-3] have investigated methods to incorporate automatically transcribed speech into the training set to reduce error rates without incurring additional transcription cost. These methods start with an initial model and run recognition on a large amount of available speech data. A selection criteria is applied that selects a subset of the automatically recognized material which is then combined with the initial model to create a larger training set. A new model is trained and the error rate of the new model is observed. This procedure may be applied iteratively on unused data to get subsequent models.

The approaches differ in the way that the selection is carried out. In Zavaliagkos [1], a confidence measure is used to select automatically transcribed data that is most likely to be correctly transcribed, setting a threshold such that the expected transcription error of the selected data is around 20%. In Kemp [3], a confidence measure is also used but the threshold is determined experimentally to be such that the most reduction in error of the resulting model can be expected. Lamel [2] does not use a confidence measure, instead relying on available close-captioned transcriptions to arbitrate where the automatic transcription is correct. Closed-captioned transcriptions are a close, but not exact, transcription of what is spoken that is coarsely time-aligned with the audio signal. Sections of the automatic transcription that align successfully with the closed-captioned transcription are selected to feed back into the training set.

In this paper, we propose an iterative training algorithm to improve speech recognition by automatically selecting a subset of the available humanly transcribed training data, thereby improving error rates without incurring additional transcription cost. We investigate the iterative training algorithm within the confines of a simple alphadigit recognition problem. Starting with an initial system trained on a small portion of the available annotated training data, we desire to reduce the error rate of our recognition system by iteratively incorporating automatically selected humanly transcribed data into the training set.

Specifically, using the OGI Alphadigit corpus [4] and the ISIP-defined training/test partition [5], we define a small subset of training to use to train our initial system. Then, using this initial system, we generate the most likely word transcription and determine the recognition error rate for each sentence in the unused portion of the full training set. Using the sentence error rate as a guide, we select sentences to feedback into the training set, generating a new model using the human transcription. By iteratively applying this training algorithm, we are able to reduce the error rate on the test set from 10.3% to 9.4%, and we do this by selecting 35% of the full training set.

The organization of this paper is as follows. Section 2 introduces the notation used through out this paper and discusses the automatic training paradigm. Section 3 discusses the corpus, the baseline systems and the balanced selection procedure. Section 4 discusses the automatic selection criteria and results. Section 5 discusses the iterative training algorithm and results. Finally, a summary and discussion is presented in Section 6.

2 AUTOMATIC TRAINING PARADIGM

In order to discuss the Automatic Training Paradigm, which is the core of our iterative algorithm, we first must introduce some notation. Given T, a set of audio speech cuts, or segments, we define:

- |T|: The size of set T.
- H(T): The human transcription of the set T.
- M(H(T)): The model M built using the human transcription of set T. (It is assumed that the transcription is paired with the audio speech cuts and the audio is also used in modeling.)
- A(M,T): Automatic transcription of the set T using the model M.
- S(A(M,T)): The subset of T selected by the selection criterion, based on the automatic transcription of T using model M.
- E(A(M,T)): The error rate of the automatic transcription of set T using model M.

The Automatic Training Paradigm, which loosely follows the procedures in [1-3], is comprised of two steps. The first step, Automatic Subset Selection [Figure 1], goes as follows: Given a training set T and a human transcript H(T), train a model M_0=M(H(T)). Given another set of speech U_0 which has human transcription H(U_0)), use M_0 to get the automatic transcription A(M_0,U_0). Apply a selection criterion to A(M_0,U_0) to get a subset U_1=S(A(M_0,U_0)) of U_0.

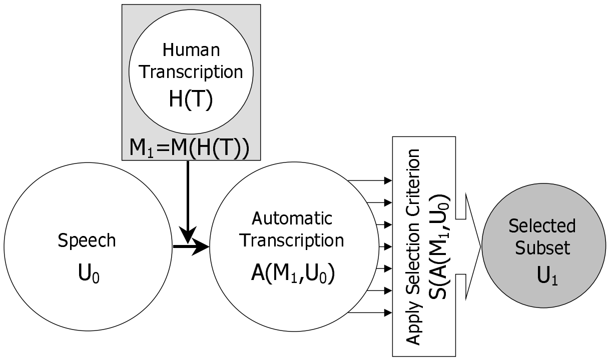

Figure 1: Automatic Training Paradigm: Automatic Subset Selection

Once the subset is selected the second step, Model Update and Test [Figure 2], goes as follows: Use human transcription H(T) and human transcription H(U_1) to train model M_1=M(H(T) U H(U_1)). Apply the model M_1 to the test set D and observe the error rate E(A(M_1,D)).

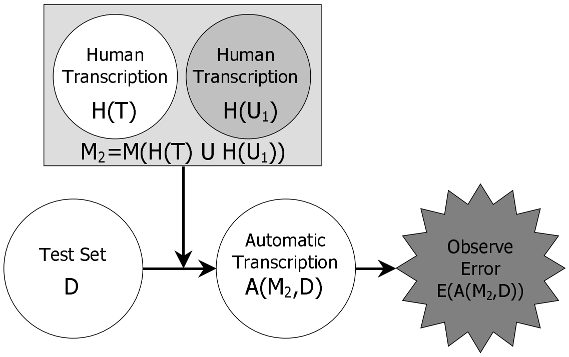

Figure 2: Automatic Training Paradigm: Model Update and Test

3 CORPUS AND BASELINE SYSTEMS

3.1 Data

The speech corpus chosen to perform this study was selected based on four criteria:

1. To focus attention on the acoustic problem.
2. To be a realistic task.
3. To have a standard train/test set defined for comparison with other published results.
4. To be small enough such that experiment turn-around time would be manageable.

The OGI Alphadigit Corpus [4] met these criteria. The corpus is a collection of over 3000 subjects speaking strings of 6 alphadigits over the telephone. The alphadigits 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 alphadigits. Researchers at ISIP [5] have defined a standard train/test partition of this corpus, and have reported error rates achieved on this corpus [6].

From the ISIP training partition of 51545 sentences, 46730 sentences were selected such that the transcription matches one of the prompts given in the prompt list [4]. This was an inexpensive way to remove possible transcription substitution errors from the training set. In this paper, this training set is denoted as T_{ALL}. From the ISIP evaluation test partition of 3329 sentences, 3112 sentences were selected using the prompt list in the same way. This test set is denoted as D.

3.2 System Description

The training and testing of the systems described in this paper were done using HTK [7]. In general, the procedures used followed the steps given in the HTK documentation.

The first baseline system follows [6], with the exception that we choose to build word models rather than syllable models. A word model is created for each alphadigit, plus silence and short pause. The silence/short pause models are built according to the procedure in the HTK documentation [7]. 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 paper have a total of 825 states.

The features 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 paper, 12 gaussians are estimated per feature per state.

An equal probability word network is used to drive recognition. This network is defined as: optional silence, followed by one or more alphadigits, followed by optional silence.

3.3 Baseline Systems

3.3.1 Full training

The first baseline system uses the training set T_{ALL}. A model M_{ALL}=M(H(T_{ALL})) is built, comprised of 36 "word" models and 2 silence models, using the human transcription of T_{ALL} and standard HTK training procedures [7]. This system gives an error rate E(A(M_{ALL},D)) of 10.3% on test set D using model M_{ALL}. This performance is comparable to reported performance on the ISIP full evaluation test of 11.1% [6].

3.3.2 Effect of Reducing the Training Set Size

Since our goal is to improve the error rate by being selective about what training material is used, it is interesting to observe how error rate is affected when the training set is reduced in an unbiased way. To do this, we implemented a Balanced Selection Procedure.

The Balanced Selection Procedure takes as input a training set and information about that set and applies a Balanced Selection Criterion to select a subset of a given size from the input training set. Information about the training set includes: the sex of each speaker, the speaker identity of each sentence, the sentence transcription, and the total number of training examples per token. The Balanced Selection Criterion, where N is the desired subset size in percent, is as follows:

1. N% of the training speakers are selected.
2. There are approximately the same number of male and female speakers.
3. The number of training tokens for each word is approximately N% of the available training for each word.
4. Each word is gender balanced.

Using the Balanced Selection procedure, we selected a subset T_1 of the training set T_{ALL} such that $|T_1| = 0.05|T_{ALL}|$, that is T_1 is a balanced 5% of T_{ALL}. We then built a model M_1=M(H(T_1)) using the human transcription of T_1 and observed the error rate E(A(M_1,D)) of the test set D to be 14.3%. So, using 1/20 of the training set T_{ALL} increases the error from 10.3% to 14.3% on test set D.

We were then interested to know how much the error rate could be affected if the training set size were doubled. We again used the Balanced Selection Procedure to select a subset T_2, disjoint from T_1, of the training set T_{ALL}-T_1 such that $|T_2| = 0.05|T_{ALL}|$. We built a model M_2=M(H(T_1) ∪ H(T_2)) using the human transcription of T_1 and T_2 and observed the error rate E(A(M_2,D)) of the test set D to be 11.8%. So, doubling the training set size from $0.05|T_{ALL}|$ to $0.1|T_{ALL}|$ decreases the error rate from 14.3% to 11.8%, showing that as training size increases, error rate decreases.

3.3.3 Baseline Summary

The full progression of error rate vs. training size for the OGI-alphadigit corpus is shown in Table 1. For small training sizes, increasing the amount of training data can dramatically reduce the error rate. But, as the training set becomes large, the impact on error rate of doubling the data is small. This implies that to improve error rate simply by using more data, it would require several times more data than is already available in the OGI-alphadigit corpus.

Training Size (% of T_{ALL})	Error Rate (%)
5%	14.3%
10%	11.8%
25%	11.0%
50%	10.4%
100%	10.3%

Table 1: Relation between Error Rate and Training Size for selected training sizes.

4 AUTOMATIC SELECTION

In [1-3] additional training material is selected based on the output of a recognizer. In Zavaliagkos [1], material is selected by inspecting a confidence score and selecting data that the recognizer has a high confidence of getting correct. In Kemp [3], material is also selected by confidence score, but more attention is paid to moderately high confidence scores rather than the highest confidence scores, thereby trying to select things that the recognizer probably got correct, but which are not already a good match to the models. In Lamel [2], material is selected where the automatic transcription aligns exactly with the closed-captioned transcription, thereby selecting material that was most probably recognized correctly.

Based on this previous work, we propose to use recognition error as a selector of suitable training material, following the Automatic Training Paradigm described in Section 2.

4.1 Selection by Low Recognition Error

First, we investigate whether doubling the training set by choosing data that is correctly recognized by the model can reduce the error rate.

Given a balanced subset set T_1 of T_{ALL} such that $|T_1|=0.05|T_{ALL}|$, the exact subset T_1 used in the baseline system described in Section 3.3.2, build a model M_1=M(H(T_1)). Using the model M_1, get the automatic transcription A(M_1,T_{ALL}-T_1)) of the rest of the training data. Observe the error rate of each sentence in A(M_1,T_{ALL}-T_1)) and choose a subset T_3 of T_{ALL}-T_1 with the lowest recognition error such that $|T_3|=|T_1|$.

Then, build a model M_3=M(H(T_1) ∪ H(T_3)) and observe the error rate E(A(M_3,D)) of the automatic transcription A(M_3,D) of test set D using model M_3. In this case, we observe the error E(A(M_3,D)) to be 14.1%.

The error rate E(A(M_1,D)) of the initial model M_1 is 14.4%, so doubling by selecting correctly recognized material gives a small improvement. But, since doubling the data using a balanced selection T_2 gave a baseline error rate of 11.8%, we conclude that doubling the training data by using data which is already recognized well by the model does little to improve the recognition error rate.

4.2 Selection by High Recognition Error

Next, we investigate whether doubling the training set by choosing data that is incorrectly recognized by the model can reduce the error rate.

Following the procedure in Section 4.1, we instead choose a subset T_4 of T_{ALL}-T_1 with the highest recognition error such that $|T_4|=|T_1|$. Using model M_4=M(H(T_1) ∪ H(T_4)), we observe the error E(A(M_4,D)) to be 11.8%.

Since doubling the data using a low error selection criterion gave an error rate E(A(M_3,D)) of 14.1%, we conclude that doubling the training data by using data which is poorly recognized by the model reduces error rate more than by using easily recognized data.

Training Size (% of T_{ALL})	Selection Method	Error Rate (%)
5%	Balanced	14.4%
10%	Low Error	14.1%
10%	High Error	11.8%
10%	Balanced	11.8%

Table 2: Comparison of Selection Methods

4.3 Conclusions

A comparison of automatic selection methods is presented in Table 2. Selecting data for training that is poorly recognized by the recognizer (labeled "High Error") is clearly superior to selecting data that is easily recognized by the recognizer (labeled "Low Error").

5 ITERATIVE TRAINING ALGORITHM

In the previous section, we investigated two selection criteria and determined that doubling the training set by feeding back poorly recognized training sentences into the recognition model greatly reduces the recognition error rate on a test set. Now we take the obvious next step and propose to iteratively apply the selection criterion to the remaining unused training data, thereby generating a sequence of models.

The Iterative Training Algorithm to generate N models is as follows: Given a set of training data T_0 and the human transcription $H(T_0)$:

1. Select a subset T_1 from T_0 such that it is balanced (i.e. by sex and token) and let i = 1.
2. Obtain the model $M_i = M(H(T_i))$ by using T_i and the human transcription $H(T_i)$.
3. Select a subset $S_i = S(A(M_i,T_0-T_i))$ of T_0-T_i by observing error in the automatic transcription $A(M_i,T_0-T_i)$.
4. Let $T_{i+1} = S_i \cup T_i$ and let i = i + 1.
5. If (i == N) stop, else go to step 2.

We applied the proposed Iterative Training Algorithm to the OGI-alphadigit corpus by 1) letting T_0=T_{ALL}, all of the available training data, 2) letting T_1 be the balanced subset selected from T_{ALL}, as described in Section 3.3.2 and, 3) replacing the stopping rule in step 5 with a heuristic stopping rule which is: stop if the error is no longer improving on a test set.

We ran the Iterative Training Algorithm using a high error selection criterion where $|S_i| = |T_1|$ for *i* = 1 to N-1. The sequence of models were judged by applying the model M_i to the test set D and observing the error rate $E_i = E(A(M_i,D))$. The error rates of the sequence of models generated are summarized in Table 3.

By iteratively applying the High Error Selection criteria, it is possible to select a subset of the full training from which to build a model that gives better recognition performance (9.4% error) than a model built using the full training set (10.3% error). In fact, it is only necessary to use 35% of the full training to achieve this result.

6 SUMMARY AND DISCUSSION

We have presented an iterative algorithm that automatically selects training material by observing recognition error and selecting subsequent data that is hard for the current model to recognize. We have successfully demonstrated the algorithm on the OGI-alphadigit corpus, reducing the error rate from 10.3%, using all of the available training data, to 9.4%, using 35% of the available training data.

The iterative training algorithm presented in this paper is a simplified application of the boosting technique. "Boosting" is a general method for improving the error rate of almost any learning algorithm. A particular boosting algorithm, called AdaBoost [8], iteratively calls a base learning algorithm in order to maintain a distribution of weights on the training examples at each iteration. The weights on the training set are initially equal, and are updated on each iteration by increasing the weights of incorrectly classified examples, thereby forcing the learning algorithm to focus on the hard to classify examples in the training set. A new classifier is trained in each iteration, based on the current weighting of the training examples. Finally, a composite classifier is based on a weighted majority vote of the generated sequence of classifiers.

Our iterative training algorithm starts by selecting a subset of the training set, thereby setting the weights of the subset to 1 and the remainder of the training set to 0. A classifier is built and applied to the portion of the training set that has weight 0. The classification error of each training example is observed, and the weights of a portion of those that are most errorful are updated to 1. A new classifier is built and the algorithm is iteratively applied to the portion of the training set that now has weight 0. The final classifier is chosen from the classifiers built at each iteration, based on performance on a held out test set.

Our algorithm can be extended to enable pragmatic use of future human transcription investment by selecting speech data to transcribe which will contribute most to reduce the error rate. By using a low confidence selection criterion rather than a high error selection criterion on a large set of untranscribed speech, a small set of data could be selected, transcribed by humans and then fed back into training. By observing error on a reasonably large test set, this process could be iterated until no reduction in error rate is observed. If this scenario were followed on our example corpus and the selection criteria performs similarly, only 40% of the speech data would have been transcribed.

Iteration	Training Size (% of T_{ALL})	Error Rate (%)
1	5%	14.3%
2	10%	11.8%
3	15%	11.1%
4	20%	10.4%
5	25%	9.9%
6	30%	9.5%
7	35%	**9.4%**
8	40%	9.7%

Table 3: Comparison of Model Error Rates at each Iteration of the Iterative Training Algorithm

7 REFERENCES

[1] G. Zavaliagkos and T. Colthurst, “Utilizing Untranscribed Training Data to Improve Performance,” presented at Broadcast News Transcription and Understanding Workshop, Lansdowne, Virginia, 1998.

[2] L. Lamel, J.-L. Gauvain, and G. Adda, “Lightly Supervised Acoustic Model Training,” presented at ISCA ITRW Workshop on Automatic Speech Recognition: Challenges for the New Millennium, Paris, France, 2000.

[3] T. Kemp and A. Waibel, “Unsupervised Training of a Speech Recognizer: Recent Experiments,” In *Proc. EUROSPEECH*, 1999, pp. 2725-2728.

[4] M. Noel, (1997), “Alphadigits,” Center for Spoken Lang. Understand., Oregon Graduate Inst. Sci. Technol., Portland, OR, [Online] Available: http://www.cse.ogi.edu/CSLU/corpora/alphadigit

[5] J. Hamaker, A. Ganapathiraju, and J. Picone, (1997), “A proposal for a standard partitioning of the OGI AlphaDigit corpus,” Inst. Signal Inform. Process., Mississippi State Univ., [Online] Available: http://www.isip.msstate.edu/projects/speech/software/asr/research/syllable/alphadigits/data/ogi_alphadigits/eval_trans.text

[6] J. Hamaker, et al., “Advances in Alpha Digit Recognition Using Syllables,” In *Proc. ICASSP*, 1998, pp. 421-424.

[7] S. Young, et al., *The HTK Book, Version 2.2*: Entropic Ltd., 1999.

[8] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” *Journal of Computer and System Sciences*, *55*(*1*), pp. 119-139, 1997.