SRI'S 2004 BROADCAST NEWS SPEECH TO TEXT SYSTEM

A. Venkataraman, R. Gadde, A. Stolcke, D. Vergyri, W. Wang, J. Zheng

Speech Technology and Research Lab
SRI International
Menlo Park, CA 94025

ABSTRACT

We describe the system that we used to automatically transcribe Broadcast News Speech in the 2004 NIST evaluations sponsored by the DARPA EARS program. We also describe some of our efforts leading up to the NIST evaluation, in particular, concerning data selection and model training. Since one of the hallmarks of the 2004 evaluation was the availability of relatively large amounts of quickly transcribed data, we describe our techniques, and those of others in adapting this data for model training, and experiments attempting to evaluate the suitability of this data for training purposes. We conclude by presenting results and also discussing whether or not increases in amounts of training data proportionately benefit the research community.

1. INTRODUCTION

The Speech Technology and Research Laboratory at SRI International was one of the four sites that participated in the NIST Broadcast News (BN) recognition evaluation in 2004. The other sites were LIMSI (France), BBN (USA), and the Cambridge University Engineering Department (CUED). In addition to submitting a system that was used in a collaborative four-way result combination, we submitted an independent set of results from our system. In this paper, we describe our efforts leading up to the evaluations and the detailed architecture of the system that was finally submitted.

The BN system architecture is described in Section 2. It is followed in Section 3 by a description of the technique (Flexalign) that we employed to rapidly repair the quickly transcribed data released by the Linguistic Data Consortium (LDC) in order to adapt it for training acoustic models. In Section 3.1, we present results that compare the suitability of Flexalign for preprocessing training data, relative to other state-of-the-art methods. Finally, Section 4 describes our experiments to gauge the effects and benefits of gradually increasing the amount of data used for model training and discusses the results of these experiments.

2. SYSTEM DESCRIPTION

2.1. Overview

The SRI DECIPHER(TM) speaker-independent continuous speech recognition system is based on continuous-density, genonic hidden Markov models (HMMs) [1]. The system uses a multiple-pass recognition strategy [2], with a vocabulary of 61,808 words. Cepstral mean and variance normalization, vocal tract length normalization (VTLN) and model-based speaker-specific feature normalization were applied in the front end. A perceptual linear prediction (PLP) front end was used to obtain a 52-dimensional input feature vector, which was reduced to 39 dimensions using heteroscedastic linear discriminant analysis (HLDA). The acoustic models were trained using the minimum phone error (MPE) discriminative criterion [3]. In the first-pass of decoding, within-word acoustic models were used with a bigram language model (LM) to generate HTK lattices. The HTK lattices were then rescored using a 5-gram SuperARV LM and a within-word duration model to produce 1-best hypotheses. These hypotheses were used for adapting three sets of acoustic models, namely cross-word MPE-trained PLP models, cross-word Maximum Likelihod Estimation (MLE)-trained PLP models, and within-word MPE-trained PLP models. Means and variances of all acoustic models were adapted to each speaker by way of affine Gaussian transforms [4, 5, 6, 7, 8]. The HTK lattices were expanded using a 5-gram SuperARV LM [9] to 4-gram probabilistic finite state grammar (PFSG) lattices and the three adapted acoustic models were used to decode these lattices to generate n-best lists. Each of these sets of n-best lists was rescored using the 5-gram SuperARV LM and a cross-word duration model [10]. The resultant rescored n-best-lists were combined using N-best-ROVER word posterior maximization [11] to generate the most likely word at each position. Figure 1 illustrates the flow of control pictorially.

2.2. Front-end Processing

The front-end signal processor computes a 256-point fast fourier transform (FFT) sample every 10ms. These are then integrated into 24 spectral bands from 94 to 6438 Hz, from which we compute 12 cepstral coefficients (C1-C12) plus C0. From these 13 cepstral features (C0-C12), first-, second- and third-order differences over time are computed giving 52-dimensional cepstral feature vectors. The dimensionality of these vectors is subsequently reduced to 39 using HLDA. The resultant 39-dimensional feature vectors are normalized for zero mean and unit variance. That is, we subtract the mean of the feature over all the segments of each speaker in the training data and the variance of the features is adjusted to be one along all dimensions.

The vocal-tract length (VTL) estimate for each speaker is calculated using the algorithm reported in [12]. For this, we use a 128-Gaussian mixture model (GMM) trained on a subset of the mean-variance normalized training data. The VTL estimate for the test data (for each speaker) is then computed by maximizing the likelihood of the mean and variance normalized test features with respect to the GMM. To compute the optimum VTL, we search over seven discrete VTL values in the interval [0.88, 1.12]. Once the VTL is estimated, we use it to recompute the features, which are now normalized for VTL, mean, and variance. No separate VTL estimates were computed for the PLP system. Instead, for

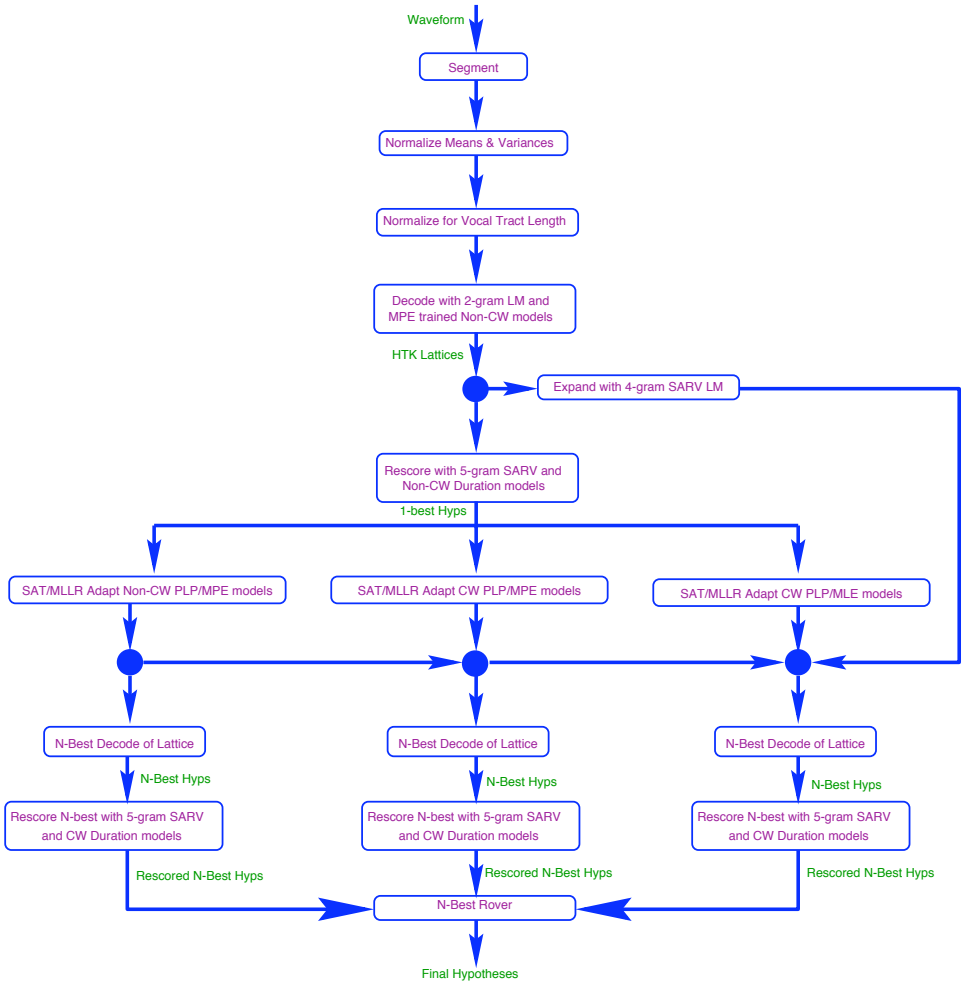

Fig. 1. Pictorial representation of the decoding strategy.

the sake of simplicity and in the interest of time, the values we had obtained earlier using mel frequency cepstrum (MFC) models were used.

We also apply a speaker-specific feature transformation to the augmented cepstral vectors (after mean and variance normalization). The transforms were estimated using constrained maximum likelihood linear regression (MLLR) so as to maximize the likelihood of the test data under the non-cross-word (non-CW) recognition models using the first-pass recognition output as reference. The same normalization is performed in training and testing, making this approach equivalent to speaker-adaptive training (SAT) [13]. However, training speakers used full-matrix transforms, while test speakers used block-diagonal transforms.

2.3. Acoustic Model Adaptation

We use maximum-likelihood transformation-based adaptation [5, 6, 7] to adapt the speaker-independent acoustic models to obtain speaker-specific acoustic models. This is done by aligning the models with the data and using the forward-backward algorithm [8] using the recognition hypotheses of the first-stage adapted acoustic models (transcription-model adaptation). Specifically, we performed a full matrix transformation of the HMM mean vectors as described in [8]. Separate transformations were estimated for different Gaussian clusters to make better use of the data. We used six separate transformations to adapt the cross-word models and the Gaussians corresponding to the silence or pause phone were adapted with one of the transformations. For the second stage of adaptation, we also used variance scaling transformation [4].

2.4. Acoustic Training

The data used for acoustic model training came from the following corpora provided by linguistic data consortium (LDC):

- 1996 Hub-4 English Broadcast News Speech (104 hours)
- 1997 Hub-4 English Broadcast News Speech (97 hours)
- TDT4 (248 hours)
- TDT2 (272 hours)
- TDT4-extra (226 hours)

- BN-CC, broadcast news audio with closed captioned transcripts (2300 hours)

It is noteworthy that the actual number of hours of data available for training is less than the number shown above in parentheses because the above numbers represent the total number of distributed hours, including silence, untranscribed musical interludes and advertisements that were excluded, and in the case of quick transcriptions, data that we deemed to be of quality unsuitable for training.

As with the test data, the training data was also processed with cepstral normalizations, VTL normalizations, and (for SAT models) model-based feature normalization. Decision tree state clustering was used to cluster the triphone states to 2500 clusters for the within-word models and 3000 for cross-word acoustic models. We trained 200 gaussians per state cluster for the within-word model and 128 for the cross-word models.

2.5. Language Model

Our recognition and rescoring language models were based on what are referred to as Super Abstract Role Value (SuperARV or SARV) tags in the literature [9]. SARV tags are a convenient way to represent the joint assignment of multiple dependencies to words in a sentence, and thereby lexically capture constraint dependency grammar (CDG) parse tree. Since the tags do not encode information pertaining to full parses of sentences, the resultant language models are commonly referred to as almost-parsing language models. Our SuperARV language model was trained on the subset of successfully parsed sentences of the LM training data, and was pruned before use.

The LM training data was first partitioned into separate training corpora based on type (acoustic training transcripts, broadcast news transcripts and newswire text), recency (before and after 1997), source (e.g., Hub-4, TDT-4 and North American Business News (NABN)) and quality (some transcriptions appeared to be more easily amenable to normalization than others, for example, SGML-format NABN data was of relatively poorer quality than the VPZ-format[1] data). Separate SuperARV LMs were generated from each of these corpora. Table 1 lists the approximate amounts of training data from each source.

Corpus	Words
hub4-acoustic	1.7M
hub4-lmtext	130M
nabn-ex-97-sgml	311M
nabn-to-97-sgml	124M
nabn-to-97-vpz	306M
tdt4-broadcast	2.5M
tdt4-newswire	9.6M
tdt2-broadcast	6.4M
tdt2-newswire	15M
BN-CC	48M

Table 1. Amounts of LM training data by subcorpora.

A subset of sentences from BN-CC and the TDT-4 development set was held out and excluded from an initial training session in order to constitute an independent tuning set. We determined the best mixture of the component language models by employing the (expectation maximization) EM algorithm to maximize the data likelihood of this held-out set of sentences. Once the mixture weights were obtained thus, the data was reintroduced into the pool and component language models were trained on the whole data. The components were then interpolated using the weights obtained earlier to yield a single class based 5-gram SARV language model. We used this language model to assign probabilities to each of the n-grams in a decoding bigram language model in a procedure we termed *language model rescoring*. Thus we were able to use a bigram SARV language model indirectly in the first pass of our recognition system as well.

[1] Compressed Verbalized Punctuation

2.6. Recognition Vocabulary

The recognizer vocabulary was selected from unigram count files of each of the corpora that we used for training language models. A maximum-likelihood-based word selection criterion [14] was used to combine normalized unigram counts from each of the corpora in a way that maximized the total unigram log likelihood of the TDT-4 development test set. The resultant combined counts were ranked by magnitude and used to generate an OOV-rate versus vocabulary size plot. From this, a vocabulary size of approximately 60,000 words was chosen. We then added 2006 frequent multiletter sequences ("C_N_N") and 1389 word bigrams and trigrams (e.g. "A_LOT_OF") as "multiwords" to the recognizer vocabulary to improve modeling of cross-word reductions and also to extend the average scope of the bigram LM.

The pronuciation dictionary was based on the CMU V0.4 lexicon. Stress information was stripped, but "AH0" was rewritten as schwa (AX), and a flap (DX) replaced D and T in the appropriate phonetic contexts. Two additional phones were dedicated to modeling of filled pauses (PUH as the vowel in "uh" and "um" and PUM for the nasal in "um").

Multiwords had alternate pronunciations including both the baseforms and any idiosyncratic or reduced forms, the latter ones were created by hand by a phonetician. All alternate pronunciations were weighted relative to each other according to their smoothed frequency of occurrence in the training data, and pronunciations with probability less than 0.3 times that of the most frequent form were pruned from the lexicon.

In addition to the lexical words, separate models were created for nonlexical nonspeech phenomena, each with a dedicated phone. These are listed in Table 2. The resulting phone set had 46 phones.

Word	Phone	Description
@reject@	rej	unintelligible, fragmented or OOV
-pau-	-	inter-word pauses

Table 2. Special phones and the words in which they occur. The @reject@ word is unique in that it has a non-linear structure. It contains a loop on the rej phone allowing it to stretch arbitrarily long. This is significant, as will become apparent in subsequent discussion on Flexalign in Section 3.

2.7. Duration Modeling

We scored our N-best hypotheses with word-specific phone duration models. The models represent permissible duration patterns

of words. For example, a word with three phones was represented by a three-dimensional feature vector comprising the durations of the three phones. This representation allowed automatic training of models from the acoustic training data. Durations were normalized for the average speaking rate of the speaker and conditioned on whether or not a pause follows a word. To train the duration models, we generated the phone backtraces of all utterances in the acoustic training data and extracted the duration patterns of the words, which were then used to train GMMs of the duration patterns. To handle words unseen in the training data, we also trained individual triphone and monophone duration models. Whenever an unseen word was encountered the triphone models were first consulted, backing off to context-independent monophone models if they could not be found. A detailed description of this approach can be found in [10].

2.8. Hypothesis Search

The decoding process used a time-synchronous forward-backward search [15] where the forward search was based on a bigram network with lexical tree-structured backoff node [16]. The goal of the rescoring step was to generate high-quality hypotheses for adaptation, while the goal of the lattice expansion step was to generate word lattices for progressive search with more elaborate acoustic and language models.

Bigram word lattices from the initial decoding step were first pruned to eliminate hypotheses with low posterior probabilities (normalized forward-backward scores), and then reduced using an iterative algorithm and finally expanded to incorporate 4-gram LM weights, using the LM's backoff structure to minimize lattice size [17]. Recognition from lattices ignores prior acoustic scores and time alignments and uses the word lattices as a constrained language model.

N-best outputs from the three different recognition systems were combined using a modified version of the NIST ROVER algorithm [18]. Our N-best ROVER algorithm combines ROVER's weighted voting among different systems with voting among different N-best hypotheses for the purpose of explicit word error minimization [19, 20, 11]. For each system's N-best list we computed posterior hypothesis probabilities and multiplied them by a global weight reflecting the system's reliability. N-best hypotheses from all systems were then merged into a single alignment matrix. For each position in the alignment the word (or empty hypothesis) with the highest combined posterior was selected. The system weights as well as the global score scaling factor (which controls the peakedness of the posterior distribution) were optimized on the TDT-4 developement test data.

2.9. Confidence Measures

Our final output consisted of a set of words with time-alignment information, together with a confidence measure. To compute the confidence measures, we used a neural network that took several word-level features as input [21]. These features were, respectively,

- the unigram probability of the hypothesized word
- its relative word position in the hypothesis (normalized by the sentence length)
- the logarithm (to base 10) of the sentence length
- the total numer of alternative candidates in the same position as the word (excluding DELETE) that have a nonzero posterior probability
- the log of the posterior probability of the word as determined from the consensus mesh
- the log of the probability of the word to the left of the current word
- the log of the probability of the word to the right of the current word
- a boolean value that is 1 if the a-posteriori most likely candidate to the left of the current word is a DELETE and 0 otherwise
- a boolean value that is 1 if the a-posteriori most likely candidate to the right of the current word is a DELETE and 0 otherwise.

The neural network was trained on the TDT-4 development testset data where the system output had been labeled as correct or incorrect by a dynamic alignment between the hypotheses and the reference word strings. Two output nodes were used to represent the two classes (correct and incorrect), with softmax output layers to produce a probability like score. The neural network had one hidden layer with four hidden nodes and was trained to minimize cross-entropy on a validation set that was extracted from the training data.

3. FLEXIBLE ALIGNMENT

We used a procedure that we call Flexalign [22] to process and repair quickly transcribed (QT) data in an efficient way in order to make it suitable for training our acoustic models. The procedure is characterized by a rapid alignment of the acoustic signal to specially designed word lattices that allow for the possibility of either skipping erroneously transcribed or untranscribed words in either the transcript or the acoustic signal, and/or the insertion of an optional disfluency before the onset of every word. We do this programatically, by processing every transcript to generate a hypothesis search graph that has the following properties.

1: Every word is made optional. This allows for arbitrary amounts of the transcript to be skipped while still entertaining the possibility of resynchronizing with the waveform at a later point. A fragment of the sublattice corresponding to the transcript for "this is **headline news** second watch with judy fortin" is depicted in Figure 2.

2: Every word is preceded by either an optional *garbage* word, which we call the @reject@ word, or one of a certain number of disfluencies, namely, um, uh, uhhuh, huh, hmm, or uhuh. This allows for arbitrary amounts of the acoustic signal to be skipped while still entertaining the possibility of resynchronizing with the transcription at a later point. It also allows some of the words frequently omitted in QT to be recovered. This sublattice is depicted in Figure 3

3: Every word is followed by an optional pause of variable length. The sublattice for the optional pause is depicted in Figure 4. In our approach the pause word is modeled using a special pause phone that is trained on background noise.

The @reject@ word nominally represents out-of-vocabulary (OOV) items in the recognition language model. Consequently, it

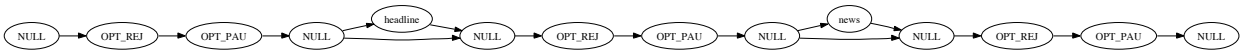

Fig. 2. A fragment of the flexalign lattice for the transcript "This is headline news with Judy Fortin.

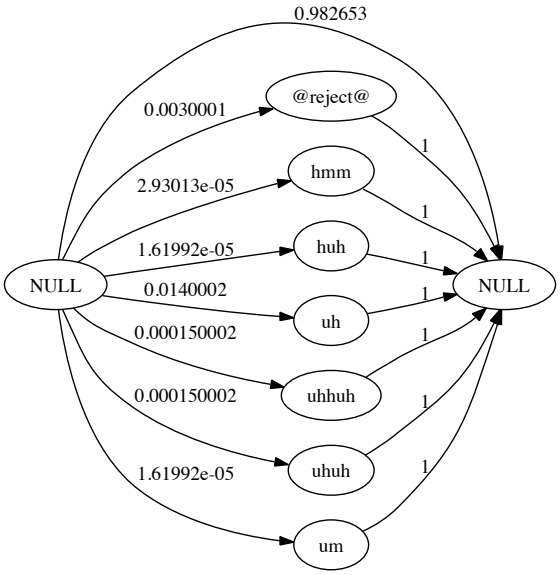

Fig. 3. The structure of the sub-lattice OPT_NSREJ for the optional nonspeech words.

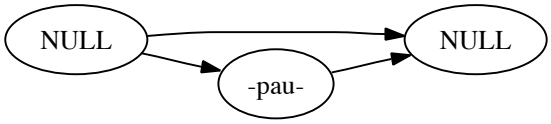

Fig. 4. The structure of the sublattice OPT_PAU for an optional pause before every word.

allows for the possibility of unknown words being present in the acoustic signal. These words are matched to the @reject@ word if none of the known words provides a better acoustic match. Because the @reject@ word ought to be able to match arbitrary unknown words, it is composed using the special phone that we had called *rej*. Also, because @reject@ nominally stands for OOV items, we encode it in the search graph with a probability that is roughly equal to that of an OOV item (the OOV rate) as measured on a development test set. Similarly, the probabilities of transitioning through each of the disfluencies are likewise determined empirically to be the relative frequencies of each disfluency on a development test set.

The acoustic models used to perform the flexible alignment were trained on the 1996 and 1997 Hub4 Broadcast News acoustic training data. They consist of gender-independent, within-word genonic triphones using standard 39-dimensional mel frequency cepstral coefficient (MFCC) features. The recognition vocabulary was selected to be the top 125000 words from the Hub4 language model training data using the maximum likelihood procedure due to [14] tuned on the TDT-4 data set on which the resultant vocabulary was found to have an OOV rate of less than 0.1

The final output of our flexible alignment procedure was a set of reference transcripts that we expected to be of better quality than the original closed captioned transcripts that were distributed by LDC. Approximately 36 hours of these transcripts were discarded as unsuitable for training; 14 of these hours were discarded by the alignment procedure itself as unalignable data. The remaining 22 hours were discarded post-hoc because the proportion (20%) of @reject@ words in them exceeded our intuitive threshold for being acceptable.

3.1. Evaluation of FlexAlign

To evaluate if the flexaligned transcripts were good enough for the purposes of training acoustic models, we compared their quality with that of unrepaired quick transcriptions from LDC. Further, to be sure, we also compared them to careful transcripts of the same set, which were also released by the LDC, and to ASR output using a biased language model from Cambridge University, which is one of the state-of-the-art techniques to deal with quick transcriptions. The Cambridge transcripts were generated by a fast, stripped-down version of the regular Cambridge ASR system that had the best performance in the NIST 2003 Broadcast News STT evaluations [23]. These transcripts were also used by Cambridge University for experiments on lightly supervised acoustic model training [24].

Comparison and evaluation of the transcripts was done indirectly, by training acoustic models with each and measuring the performance of each set of models on a standard ASR task. Three data sets were used to evaluate the acoustic models. These were the 2003 and 2004 TDT-4 development test sets defined by the DARPA EARS program participants, which we denote as **dev2003** and **dev2004** and the RT-03 Speech To Text (STT) evaluation data set for broadcast news distributed by NIST which we denote as **eval2003**.

Besides the TDT-4 reference and acoustic data, the data used for acoustic model training included 1996 and 1997 Hub4 English Broadcast News Speech (75 hours and 71 hours, respectively). The acoustic training data was processed with cepstral normalizations and VTLN. 52-dimensional MFC features (13 MFCs + first-, second- and third-order differences) were reduced to 39 dimensions using HLDA. The acoustic models were trained using the maximum likelihood criterion [23] as follows: After initially training phonetically tied mixture (PTM) models, the models were clustered and genonic acoustic models were trained. Phonetic models had the usual three-state HMM structure with left-to-right transitions and self-loops (enforcing a minimum duration of three frames).

All models were trained on the same 146 hours of Hub4 Broadcast News training data from 1996 and 1997, but with the different supplemental sources of TDT-4 data, and in the baseline, no supplemental TDT-4 data. The versions of the TDT-4 transcripts that we evaluated include the original and hand-corrected closed

captions from the LDC, Cambridge University's ASR transcripts, and our own flexibly aligned transcripts. Transcripts that did not align during training were simply discarded. We also discarded any training shows from the two-week period from which the two development test sets were drawn and shows that did not belong to the subset of hand-corrected TDT-4 transcripts provided by the LDC.

To reduce the influence of varying segmentation strategies between the systems employed by Cambridge University and us, we trained two variations of acoustic models from the Cambridge transcripts. One used their own segmentations and the other had segment lengths determined by our waveform segments. Consequently, there were five acoustic models evaluated on the three test sets as shown in Table 3.

Model	**Dev-03**	**Eval-03**	**Dev-04**
Baseline	17.8	14.9	19.2
LDC-raw	16.8	14.7	18.9
LDC-hand-corrected	15.9	13.9	18.1
CUED-CUED-segs	16.0	14.1	18.2
CUED-SRI-segs	15.9	14.0	18.2
Flexalign	15.8	14.4	18.0

Table 3. WER results (%) using models trained with flexalign transcripts compared against those with (1) only Hub4 transcripts (no TDT-4 data) (2) Raw closed captioned transcripts processed just as with our flexible alignment procedure except that optional words were not used and unalignable transcripts were discarded (3) A hand-corrected subset of (2) provided by LDC (4) Transcripts from Cambridge (CUED) with lengths determined by CUED's segmentation and (5) Transcripts from Cambridge but with lengths determined by our waveform segments. The WERs refer to the word error rate after 5-gram language model rescoring.

As Table 3 shows, the Flexalign model produced the lowest word error rate (WER) on both the Hub4 Broadcast News 2003 and 2004 TDT-4 development test set. On the Eval 2003 test set, the performance of the Flexalign model is still competitive with the performance of the two best models. Especially significant is the fact that the LDC-hand-corrected transcripts are only about as good as the automatically repaired transcripts. Taking into account that the flexible alignment approach is faster than real time at about $0.53\times$RT these results verify that this approach is at once effective and efficient.

3.2. Precision and Recall on Disfluency Insertion

A final check examined the accuracy of inserting disfluencies using the flexible alignment approach. The experiment was performed on the 1996 Hub4 Broadcast News training transcripts by initially removing all of the disfluencies from the set and re-inserting them using flexalign. The original transcripts with disfluencies served as the reference set. We then evaluated the precision (proportion of inserted disfluencies that were correct) and the recall (proportion of correct disfluencies that were inserted). On the 1996 broadcast news acoustic training data, the flexible alignment approach obtains a precision of 68% and a recall of 54%. While it is hard to assess these numbers in absolute terms for lack of a point of reference, we take them as further indication that the flexible alignment approach works reasonably on its intended task of fixing inaccuracies in quick transcriptions.

4. EFFECT OF VARYING TRAINING DATA SIZE

Besides comparing the quality of transcripts generated using different methods for acoustic model training, we were interested in investigating the effect of varying the amount of training data for our proposed approach. This experiment was particularly significant under the circumstances when large amounts of data were being made incrementally available to the speech recognition community. This resulted in a corresponding proliferation in the number of different acoustic models that were trained and the consequent housekeeping involved in maintaining them. We thus decided to determine if the additional data was really beneficial or not and if so to what degree.

4.1. Strategy

Our strategy to determine the benefits of adding to training data was again indirect. We train a number of acoustic models, each with increasing amounts of training data, and measured the performance of these models on various test sets with the following reasoning. If the additional data was helpful, then we would see a corresponding decrease in the word error rate of models trained using it. If not, then the quality of models would tend to asymptote at some point and it would become increasingly difficult to obtain performance improvements simply by boosting the amount of training data.

Table 4 lists the various data sets chosen by us to perform this experiment. Our baseline models were trained with a mere 146 hours of carefully transcribed Hub-4 data from 1996 and 1997. We progressively increase the quantity of data using TDT-4, TDT-2, TDT-4-extra, and BN-CC until we have almost 1700 hours of training data, an order of magnitude more than we started with.

Source	Hours	Cumulative hours
Hub-4 (1996/97)	146	146
TDT-4 CC	200	346
TDT-2 Token text	240	586
TDT-4-extra CC	191	777
BN-CC	854	1631

Table 4.

The acoustic models for these experiments consisted of gender-independent, crossword triphones using 52-dimensional PLP features reduced to 39 using HLDA. We used top-down decision-tree-based clustering of triphone states [25] generating 3000 clusters. The number of gaussians per cluster determines the total number of gaussians in the system. This was calculated using a rule of thumb from personal communication with BBN and set at the values shown in Table 5.

4.2. Decoding Strategy

In view of the large model sizes involved and the number of experiments that need to be run, the decoding strategy used for this set of experiments was designed to be fast and efficient. After the usual segmentation, clustering, mean-variance normalization and vocal tract length normalization, we used a word bigram LM to generate HTK lattices. These were rescored using cross-word duration models and expanded with a 5-gram word LM. The 1-best hypotheses were extracted from these lattices and used to adapt

Source	Tot. hours	Gauss/leaf	Tot. Gauss
Hub-4 (1996/97)	146	64	192K
+TDT-4 CC	346	64	192K
+TDT-2 Token text	586	128	384K
+TDT-4-extra CC	777	160	480K
+BN-CC	1631	200	600K

Table 5. Total number of gaussians in the system for each set of models. The rule of thumb specified approximately 15 times the square root of the number of hours. Note that due to time constraints, we had used an existing and older set of acoustic models for line 2 (TDT4-CC) and thus the total number of gaussians for this set of models does not appear to follow the rule.

the same acoustic models as used initially. HTK bigram lattices were again generated using these adapted models, again rescored with crossword duration models and expanded using the 5-gram LM. The 5-gram expanded lattices were processed directly by the consensus decoding algorithm to generate the final hypotheses for scoring.

4.3. Results and Discussion

Table 6 shows the WER results obtained on various test sets using each of the set of models described earlier. A graphical plot of the WER against the amounts of training data is presented in Figure 5

Source	Eval-04	Dev-04f	Eval-03
Hub-4 (1996/97)	19.2	22.3	13.0
+TDT-4 CC	19.2	21.9	12.5
+TDT-2 Token text	18.2	20.9	11.9
+TDT-4-extra CC	18.2	20.9	11.6
+BN-CC	17.1	19.9	11.4

Table 6. Word Error Rates on three different test sets using models trained with increasing amounts of data.

As the table and the plot show, we see gains to be obtained by increasing the amount of training data. However, we also notice a trend for the gain to be dependent upon the specific type and character of data that is used to augment the training set. On Eval-04, we see no gain upon the addition of either TDT-4 or TDT-4-extra, presumably because the test set is different in character from either of these. Our tentative conclusions from these results are that while more data is always good, it is significantly better for the data to be matched to the evaluation conditions than not since the results seem to depend heavily on the source of the training data. Because increasing amount of training data also increases the sizes of the acoustic models, and consequently lengthens the decoding time, we believe that we are at a point at which we ought to examine issues of data selection and model training in greater detail before processing any more data. This situation is especially true considering that the vast quantities of data that are available to us at present are likely to only be quickly transcribed. Hence it is important that methods for selecting and repairing it are refined and tuned to be optimal before we refocus our attention on procuring additional training data.

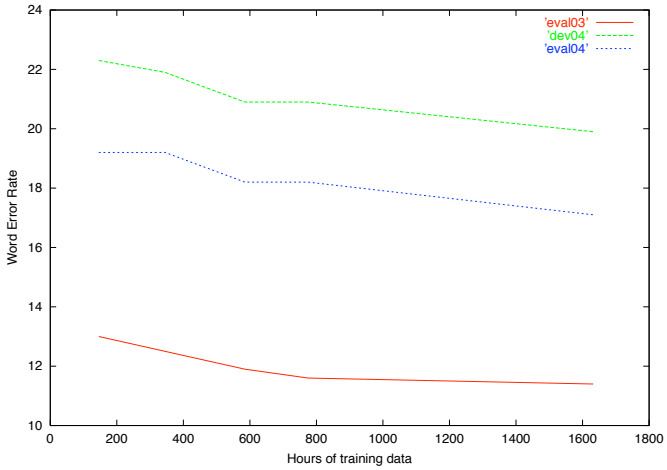

Fig. 5. Plot of WER versus the number of hours of acoustic training data

5. SUMMARY

We have described in detail the system that we used to generate results for the NIST 2004 BN-STT task. We have also described our Flexalign technique to repair the vast amounts of quick transcriptions distributed by LDC. We described experiments to test the efficacy of this method in comparison to both hand-transcribing the data and using other state-of-the-art methods, showing that the procedure is at once both competitive and efficient. We then discussed whether increasing the amounts of training data indiscriminatively is bound to be useful or not in light of the fact that much of it will be quickly transcribed. We finally drew a tentative conclusion that a number of research issues relating to data selection and repair might be better addressed first, before refocusing our attention on augmenting our training data sets.

6. ACKNOWLEDGMENTS

This research was supported by DARPA under contract MDA972-02-C-0038. Distribution is unlimited. We are grateful to Phil Woodland and Cambridge collaborators for making their ASR transcripts available for our work, as well as for useful discussions.

7. REFERENCES

[1] Vassilios Digalakis and Hy Murveit, "GENONES: An algorithm for optimizing the degree of tying in a large vocabulary hidden Markov model based speech recognizer", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 537–540, Adelaide, Australia, 1994.

[2] Hy Murveit, John Butzberger, Vassilios Digalakis, and Mitch Weintraub, "Large-vocabulary dictation using SRI's DECIPHER speech recognition system: Progressive search techniques", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. II, pp. 319–322, Minneapolis, Apr. 1993.

[3] Dan Povey and Phil C. Woodland, "Minimum phone error and i-smoothing for improved discriminative training", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 105–108, Orlando, FL, May 2002.

[4] Ananth Sankar and Chin-Hui Lee, "A maximum-likelihood approach to stochastic matching for robust speech recognition", *IEEE Transactions on Speech and Audio Processing*, vol. 4, pp. 190–202, May 1996.

[5] V. V. Digalakis, D. Rtischev, and L. G. Neumeyer, "Speaker adaptation using constrained estimation of Gaussian mixtures", *IEEE Transactions on Speech and Audio Processing*, vol. 3, pp. 357–366, Sep. 1995.

[6] Leonardo Neumeyer, Ananth Sankar, and Vassilis Digalakis, "A comparative study of speaker adaptation techniques", in J. M. Pardo, E. Enríquez, J. Ortega, J. Ferreiros, J. Macías, and F. J. Valverde, editors, *Proceedings of the 4th European Conference on Speech Communication and Technology*, vol. 2, pp. 1127–1130, Madrid, Sep. 1995.

[7] C. J. Legetter and P. C. Woodland, "Flexible speaker adaptation using maximum likelihood linear regression", *in Proc. SLS Workshop*, 1995.

[8] Ananth Sankar, Leonardo Neumeyer, and Mitchel Weintraub, "An experimental study of acoustic adaptation algorithms", In ICASSP [26].

[9] Wen Wang and Mary Harper, "The superarv language model: Investigating the effectiveness of tightly integrated multiple knowledge sources", *in Proc. Conf. on Empirical methods in Natural Language Processing*, Philadelphia, July, 2002.

[10] Venkata Ramana Rao Gadde, "Modeling word duration for better speech recognition", In NIST [27].

[11] Andreas Stolcke, Harry Bratt, John Butzberger, Horacio Franco, Venkata Ramana Rao Gadde, Madelaine Plauché, Colleen Richey, Elizabeth Shriberg, Kemal Sönmez, Fuliang Weng, and Jing Zheng, "The SRI March 2000 Hub-5 conversational speech transcription system", In NIST [27].

[12] Steven Wegmann, Don McAllaster, Jeremy Orloff, and Barbara Peskin, "Speaker normalization on conversational telephone speech", In ICASSP [26], pp. 339–341.

[13] Hubert Jin, Spyros Matsoukas, Richard Schwartz, and Francis Kubala, "Fast robust inverse transform SAT and multistage adaptation", *in Proceedings DARPA Broadcast News Transcription and Understanding Workshop*, pp. 105–109, Lansdowne, VA, Feb. 1998. Morgan Kaufmann.

[14] Anand Venkataraman and Wen Wang, "Techniques for effective vocabulary selection", *in Proceedings of the 8th European conference on Speech Communication and Technology*, pp. 245–248, Geneva, 2003.

[15] L. Nguyen, R. Schwartz, F. Kubala, and P. Placeway, "Search algorithms for software-only real-time recognition with very large vocabularies", *in Proceedings of the ARPA Workshop on Human Language Technology*, Plainsboro, NJ, Mar. 1993.

[16] Hy Murveit, Peter Monaco, Vassilios Digalakis, and John Butzberger, "Techniques to achieve an accurate real-time large-vocabulary speech recognition system", *in Proceedings of the ARPA Workshop on Human Language Technology*, Plainsboro, NJ, Mar. 1994.

[17] Fuliang Weng, Andreas Stolcke, and Ananth Sankar, "Efficient lattice representation and generation", in Robert H. Mannell and Jordi Robert-Ribes, editors, *Proceedings of the International Conference on Spoken Language Processing*, vol. 6, pp. 2531–2534, Sydney, Dec. 1998. Australian Speech Science and Technology Association.

[18] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER)", *in Proceedings IEEE Automatic Speech Recognition and Understanding Workshop*, pp. 347–352, Santa Barbara, CA, 1997.

[19] Andreas Stolcke, Yochai Konig, and Mitch Weintraub, "Explicit word error minimization in N-best list rescoring", in G. Kokkinakis, N. Fakotakis, and E. Dermatas, editors, *Proceedings of the 5th European Conference on Speech Communication and Technology*, vol. 1, pp. 163–166, Rhodes, Greece, Sep. 1997.

[20] Lidia Mangu, Eric Brill, and Andreas Stolcke, "Searching for consensus to improve recognition output", *in 9th Hub-5 Conversational Speech Recognition Workshop*, Linthicum Heights, MD, Sep. 1998.

[21] Mitch Weintraub, Françoise Beaufays, Ze'ev Rivlin, Yochai Konig, and Andreas Stolcke, "Neural-network based measures of confidence for word recognition", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 887–890, Munich, Apr. 1997.

[22] Anand Venkataraman, Andreas Stolcke, Wen Wang, Dimitra Vergyri, Venkata Ramana Rao Gadde, and Jing Zheng, "An efficient repair procedure for quick transcriptions", *in Proceedings of the International Conference on Spoken Language Processing*, Jeju, Korea, Oct. 2004.

[23] A. Stolcke, H. Franco, R. Gadde, M. Graciarena, K. Precoda, A. Venkataraman, D. Vergyri, W. Wang, J. Zheng, Y. Huang, B. Peskin, I. Bulyko, M. Ostendorf, and K. Kirchhoff, "Speech-to-text research at SRI-ICSI-UW", *in DARPA RT-03 Workshop*, Boston, May 2003, http://www.nist.gov/speech/tests/rt/rt2003/spring/presentations/sri+-rt03-stt.pdf.

[24] H. Y. Chan and Phil Woodland, "Improving broadcast news transcription by lightly supervised discriminative training", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, Montreal, May 2004, To appear.

[25] Julian Odell, *The use of context in large vocabllary speech recognition*, PhD thesis, University of Cambridge, Cambridge, UK, 1995.

[26] *Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, Atlanta, May 1996.

[27] *Proceedings NIST Speech Transcription Workshop*, College Park, MD, May 2000.