Active Learning: Theory and Applications to Automatic Speech Recognition

Giuseppe Riccardi, *Senior Member, IEEE,* and Dilek Hakkani-Tür, *Member, IEEE*

***Abstract*—We are interested in the problem of adaptive learning in the context of automatic speech recognition (ASR). In this paper, we propose an active learning algorithm for ASR. Automatic speech recognition systems are trained using human supervision to provide transcriptions of speech utterances. The goal of Active Learning is to minimize the human supervision for training acoustic and language models and to maximize the performance given the transcribed and untranscribed data. Active learning aims at reducing the number of training examples to be labeled by automatically processing the unlabeled examples, and then selecting the most *informative* ones with respect to a given cost function for a human to label. In this paper we describe how to estimate the confidence score for each utterance through an on-line algorithm using the lattice output of a speech recognizer. The utterance scores are filtered through the *informativeness* function and an optimal subset of training samples is selected. The active learning algorithm has been applied to both batch and on-line learning scheme and we have experimented with different selective sampling algorithms. Our experiments show that by using active learning the amount of labeled data needed for a given word accuracy can be reduced by more than 60% with respect to random sampling.**

***Index Terms*—Acoustic modeling, active learning, language modeling, large vocabulary continuous speech recognition, machine learning.**

I. Introduction

In THE 1990s, there was a large body of research work on data driven algorithms for Large Vocabulary Continuous Speech Recognition (LVCSR). This work has had permanent impact on state-of-the-art automatic speech recognition [1] and stochastic modeling [2]. In those papers, the fundamental assumption is on the nature of the input channel statistics. The assumption is that the performance of the stochastic models are based on the fact that the training examples are drawn randomly from a large sample set $\mathcal{X}$. Moreover, most of these algorithms are trained and tested using the identical and independent distribution (i.i.d.) assumption. This approach leads to models that are by design suited for stationary channels. While this assumption holds for a large number of cases, it has two drawbacks. First it makes inefficient use of data which is expensive to transcribe. Second it restricts the machines behavior to adapt dynamically to nonstationary input channels. In the most recent work the approach to adapt to nonstationary channels has been to adapt the *model* [3]–[6]. Even, in the adaptation framework it is not determined *how* to track time-varying statistics.

In this work we take a fundamentally different approach to train adaptive LVCSRs based on the concept of active learning (AL). Active learning has the distinct advantage of efficiently exploiting transcribed data and thus reduces human effort. Moreover, modeling under the active learning paradigm has the intrinsic capability to adapt to nonstationary events by means of a feedback mechanism in the training algorithm. Active learning can optimize the performance of LVCSRs by selectively sampling the number of examples that maximizes word accuracy. In the next two sections we contrast the traditional method of *passive* learning versus the *active* learning approach. In Section III we define the problem of active learning in its general formulation. In Sections IV and V we show how active learning applies to both acoustic and language modeling for LVCSR. In the last Section VI we give experimental results to support the claims in the previous sections.

II. Passive Learning

The most established method for training acoustic and language models is supervised training. In this case the set of training examples are speech utterances, $\mathbf{x} \in \mathcal{X}$, drawn at random from the set $\mathcal{X}$, which has been selected a-priori and fixed in time[1]. *All* of the examples $\mathbf{x}$ are transcribed by human supervision and the transcriptions are provided with a time delay $\Delta\tau > 0$. This is in contrast with *unsupervised* learning [7]–[10], where a set of training examples is generated automatically without supervision ($\Delta\tau \approx 0, \Delta\chi > 0$). In Fig. 1 we give the architecture of the learning process in the case of supervised learning. The training examples $\mathbf{x} \in \mathcal{X}$ are human labeled and their statistics are used to estimate means (μ) and variances (σ) of the stochastic models (e.g., n-gram, HMMs). The statistical models are evaluated in terms of error rates ε (e.g., Word Error Rate for ASR). In this learning scheme there is no relation between the expected error rate and the set of training examples $\mathcal{X}$. In other words, if a new set of set of training examples $\mathcal{X}'$ are provided, it is not possible to predict if this set would decrease or increase the error rate estimated on $\mathcal{X}$.

This specific type of supervised training is also called *passive* learning [11]. The approach is passive for the following reasons.

- **Selection of training set $\mathcal{X}$.** The training set $\mathcal{X}$ is fixed a-priori and for a given time horizon, ΔT.

Manuscript received July 21, 2003; revised July 21, 2004. The Associate Editor coordinating the review of this manuscript and approving it for publication was Dr. Tanja Schultz.

The authors are with AT&T Labs—Research, Florham Park, NJ 07932 USA (e-mail: dsp3@research.att.com).

Digital Object Identifier 10.1109/TSA.2005.848882

[1]In practice $\mathcal{X}$ is the by-product of the *data collection*, which is usually done via Wizard-of-Oz paradigm or by an automated spoken dialog system.

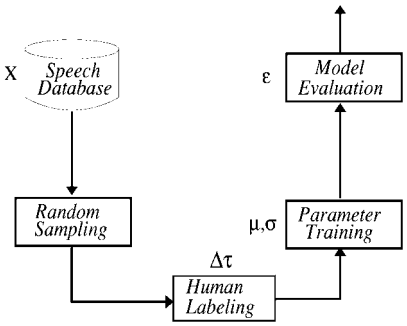

Fig. 1. Supervised passive learning.

- **Selection of labeled training examples**. Training and test sets are sampled randomly from $\mathcal{X}$. *All* examples $\mathbf{x}$ are considered equally *informative* for the purpose of learning. No $\mathbf{x}$ is disregarded or generated automatically.

III. Active Learning

A. Background

The search for effective training data sampling algorithms has been studied in the machine learning research. Previous work in active learning has concentrated on two approaches: certainty-based methods and committee-based methods. In the *certainty-based methods*, an initial system is trained using a small set of annotated examples [12]. Then, the system examines and labels the unannotated examples, and determines the certainties of its predictions of them. The K examples with the lowest certainties are then presented to the labelers for annotation. In the *committee-based methods*, a distinct set of classifiers is also created using the small set of annotated examples [11], [13]. The unannotated instances, whose annotations differ most when presented to different classifiers are presented to the labelers for annotation. In both paradigms, a new system is trained using the new set of annotated examples, and this process is repeated until the system performance converges. A recent committee-based method, which is applicable to multiview problems (i.e., problems with several sets of uncorrelated attributes that can be used for learning) is *co-testing* [14]. In co-testing, the committee of classifiers is trained using different views of the data.

In the language processing research, certainty-based methods have been used for information extraction, and natural language parsing [15]–[17], committee-based methods have been used for text categorization [18], [14]. Our previous work concentrated on using active learning for language model training for ASR [19], [20], and spoken language understanding [21]. Concurrently, [22] has proposed a similar selective sampling approach for ASR, and have shown improvements for training acoustic models on a small vocabulary task.

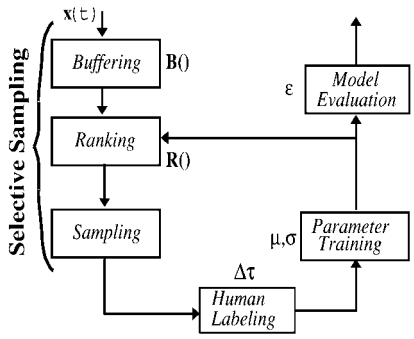

Fig. 2. Supervised active learning architecture. μ and σ are the statistical parameters of acoustic or language models.

B. The Algorithm

Passive learning delegates the burden of estimating *good* models to the estimation techniques (e.g., acoustic or language modeling). In contrast, active learning emphasizes the role of the input selection for the purpose of improving the expected error rate over time.

Active learning is defined in terms of two basic concepts: *selective sampling* and *error rate prediction* as function of the training examples [11]. In Fig. 2 we give the general scheme of active learning machines. In the general formulation, active learning considers the input $\mathbf{x} = \mathbf{x}(t)$ as an example (e.g., speech utterance, feature vector) being sampled at time t. This formulation is suitable for dynamic systems which are sought to track stationary and nonstationary statistics. For $\Delta T = \infty$, we have active learning in the more traditional sense of an *optimization* problem over a-priori set of training examples $\mathcal{X}$ [11]. In the rest of the paper we will consider both scenarios as they apply to automatic speech recognition.

While active learning can act upon each individual sample $\mathbf{x}(t)$, selective sampling is optimized by buffering $(\mathbf{B}(\mathbf{t}))$ the samples over a time horizon, $\Delta T = J$

$$\mathbf{B}(t) = (\mathbf{x}(t), \ldots, \mathbf{x}(t+J)). \tag{1}$$

Selective sampling searches for the subset of examples that are mostly informative to decrease the word error rate ε. In Section VI we address the relation between the J and the first order difference for the error rate, $\Delta(\varepsilon)$.

Let us assume that there exists a function $\Phi(\mathbf{x}(t))$, which computes an estimate of the error rate for each example $\mathbf{x}(t)$. The *informativeness* function $I(\Phi)$ assigns a weight to each training example as a function of its error estimate. In Fig. 3 we have two candidate functions which characterize two different learning strategies. $I_1(\Phi)$ is the uncertainty based linear function and it ranks monotonically all examples starting with the

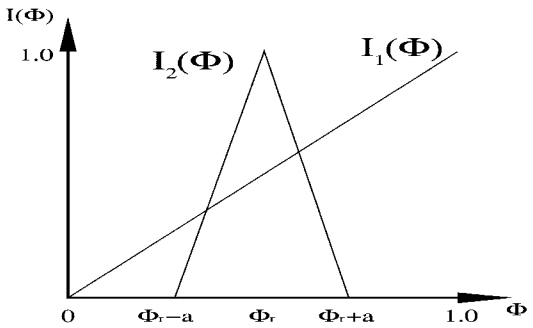

Fig. 3. Characterization of the Informativeness function $I(\Phi)$. The x axis is for the error estimates (ε). The y axis is for the the informativeness function $I(\Phi)$ which could be either linearly related with with $\Phi(I_1)$ or skewed around the random guess estimate $\Phi_r, (I_2)$.

lowest error rate[2]. In this case the informativeness of an example $\mathbf{x}$ will favor the *exploitation* strategy in active learning and expecting high reward from human supervision of poorly predicted samples. However, in general the classes we are learning are not separable and we should expect examples that are either outliers or hard to learn. This leads to an alternate informativeness function $I_2(\Phi)$, which penalizes samples with very high (and low) error rates and emphasizes samples which are estimated (almost) randomly. In Fig. 3, $I_2(\Phi)$ is plotted for the case of random predictions occurring at Φ_r. The triangular shaped informativeness function $I_2(\Phi)$ has only one parameter, $0 \leq a \leq 1$, to be computed based on held-out data. Let us recall here that the examples ranking is dependent on the current model parameters and thus examples may have different informativeness score at different time intervals.

The *informativeness* function $I(\Phi)$ allows us to rank each example $\mathbf{x}$ and associate an integer number r, which spans from 1 to the size of the buffer, K. If more than one independent error estimator is provided then there is the problem of *merging* ranked lists, which is a research topic well studied in statistics and machine learning [23], [24]. The last step of selective sampling (see Fig. 2) is to take the first K examples from the ranked list $(\mathbf{R}(\mathbf{B}(t)))$ and have it labeled (human).[3]

Once the statistics and the error estimator parameters have been updated, the learning loop continues by processing a new batch $(\mathbf{B}())$ of training examples. In general if we are not making any assumptions about the model learning algorithm (*black box* paradigm) there is no guarantee that the expected error rate will decrease as function of the selected samples. There are convergence results for particular learning algorithms such as the perceptron [25], however they do not apply in the general case. In LVCSR, the class of estimation methods is large, ranging from maximum likelihood (ML) to discriminative training (DT). For ASR we will show strong experimental relation between recognition error and the sequence of selectively sampled training sets using ML trained stochastic models.

In the next section we will exploit the active learning concepts and apply them to automatic speech recognition.

[2]Obviously, any strictly monotonic function between $\Phi = 0$ and $\Phi = 1$ will provide one and only example ranking.

[3]Throughout the paper we will use the term *label* and depending on the context it will refer to different types of human supervision. For instance, in text classification text transcriptions are annotated with a label which is one out of a fixed set of allowed labels. In speech recognition, speech utterances are transcribed using a set of guidelines to describe the event contained in them (e.g., spoken words or speech disfluencies).

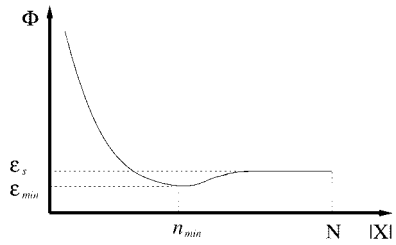

Fig. 4. Minimum sample size and minimum error rate on the learning curve.

IV. Active Learning for ASR

In speech recognition the common training paradigm is the optimization of the word error rate (WER) for a given training and test set, both drawn at random from a fixed $\mathcal{X}$. In this procedure, it is assumed that all training examples are *useful*. In the case of the so-called *data mismatch* there is a performance gap between expected (i.i.d. case) and actual performance (nonstationary case). Traditionally this problem is approached with *model* adaptation techniques with a supervised passive scheme [26], [6]. However, even in the latter case it is not possible to automatically detect the *mismatch* or the time-varying statistics.

Active learning encapsulates the traditional LVCSRs statistical estimation techniques into the framework of adaptive machine learning. It detects automatically the training examples that are difficult to recognize because of data sparseness or nonstationarities. In general it selects, out of all the available speech corpora, only those samples that maximizes word accuracy (1-WER), overcoming the problem of training from outliers or superfluous data. In Fig. 4 we show the typical learning curves of LVCSR systems for a large[4] training set $\mathcal{X}$. We are interested in minimizing the error rate, $\varepsilon_{\min}$, regardless of the amount of samples needed to train the stochastic models. In general if we train from all the available training examples, the error rate will be suboptimal $(\varepsilon_s \geq \varepsilon_{\min})$. This is due to overtraining or to the presence of outliers. From a sample size point of view, $n_{\min}$ is the optimal value and the performance saturation occurs at earlier stages of training $(n_{\min} \leq N)$.

The central concept in active learning is selective sampling from the training set. There are two components, namely the ranking $(\mathbf{R}())$ of each sample in $\mathcal{X}$ and the selection of a set of examples. The latter component controls directly the model update rate. In the next sections we describe the core algorithms for ranking $(\mathbf{R}())$, estimating the error rate and training with AL.

V. Training With Active Learning

A. Word Score Estimation

In the literature, there are two leading methods for confidence score estimation. The first one is based on acoustic measure-

[4]In practice for LVCSRs, 10 K speech utterances are considered a reasonable size of the training set.

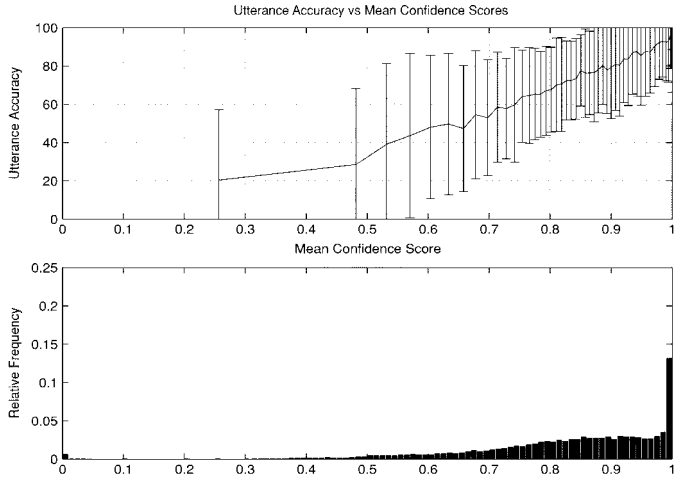

Fig. 5. (Top) Utterance accuracy versus mean confidence scores. For each bin the mean and the standard deviation are plotted. (Bottom) Relative frequency histogram for mean confidence scores.

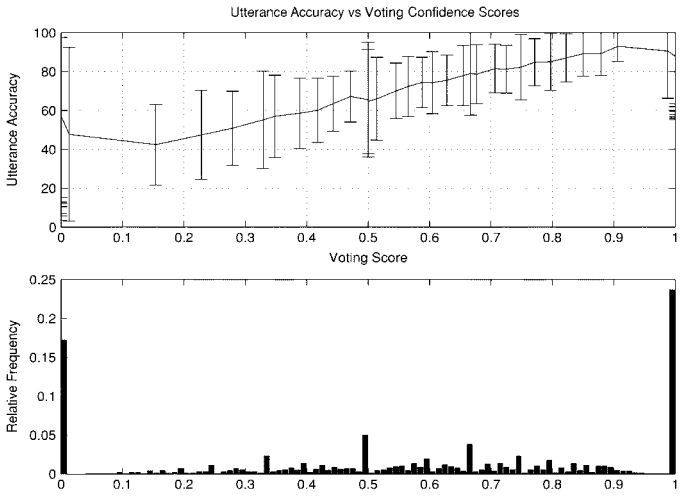

Fig. 6. (Top) Utterance accuracy versus voting confidence scores. for each bin the mean and the standard deviation are plotted. (Bottom) Relative frequency histogram for voting scores.

ments [27] and the other one is based on word lattices [28], [29]. The latter one has the advantage that the probability computation does not require training of an error estimator. There are also hybrid approaches, which use features from the two types of methods [30].

We extract word confidence scores from the lattice output of ASR. A detailed explanation of this algorithm and the comparison of its performance with other approaches is presented in [31]. A summary of the algorithm is as follows.

1) Compute the posterior probabilities for all transitions in the lattice.
2) Extract a path from the lattice (which can be the best, longest or a random path), and call this as the *pivot* of the alignment.
3) Traverse the lattice, and align all the transitions with the pivot, merging the transitions that correspond to the same word (or label) and occur in the same interval (by summing their posterior probabilities).

We use the state times or approximate state locations on the lattice, to align transitions that have occurred at around the same time interval. We call the final structure as *pivot alignment*. We use the word posterior probability estimates on the best path of the pivot alignments as word confidence scores, c_{w_i}, where we use w_i to denote a word and use the notation $c_{w_1^n}$ to represent the confidence score of the word sequence $w_1, \ldots, w_n$.

B. Utterance Score Estimation

For active learning, we used different approaches to obtain utterance level confidence scores from word confidence scores [19], and two of them resulted in a better performance. One approach is to compute the confidence score of an utterance as the arithmetic mean of the confidence scores of the words that it contains

$$c_{w_1^n} = \frac{1}{n}\sum_{i=1}^{n} c_{w_i}. \tag{2}$$

The second approach is using a voting scheme with a threshold. The words with a score less than the threshold do not have any contribution to the utterance score. The words with a confidence score greater than the threshold contribute to the utterance score by 1. The final utterance score is normalized by the utterance length.

$$c_{w_1^n} = \frac{1}{n}\sum_{i=1}^{n} \mathrm{tr}\left(c_{w_i}\right) \tag{3}$$

where

$$\mathrm{tr}\left(c_{w_i}\right) = \left\{ \begin{matrix} 1, & \text{if } c_{w_i} > \text{threshold} \\ 0, & \text{otherwise} \end{matrix} \right\}. \tag{4}$$

In Fig. 5(top) we plot the joint statistics of the mean confidence scores (2) and the corresponding utterance accuracy $(1 - \varepsilon_i)$. The confidence scores are binned uniformly (100 utterances per bin) and we plot the mean and standard variation of the utterance accuracy within each bin. In Fig. 5 (bottom) we plot the relative frequency histogram of the utterance confidence score. As can be seen in Fig. 5 the utterance score is linearly correlated to the utterance accuracy.

In Fig. 6 we plot the joint statistics of the voting confidence scores (3) and the corresponding utterance accuracy $(1 - \varepsilon_i)$ using the same binning procedure used in Fig. 5.

Both figures are the experimental evidence for the *informativeness* functions in Fig. 3.

C. Algorithm

The algorithm for active learning for ASR is depicted in Fig. 7. In the figure, the solid lines show the operation flow, and the dashed lines show data flow. As the initialization step, we first train a speech recognizer, using a small set of transcribed data, S_t. The S_t can also include off-the-shelf speech corpora. Using the initial ASR models, we automatically transcribe the utterances that are candidates for transcription, S_u (see $\mathbf{B}()$ in Fig. 2). We then compute lattice based confidence scores for

each word and use either (2) or (3) to assign an utterance based score. We apply the *informativeness* function $I(\Phi)$ to predict which candidates are more likely to reduce the word error rate. We provide human transcriptions solely for the selected utterances from S_u. This set of K utterances are denoted by S_k in the figure. We then add the transcribed utterances to S_t and exclude them from S_u. We iterate this process as long as there are additional untranscribed utterances and we stop, if the WER on the development test set has converged.

D. Error Convergence

By treating the model learning algorithms as black boxes, in general there is no guarantee to minimize the error rate monotonically. For instance for stochastic language modeling, most of the state-of-the-art techniques are based on ML estimates of n-gram counts [32]. In acoustic modeling, there is a quite effective set of techniques that address the problem of minimizing the word error rate [1]. In both cases, it is an open research problem how to combine the selective sampling into the estimation algorithm. It is straightforward to show that an arbitrarily biased sampling bears no correlation with word error rate. However, in the case of ML based model estimation we will show an empirical monotonic correlation between recognition error rate and two sampling methods: random and selective sampling of training sets using ML trained stochastic models.

VI. Experimental Results

We performed a series of experiments to verify that the posterior probabilities of the ASR pivot alignments can be used to select more informative utterances to transcribe. For all these experiments, we used utterances from the *How May I Help You?*^{SM} human-machine speech dialog database [33]. The language models used in all our experiments are trigram models based on Variable Ngram Stochastic Automata (VNSA) [34]. The acoustic models are subword unit based, with triphone context modeling.

A. Training and Test Data

The initial set of transcribed utterances, which is used to train the initial acoustic and language models consists of 1000 utterances (10 722 words). The additional set of transcription candidate utterances consists of 26 963 utterances (307 649 words). The test data consists of 1 000 utterances (10 646 words). In the experiments, where we retrained the acoustic model, we used a 1 000 utterance (11 515 words) development test set to tune the parameters (the number of mixtures, etc.).

B. Active Learning

Using the initial ASR acoustic and language models, we generated lattices and pivot alignments for our additional training data, and computed the confidence scores for words and utterances. We ran the algorithm for only a single iteration, with K equal to the additional training set size, and sorted the data in the order of increasing usefulness for ASR. We then incrementally trained acoustic and language models, every 2000 utterances (100, 250, 500, and 1000 utterances at the initial points),

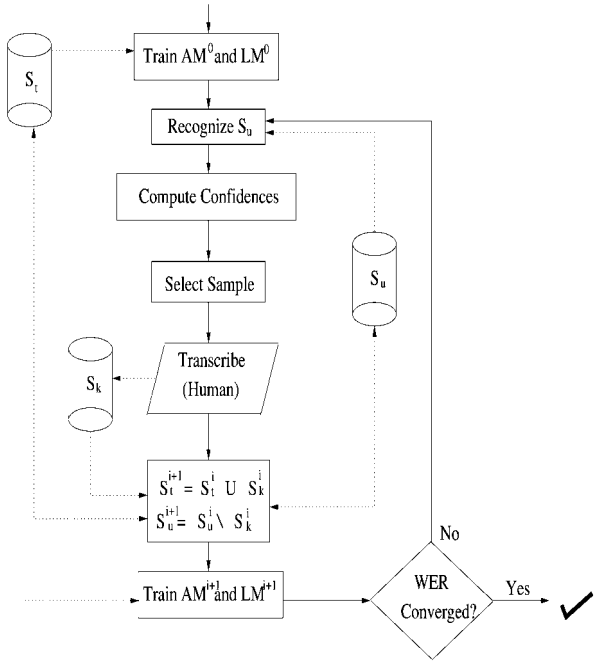

Fig. 7. The algorithm.

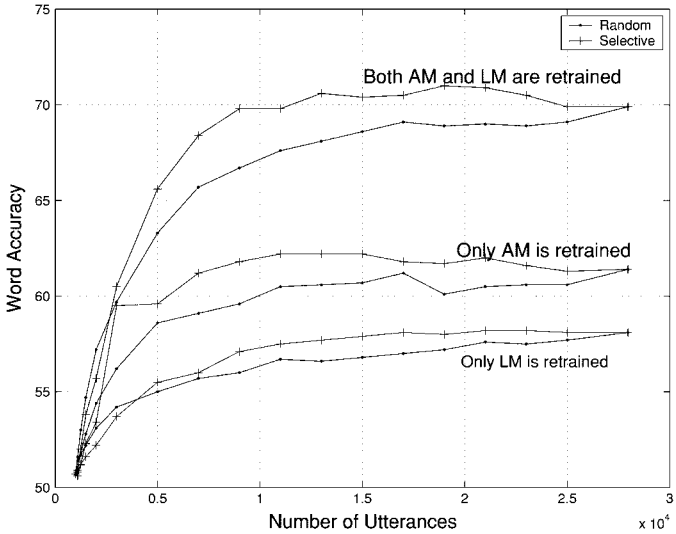

Fig. 8. Word accuracy learning curves.

and generated learning curves for Word Accuracy $(=1-\varepsilon)$, which are presented in Fig. 8. In that figure, there are three sets of curves for random sampling and selective sampling. In the top experiment, both the acoustic and language models are retrained as we have more transcribed training data. In the center experiment, the acoustic model is retrained for each set of new K examples, while the language model is fixed and trained from the initial set. In the bottom experiment, the language model is retrained for each set of newly selected K examples and the acoustic model is fixed and trained from the initial set. In all of them, the random and selective sampling curves meet at the same point, as all the data is used in that case $(\Delta T = \infty)$. Of course, in practice active learning would stop where the performance saturates. We plot the results using the arithmetic mean of the word confidence scores (Φ is the mean function in (2)) and

$I_2(\Phi)$, which performed slightly better than $I_1(\Phi)$.[5] From these curves, we see that selective sampling is effective in minimizing the amount of labeled data necessary to achieve best word accuracy. When we retrained both the acoustic and language models, the best performance with random sampling was achieved using all of the training data (27 963 utterances). We achieved the same word accuracy (69.9%) with selective sampling and using 68% less data (with around 9 000 utterances). Therefore, by selective sampling, it is possible to speed up the learning rate of ASR with respect to the amount of labeled transcriptions. We also achieved a higher accuracy with active learning (71.0%) than using all the data, when we used 19 000 utterances. At each point on these curves, we selected the acoustic model parameters that maximize the word accuracy on the development test set, and plotted the real test set word accuracy with these parameters.

One reason for the better learning is that, with active learning, we achieve a faster learning rate for new words and new n-grams. Fig. 9 shows the vocabulary size learning curves for random and selective sampling. As can be seen from the figure, we detect new n-grams at a higher rate with selectively sampling as compared to random sampling.

For a given acoustic channel (e.g., telephone), we are interested to evaluate the performance of AL for learning novel domain language. In this experimental scenario we use an off-the-shelf acoustic model trained on the same acoustic channel from off-the-shelf speech corpora. We used the set in the previous experiment to train the initial language model and used these models to select examples from the buffer. The random and selective sampling learning curves are plotted in Fig. 10. In this experiment, the best performance with random sampling was again achieved using all of the training data (27 963 utterances). We achieved the same word accuracy (68.1%) with selective sampling and using 64% less data (with around 10 000 utterances). We achieved the best accuracy with active learning (68.6%) when we used 13 000 utterances (less than half of all the data).

We have simulated the dynamic case ($\mathbf{x} = \mathbf{x}(t)$) by sorting the utterances according to their time stamps. The time span for the entire training set is three months of live recordings from the "How May I Help You?" system. We have buffered 1 000 utterances at time instant t ($J = 1\,000$) and selectively sampled 500 utterances ($K = 500$), and transcribed them. We discarded the remaining 500 utterances, and used the new set of transcribed utterances in the selective sampling for $t+1$. In this experiment we used the acoustic model from the previous experiment (see Fig. 10). Fig. 11 depicts the results of such an experiment. In this case, the selective sampling learning curve ends at around 14 500 utterances, as we are discarding half of the utterances in the buffer at each instant $\mathbf{B}(t)$. According to this plot, we again achieve the best accuracy with random sampling, using 64% less transcribed utterances with selective sampling. With only half of the additional data, we achieve a word accuracy of 68.9%, which is 0.8% points better than the accuracy achieved using all the data, and 1.7% points better than using half of the data with random sampling.

[5]We also used the normalized utterance likelihood as a sampling criterion, and it gave inferior performance.

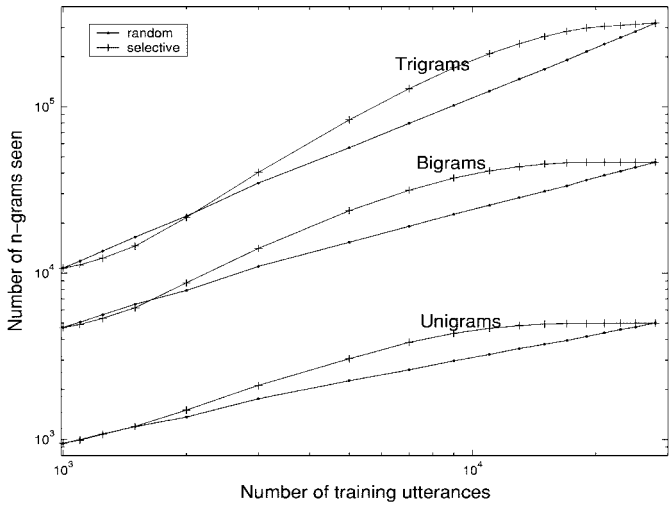

Fig. 9. Vocabulary size learning curves.

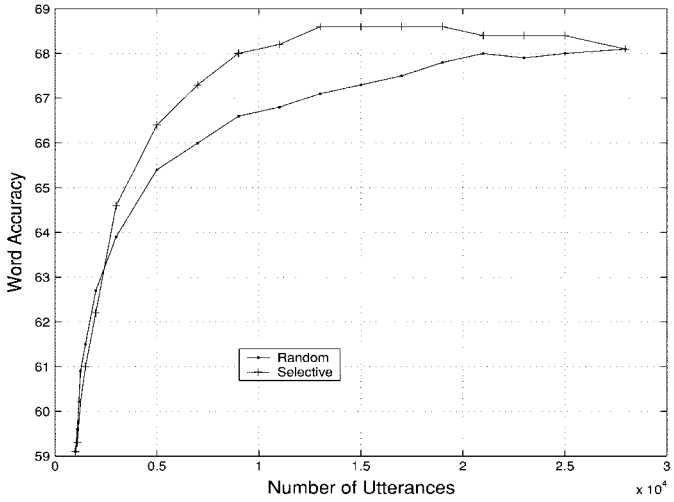

Fig. 10. Learning curves for novel domain language (off-the-shelf acoustic model).

C. Buffer Size

The buffering of the input $\mathbf{x}(t)$ in active learning is designed to increase the probability of minimizing the error rate. To show this, we plotted the effect of buffer size, J, to the active learning performance in Fig. 12. We selected three set sizes for transcription, $K = \{500, 1\,000, 2\,000\}$ utterances. We selectively sampled these utterances from buffers of different sizes. As can be seen from the figure, the word accuracy is maximized for a buffer of size three times the selection size K. This shows that there is a maximum performance we can achieve using a given transcribed data set size. The performance degrades as we increase the buffer size, as our algorithm is not outlier-proof, and the percentage of outliers increases as we increase the buffer size. But in all cases, the performance is better than random sampling, which corresponds to $i = 1$ in Fig. 12.

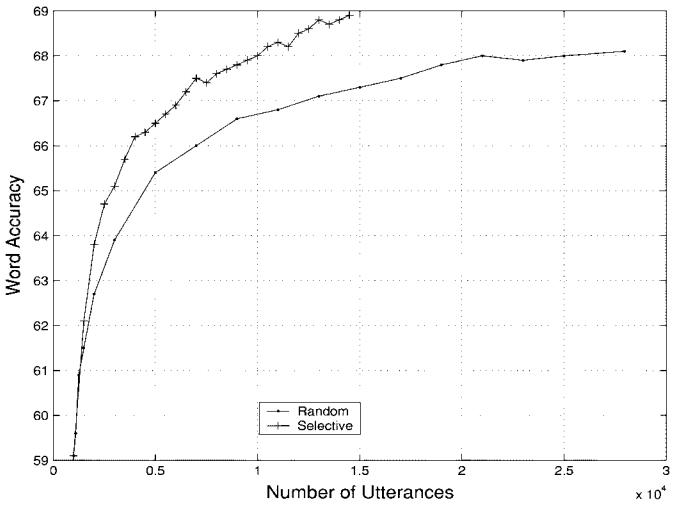

Fig. 11. Word accuracy learning curves for the dynamic case of AL ($\mathbf{x} = \mathbf{x}(t)$). Examples are buffered ($\mathbf{B}(t)$) according to the time stamp of the data collection.

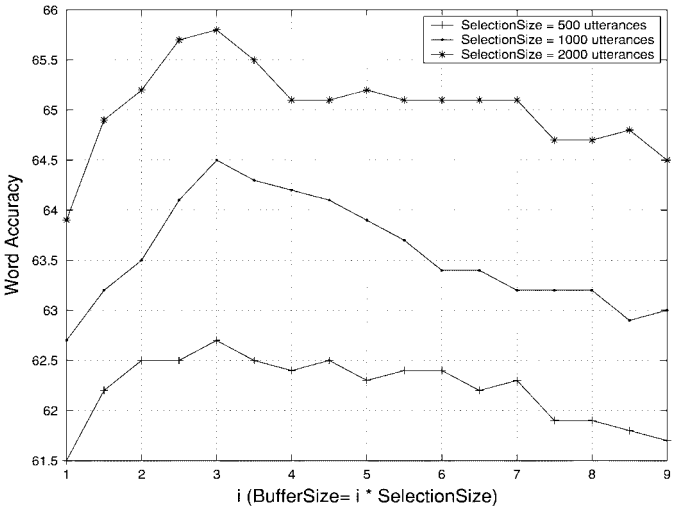

Fig. 12. Effect of buffer (J) size on active learning performance.

VII. Conclusion

In this paper, we have proposed a novel approach to automatic speech recognition based on active learning. Active learning makes efficient use of data which is expensive to transcribe. Moreover, active learning has the built-in feature to adapt to nonstationary events by means of feedback mechanism in the training algorithm. Active learning can also be seen as an optimization algorithm that selects the training examples that optimize the test set word accuracy. Our experiments show that by using active learning the amount of labeled data needed for a given word accuracy can be reduced by more than 60% with respect to random sampling and word accuracy is improved as well.

Acknowledgment

The authors would like to thank M. Rahim, J. Wilpon, and R. Cox for their continued support on this research topic. They would also like to thank G. Tur and M. Saraclar for their technical help and useful discussions.

References

[1] L. Rabiner and B. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
[2] F. Jelinek, *Statistical Methods for Speech Recognition*. Cambridge, MA: MIT Press, 1997.
[3] V. Digalakis, D. Rtischev, and L. G. Neumeyer, "Speaker adaptation using constrained estimation of Gaussian mixtures," *IEEE Trans. Speech Audio Processing*, vol. 3, no. 5, 1995.
[4] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Comput. Speech Lang.*, vol. 9, no. 2, 1995.
[5] M. Federico, "Bayesian estimation methods for n-gram language model adaptation," in *Proc. Int. Conf. Speech and Language Processing*, Philadelphia, PA, 1996, pp. 240–243.
[6] G. Riccardi and A. L. Gorin, "Stochastic language adaptation over time and state in a natural spoken dialog system," *IEEE Trans. Speech Audio Processing*, vol. 8, no. 1, pp. 3–9, 2000.
[7] R. Gretter and G. Riccardi, "On-line learning of language models with word error probability distributions," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, 2001.
[8] T. Kemp and A. Waibel, "Learning to recognize speech by watching television," *IEEE Intell. Syst.*, pp. 51–58, 1999.
[9] G. Zavaliagkos and T. Colthurst, "Utilizing untranscribed training data to improve performance," in *Proc. Broadcast News Transcription and Understanding Workshop*, 1998, pp. 301–305.
[10] A. Stolcke, "Error modeling and unsupervised language modeling," in *Proc. 2001 NIST Large Vocabulary Conversational Speech Recognition Workshop*, Linthicum, MD, 2001.
[11] D. Cohn, L. Atlas, and R. Ladner, "Improving generalization with active learning," *Mach. Learn.*, vol. 15, pp. 201–221, 1994.
[12] D. D. Lewis and J. Catlett, "Heterogeneous uncertainty sampling for supervised learning," in *Proc. 11th Int. Conf. Machine Learning*, 1994, pp. 148–156.
[13] I. Dagan and S. P. Engelson, "Committee-based sampling for training probabilistic classifiers," in *Proc. 12th Int. Conf. Machine Learning*, 1995, pp. 150–157.
[14] I. A. Muslea, "Active Learning with Multiple Views," Ph.D. dissertation, Univ. Southern California, Los Angeles, 2000.
[15] C. Thompson, M. E. Califf, and R. J. Mooney, "Active learning for natural language parsing and information extraction," in *Proc. 16th Int. Conf. Machine Learning*, 1999, pp. 406–414.
[16] M. Tang, X. Luo, and S. Roukos, "Active learning for statistical natural language parsing," in *Proc. 40th Anniversary Meeting of Association for Computational Linguistics (ACL-02)*, Philadelphia, PA, 2002, pp. 120–127.
[17] R. Hwa, "On minimizing training corpus for parser acquisition," in *Proc. 5th Computational Natural Language Learning Workshop*, San Francisco, CA, 2001, pp. 84–89.
[18] R. Liere, "Active Learning with Committees: An Approach to Efficient Learning in Text Categorization Using Linear Threshold Algorithms," Ph.D. dissertation, Oregon State Univ., Portland, 2000.
[19] D. Hakkani-Tür, G. Riccardi, and A. Gorin, "Active learning for automatic speech recognition," in *Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, Orlando, FL, 2002, pp. 3904–3907.
[20] G. Riccardi and D. Hakkani-Tür, "Active and unsupervised learning for automatic speech recognition," in *Proc. EUROSPEECH*, 2003.
[21] G. Tur, R. E. Schapire, and D. Hakkani-Tür, "Active learning for spoken language understanding," in *Proc. ICASSP*, Hong Kong, May 2003.
[22] T. M. Kamm and G. G. L. Meyer, "Selective sampling of training data for speech recognition," in *Proc. Human Language Technology Conf.*, San Diego, CA, 2002.
[23] D. Critchlow, *Metric Methods for Analyzing Partially Ranked Data*. New York: Springer-Verlag, 1980.
[24] G. Lebanon and J. Lafferty, "Cranking: Combining rankings using conditional probability models on permutations," in *Proc. 19th Int. Conf. Machine Learning*, 2002.
[25] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby, "Improving generalization with active learning," *Mach. Learn.*, vol. 15, pp. 201–221, 1994.
[26] J. L. Gauvain and C. H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Trans. Speech Audio Processing*, vol. 2, no. 2, pp. 291–298, 1994.

[27] R. C. Rose, B. H. Juang, and C. H. Lee, "A training procedure for verifying string hypotheses in continuous speech recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 1995, pp. 281–284.
[28] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: Word error minimization and other applications of confusion networks," *Comput. Speech Lang.*, vol. 14, no. 4, pp. 373–400, 2000.
[29] D. Falavigna, R. Gretter, and G. Riccardi, "Acoustic and word lattice based algorithms for confidence scores," in *Proc. Int. Conf. Spoken Language Processing*, 2002.
[30] R. Zhang and A. Rudnicky, "Word level confidence annotation using combinations of features," in *Proc. 7th Eur. Conf. Speech Communication and Technology*, 2001, pp. 2105–2108.
[31] D. Hakkani-Tür and G. Riccardi, "A general algorithm for word graph matrix decomposition," in *Proc. ICASSP*, Hong Kong, 2003.
[32] R. De Mori, *Spoken Dialogues with Computers*. New York: Academic, 1998.
[33] A. Gorin, A. Abella, T. Alonso, G. Riccardi, and J. H. Wright, "Automated natural spoken dialog," *IEEE Computer*, vol. 35, pp. 51–56, 2002.
[34] G. Riccardi, R. Pieraccini, and E. Bocchieri, "Stochastic automata for language modeling," *Comput. Speech Lang.*, vol. 10, pp. 265–293, 1996.

Giuseppe Riccardi (M'96-SM'04) received the Laurea degree in electrical engineering and Master degree in information technology, in 1991, from the University of Padua and CEFRIEL Research Center, respectively. In 1995, he received the Ph.D. degree in electrical engineering from the Department of Electrical Engineering, University of Padua.

From 1990 to 1993, he collaborated with Alcatel-Telettra Research Laboratories Milan, Italy, and investigated algorithms for speech and audio coding for medium-low bit rates for the half-rate GSM standard (New European Digital Trunking System). In 1993, he spent two years as research fellow at AT&T Bell Laboratories investigating automata learning for stochastic language modeling for speech recognition and understanding. In 1996, he joined AT&T Labs-Research where he is currently Technology Leader. His research on stochastic finite state machines for speech and language processing has been applied to a wide range of domains for task automation. He participated at the creation of the state-of-the-art AT&T spoken language system used in the 1994 DARPA ATIS evaluation. He has been pioneering the speech and language research in spontaneous speech within the "*How May I Help You?*" research program. His research on learning finite state automata and transducers has lead to the creation of the first large scale finite state chain decoding for machine translation (*Anuvaad*). He has co-authored more than 60 papers in the field of speech and audio coding, speech recognition and understanding and machine translation. His current research interests are stochastic language modeling, language understanding, spoken dialogue, language acquisition and machine translation. He is on the Editorial Board of the *ACM Transactions of Speech and Language*.

Dr. Riccardi has been on the scientific committees of EUROSPEECH, ICASSP, and ACL. He co-organized the IEEE ASRU Workshop in 1993, 1999, and 2001. Dr. Riccardi is the Guest Editor of the IEEE Special Issue on Speech-to-Speech Machine Translation. He is senior member of ACL and the New York Academy of Science.

Dilek Hakkani-Tür (M'00) received the B.Sc. degree in computer engineering from Middle East Technical University in 1994, and the M.Sc. and Ph.D. degrees from the Department of Computer Engineering, Bilkent University, in 1996 and 2000, respectively. Her Ph.D. thesis is on statistical language modeling for agglutinative languages.

She worked on machine translation during her visit to the Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA, in 1997, and to the Computer Science Department, Johns Hopkins University, Baltimore, MD, in 1998. In 1998 and 1999, she visited the Speech Technology and Research Labs, SRI International, and worked on lexical and prosodic information for information extraction from speech. Her research interests include natural language and speech processing, spoken dialog systems, and machine learning. She co-authored more than 20 papers in natural language and speech processing. She is a Senior Technical Staff Member, AT&T Labs.-Research, Florham Park, NJ.

Dr. Hakkani-Tür is a member of the Association for Computational Linguistics.