An Iterative Relative Entropy Minimization-Based Data Selection Approach for n-Gram Model Adaptation

Abhinav Sethy, Panayiotis G. Georgiou, *Member, IEEE*, Bhuvana Ramabhadran, and Shrikanth Narayanan, *Senior Member, IEEE*

***Abstract*—Performance of statistical n-gram language models depends heavily on the amount of training text material and the degree to which the training text matches the domain of interest. The language modeling community is showing a growing interest in using large collections of text (obtainable, for example, from a diverse set of resources on the Internet) to supplement sparse in-domain resources. However, in most cases the style and content of the text harvested from the web differs significantly from the specific nature of these domains. In this paper, we present a relative entropy based method to select subsets of sentences whose n-gram distribution matches the domain of interest. We present results on language model adaptation using two speech recognition tasks: a medium vocabulary medical domain doctor-patient dialog system and a large vocabulary transcription system for European Parliamentary Plenary Speeches (EPPS). We show that the proposed subset selection scheme leads to performance improvements over state of the art speech recognition systems in terms of both speech recognition word error rate (WER) and language model perplexity (PPL).**

***Index Terms*—Data selection, language model adaptation, relative entropy.**

I. Introduction

THERE is a growing interest in using the World Wide Web (WWW) as a corpus for training models for natural language processing (NLP) tasks [1]–[3]. One common component of many statistical NLP systems which can benefit from the use of web as a corpus is the n-gram language model. The n-gram model provides an estimate of the probability of a word sequence under Markovian assumptions. In speech recognition applications, the n-gram model is frequently used to provide a prior for decoding the acoustic sequence. The n-gram model is trained from counts of word sequences seen in a corpus and hence its quality depends on the amount of training data as well as the degree to which the training statistics represent the target application.

Text harvested from the web and other large text collections such as the English Gigaword [4] corpus provides a good resource to supplement the in-domain data for a variety of applications [5], [6]. However, even with the best queries and text collection schemes, both the style and content of the data acquired tend to differ significantly from the specific nature of the domain of interest. For example, a speech recognition system for spoken dialog applications requires conversational style text for the underlying language models whereas most of the data on the web is written style. To benefit from a generic corpora, we need to identify subsets of text relevant to the target application. In most cases we have a set of in-domain example sentences available to us which can be used in a semi-supervised [7], [8] fashion to identify the text relevant to the application of interest. The dominant theme in recent research literature for achieving this is the use of various rank-and-select schemes for identifying sentences from the large generic collection which match the in-domain data [5], [6]. The central idea behind these schemes is to rank order sentences in terms of their match to the seed in-domain set and then select top sentences. Rank-and-select filtering schemes select individual sentences on the merit of their match to the in-domain model. As a result, even though individual sentences might be good in-domain examples, the overall distribution of the selected set is biased towards the high probability regions of the distribution.

In this paper, we build on our work in [9] and present an improved incremental selection algorithm which compares the distribution of the selected set and the in-domain examples by using a relative entropy (R.E.) criterion at each step. Section II presents several methods for data selection against which the proposed scheme is benchmarked. The proposed algorithm is described in Section III. A brief description of the setup used to build the large corpus used in our experiments and other implementation details is given in Section IV. To validate our approach, we present and compare the performance gains achieved by the proposed approach on two automatic speech recognition (ASR) systems. The first system is a medium vocabulary system for doctor-patient conversations in English [10]. The second system is a large vocabulary transcription system for European parliamentary speeches [11]. Experimental results are provided in Section V. We conclude with a summary of this work and directions for future research.

II. Rank-and-Select Methods for Text Filtering

In recent literature, the central idea behind text data selection schemes for using generic corpora to build language models, has been to use a scoring function that measures the similarity of each observed sentence in the corpus to the domain of interest (in-domain) and assign an appropriate score. The subsequent step is to set a threshold in terms of this score or the number of

Manuscript received August 10, 2007; revised July 16, 2008. Current version published December 11, 2008. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Mary P. Harper.

The authors are with the Signal and Image Processing Institute, University of Southern California, Los Angeles, CA 90089-2564 USA (e-mail: sethy@usc.edu).

Digital Object Identifier 10.1109/TASL.2008.2006654

top scoring sentences, usually done on a heldout data set, and use this threshold as a criterion in the data selection process. A dominant choice for a scoring function is in-domain model perplexity [5], [12] and variants involving comparison to a generic language model [13], [14]. A modified version of the BLEU metric which measures sentence similarity in machine translation has been proposed by Sarikaya [6] as a scoring function. Instead of explicit ranking and thresholding, it is also possible to design a classifier to Learn from Positive and Unlabeled examples (LPU) [15]. In LPU, a binary classifier is trained using a subset of the unlabeled set as the negative or noise set and the in-domain data as the positive set. The binary classifier is then used to relabel the sentences in the corpus. The classifier can then be iteratively refined by using a better and larger subset of the sentences labeled in each iteration. For text classification, support vector machine (SVM)-based classifiers are shown to give good classification performance with LPU [15].

Ranking-based selection has some inherent shortcomings. Rank ordering schemes select sentences on individual merit. Since the merit is evaluated in terms of the match to in-domain data, there is a natural bias towards selecting sentences which already have a high probability in the in-domain text. Adapting models on such data has the tendency to skew the distribution towards regions in the in-domain data that are highly probable. An illustration of this is short sentences containing the word "okay" such as "okay," "yes okay," "okay okay" which were very frequent in the in-domain data for the doctor-patient interaction task. Perplexity and other similarity measures assign a high score to all such examples, boosting the probability of these words even further. In contrast, other pertinent sentences seen rarely in the in-domain data such as "Can you stand up please?" receive a low rank and are more likely to be rejected. Simulation results provided in [9] show the skew towards high probability regions clearly.

III. Data Selection Using Relative Entropy

In order to achieve an unbiased selection of data, we proposed an iterative text selection algorithm based on relative entropy [9]. The idea is to select a sentence if adding it to the already selected set of sentences reduces the relative entropy with respect to the in-domain data distribution.

Stolcke [16] introduced relative entropy as a measure for pruning back-off n-gram language models. In relative entropy-based pruning of n-gram language models, a pruning threshold is set for the relative entropy between the n-gram distribution with history $w_1, \ldots, w_{n-1}$ and the back-off n-gram distribution with history $w_2, \ldots, w_{n-1}$. Higher order n-grams which have low relative entropy with respect to the lower order back-off n-grams are discarded. In this paper, we provide a data selection algorithm based on R.E. minimization that serves a complimentary goal to R.E. based n-gram model pruning. The data selection algorithm aims at finding a good subset of data for building language models while the goal of R.E. based pruning is to find a compact n-gram model which closely matches the unpruned model.

A. Core Algorithm

In this section, we derive the proposed R.E. based data selection algorithm when the in-domain data is modeled using an unigram LM. A detailed derivation for the general case, when the in-domain data is modeled using a back-off n-gram language model is included in the Appendix. Let us define the following symbols.

w	Word.
V	Vocabulary of the in-domain model.
$P(w)$	The language model built from the in-domain data.
$C(w)$	The count of word w in the already selected text.
$N = \sum_{w \in V} C(w)$	The total number of words in the text already selected.
$c(w)$	The count of word w in the sentence being considered for selection.
$n = \sum_{w \in V} c(w)$	The number of words in the sentence.

[1]The skew divergence [17] of the maximum-likelihood estimate of the language model of the selected sentences to the initial model $P(w)$ is given by

$$D = \sum_{w \in V} P(w) \ln \frac{P(w)}{\beta P(w) + \alpha C(w)/N} \tag{1}$$

where $\beta = 1 - \alpha$.

The skew divergence is a smoothed version of the Kullback–Leibler (KL) distance with the alpha parameter denoting the smoothing influence of model $P(w)$ on the current maximum-likelihood (ML) model. When $\alpha = 1$, the skew divergence expression is equivalent to the KL distance. Using skew divergence in place of the KL distance was useful in improving the data selection especially in the initial iterations where the counts $C(w)$ are low and the ML estimate $C(w)/N$ changes rapidly. If a sentence is selected to be included in the language model, the updated divergence is given by

$$D^+ = \sum_{w \in V} P(w) \ln \frac{P(w)}{\beta P(w) + \alpha \left(C(w) + c(w)\right)/(N+n)} \tag{2}$$

If a sentence is not selected, then the model parameters and the divergence measure remain unchanged.

Direct computation of divergence using the above expressions for every sentence in a large corpus has a high computational cost since $O(V)$ computations per sentence are required. The number of sentences can be very large, easily on the order 10^8 to 10^9, which makes the total computation cost for even moderate vocabularies (approximately 10^5) large.

However, given the fact that $c(w)$ is sparse, we can split the summation D^+ into equations (3) and (4), as shown at the bottom of the next page.

[1]The in-domain model $P(w)$ is usually represented by a linear interpolation of n-gram LMs built from different in-domain text corpora available for the task.

Intuitively, the term T_1 accounts for the scaling of the ML probability estimates when the denominator in the estimate $C(w)/N$ increases from N to $N+n$ for all words w in the vocabulary. The term T_2 accounts for the increase in probability for words seen in the sentence where the numerator in the ML estimate increases from $C(w)$ to $C(w)+c(w)$. Equation (3) makes the computation of the stepwise changes in divergence tractable by reducing the required computations to the number of words in a sentence n, instead of a summation over all the words in the vocabulary, i.e., $|V|$ computations. The approximation in (4) is valid if the number of total words selected is significantly larger than the number of words expected to be seen in a single sentence ($N \gg n$). As we describe in the next subsection on initialization, in the beginning of the data selection process, the counts $C(w)$ are initialized in a manner such that $N \gg n$. As the data selection process selects more data, N increases reducing the approximation error further.

B. Initialization

We use the following bootstrap strategy for initializing the counts $C(w)$.

- Choose a random subset (without replacement) of the adaptation data. The size of the random subset is taken to be the same as the size of the in-domain set.
- Initialize $C(w)$ with the count of word w in the random subset. The counts are incremented by 1 to ensure non zero $C(w)$.
- The counts initialized in the previous step are used to select data using the alpha skew divergence criterion presented above.
- $C(w)$ is reset to the count of the word w in the selected set. The counts are incremented by 1 to ensure non zero $C(w)$.

$C(w)$ should be non zero for ensuring finite value of T_2. In general, we have observed that in comparison to uniform initialization or initialization from a random subset, we are able to reduce the size of the selected data set by 10%–15% using the two step initialization technique with no loss in performance either in perplexity or WER.

C. Alpha Parameter

The alpha parameter in (3) controls the smoothing influence of the in-domain language model. The motivation behind this smoothing was to make the relative entropy function behave smoothly during the initial part of data selection. For this purpose, a high value of alpha in the range 0.95–1.0 was found to give good results on the two tasks described in this paper (Section V). The performance of the algorithm was not sensitive to the choice of alpha in this range. In general, a low value of alpha reduces the number of sentences selected (when $\alpha = 0$, no sentence will be selected).

D. Randomization and Multiple Passes

The proposed algorithm is sequential and greedy in nature and can benefit from randomization of the order in which the corpus is scanned. We generate random permutations of the sentences and select the union of the set of sentences identified for selection in each permutation. Sentences that have already been included in more than two permutations are skipped during the selection process, thus forcing the selection of different sets of sentences. After each permutation and data selection iteration, we build a language model from the union of the data selected and compute perplexity on the heldout data set. The heldout set perplexity is used as a stopping criterion to fix the number of permutations. If the perplexity increases on addition of data selected after a random permutation, no further permutations are carried out. For the purpose of fixing the number of random permutes in our experiments, we used a trigram language model with the same vocabulary as the in-domain model.

E. Smoothing

Smoothing [18] can be used after a certain fixed number of sentences are selected to modify the counts of the selected text $C(w)$. We have experimentally found out that Good–Turing smoothing after selection of every 500 K words is sufficient for the tasks considered in this paper. The impact of smoothing was not seen to be significant to warrant further exploration.

$$
\begin{aligned}
D^{+} &= \sum_{w\in V} P(w)\ln P(w) - \sum_{w\in V} P(w)\ln\left(\beta P(w) + \frac{\alpha\,(C(w)+c(w))}{N+n}\right) \\
&= D - \sum_{w\in V} P(w)\ln N + \sum_{w\in V} P(w)\ln\left(\beta P(w)N + \alpha C(w)\right) - \sum_{w\in V} P(w)\ln\left(\beta P(w) + \frac{\alpha\,(C(w)+c(w))}{N+n}\right) \qquad (3) \\
&= D + \underbrace{\ln\frac{(N+n)}{N}}_{T_1} - \underbrace{\sum_{w\in V, c(w)\neq 0} P(w)\ln\frac{\beta P(w)(N+n)+\alpha\,(C(w)+c(w))}{\beta P(w)N+\alpha C(w)}}_{T_2} \\
&\quad - \underbrace{\sum_{w\in V, c(w)=0} P(w)\ln\frac{\beta P(w)(N+n)+\alpha C(w)}{\beta P(w)N+\alpha C(w)}}_{\approx 0} \qquad (4)
\end{aligned}
$$

F. Extension to n-Gram Models

As mentioned earlier, we have introduced the data selection algorithm using unigram models to represent the in-domain data set. The extension of this R.E.-based data selection algorithm to a more general, back-off n-gram model is presented in the Appendix. The computation time of the algorithm depends on the order of the n-gram model used in the data selection procedure. The number of computations required grows linearly with the total number of n-grams in the language model. In general, the total number of n-grams grows exponentially with the order of the model, making the computational cost an exponential function of the language model order (Section V). For initialization of the back-off n-gram based data selection algorithm, we use a random subset of the data selected using a unigram model.

Finally, the selected data is then merged with the in-domain data set to build a language model. The choice of the order and vocabulary of this language model can be different from the order, vocabulary and choice of smoothing method used in the model that was used in the data selection procedure.

In the next section, we describe our infrastructure for collecting the large corpus used in our experiments (Section V) and cover key implementation details of the proposed algorithm.

IV. Implementation Details and Data Collection

The vast text resources available over the WWW were crawled to build the large text corpora used in our experiments. Queries for downloading relevant data from the web were generated using a technique similar to [5] and [13]. An in-domain language model was first generated using the training material and compared to a generic background model of English text [13] to identify the terms which would be useful for querying the web. For every n-gram n in the language model, we calculated the weighted ratio $p(n)\ln(p(n)/q(n))$ where p is the in-domain model and q is the background model. The top-scoring trigrams, bigrams, and unigrams were selected as query terms in that order. The set of URLs returned by our search were downloaded and the non-text files were deleted. The HTML files were converted to text by stripping off tags. The converted text typically does not have well defined sentence boundaries. We found that using an off-the-shelf maximum entropy-based sentence boundary detector[2] [19], seemed to improve sentence boundaries. Sentences and documents with high out-of-vocabulary (OOV) rates were rejected as noise to keep the converted text clean.

As a pre-filtering step, we computed the perplexity of the downloaded documents with the in-domain model and rejected text which had very high perplexity [13]. The goal of the pre-filtering step is to remove artifacts such as advertisements, and other spurious text. Most of these artifacts show up very clearly as a very high perplexity cluster compared to the rest of the data. Thus, by using a perplexity histogram we could easily choose and use a perplexity threshold for pre-filtering. Data were mined separately for the two ASR tasks presented in this paper. In both cases, the initial size of the data downloaded from the web was around 750 M words. After filtering and normalization the downloaded data amounted to about 500 M words.

The generalized algorithm for data selection using n-gram models (see Appendix) is significantly slower than the unigram implementation (Section V) because of the need to update lower order back-off weights. Simulation experiments [9] and experiments on Web-data indicate that bigram and unigram language models seem to perform well for data selection using the R.E. minimization algorithm. No performance gains were observed when using trigram models for selection. For this reason the experimental results presented in this paper are restricted to the use of bigram models for data selection. Note however that the order of the LM used for data selection does not put any restrictions on the order of the language models used for generating query terms or the adapted language model we build from the selected data.

V. Experiments

To provide a more general picture of the performance of our data selection algorithm we provide experimental results on two systems which differ significantly in their system design and the nature of the ASR task that they address.

The first set of experiments were conducted on the English ASR of the Transonics [10] English–Persian speech-to-speech translation system for doctor–patient interactions developed at USC. The second set of experiments were conducted using IBM's speech recognition system for English, submitted to the 2006 evaluation within the TC-STAR project. TC-STAR, which stands for Technology and Corpora for Speech to Speech Translation, financed by the European Commission within the Sixth Framework Program is a long-term effort to advance research in speech-to-speech translation technologies.[3] The 2006 Evaluation was open to external participants as well as the TC-STAR partner sites [20].

We begin by presenting results on the Transonics task. This task was also used to provide comparisons against the large class of rank-and-select schemes described in Section II. We will then provide results on the TC-STAR task. As stated in Section IV bigram models generated from in-domain data were used for data selection. All language models used for decoding and perplexity measurements are trigram models estimated using Kneser–Ney smoothing.

A. Medium-Vocabulary ASR Experiments on Transonics

The English ASR component of the Transonics speech to speech translation system is a medium vocabulary speech recognizer built using the SONIC [21] engine. We had 50 K in-domain sentences (200 K words) for this task to train the language model. A generic conversational-speech language model was built from the WSJ [22], Fisher [23], and SWB [24] corpora interpolated with a conversation speech LM from CMU for broadcast news [25]. All language models built from the selected data and the in-domain data were interpolated with this generic conversational language model. The linear interpolation weight was determined using a heldout set. The test set for word error rate evaluation consisted of 520 utterances. A separate test set used

[2]http://www.id.cbs.dk/dh/corpus/tools/MXTERMINATOR.html.

[3]Project No. FP6-506738.

TABLE I
TRANSONICS: PERPLEXITY, WORD ERROR RATE, AND PERCENTAGE DATA SELECTED FOR DIFFERENT NUMBER OF INITIAL SENTENCES FOR A CORPUS SIZE OF 150 M

		Number of in-domain sentences		
		10K	20K	40K
Perplexity	NoWeb	60.0	49.6	39.7
	AllWeb	57.1	48.1	38.2
	PPL	56.1	48.1	38.2
	BLEU	56.3	48.2	38.3
	LPU	56.3	48.2	38.3
	Proposed	**53.7**	**46.6**	**38.0**
Word error rate (in %)	NoWeb	19.8	18.9	17.9
	AllWeb	19.5	19.1	17.9
	PPL	19.2	18.8	17.9
	BLEU	19.3	18.8	17.9
	LPU	19.2	18.8	17.8
	Proposed	**18.1**	**17.9**	**17.1**
Data selected (in %)	NoWeb	0	0	0
	AllWeb	100	100	100
	PPL	93	92	91
	BLEU	91	90	89
	LPU	90	88	87
	Proposed	**11**	**10**	**11**

for perplexity evaluations consisted of 5000 sentences (35 K words) and the heldout set had 2000 sentences (12 K words).

We report results with increasing amounts of in-domain training material, ranging from 10 K sentence to 40 K sentences. For every choice of in-domain training data size, we carry out data selection using the baseline methods and the proposed R.E. based method. The language models used for data selection with the perplexity rank-and-select baseline (Section II) and R.E. based data selection (Section III) are also built separately for every set of experiments conducted with a different in-domain data size.

We first compare our proposed algorithm against the baseline rank-and-select data selection schemes enumerated in Section II. LPU and BLEU based rank-and-select schemes are computationally intensive. LPU requires iterative retraining of a binary SVM classifier which has high computational complexity. The computational complexity of the BLEU-based ranking scheme is square in the order of the number of sentences. Our results which include comparisons against these two systems are thus limited to a smaller 150 M word Web-collection. The thresholds for data selection using the ranking-based baselines were fixed using the heldout set perplexity.

For the 150 M word Web-collection, Table I shows the fraction of sentences selected and the resulting perplexity and WERs for the various data selection schemes with different amounts of in-domain data used to seed the data selection. In Table I, *NoWeb* refers to the language model built solely from in-domain data and *AllWeb* refers to the case where the entire 150 M web-collection was used. As the comparison shows, the proposed algorithm outperforms the rank-and-select schemes with just 10% of data selected from the web collection. The best reduction in perplexity with the proposed scheme is 5% relative corresponding to a reduction in WER of 3% (relative).

One of the goals of the Transonics task was to find an optimal vocabulary size as the initially available data was quite small [26]. Hence, the vocabularies of the language models used to compute perplexity presented in Table I are different. However, the OOV rate on the heldout data was less than 1% for all the vocabularies. The vocabularies for the language models in Table I ranged from 70 K to 110 K words.

TABLE II
PERPLEXITY, WORD ERROR RATE, AND PERCENTAGE DATA SELECTED FOR DIFFERENT NUMBER OF INITIAL SENTENCES FOR A CORPUS SIZE OF 850 M

		Number of in-domain sentences		
		10K	20K	40K
Perplexity	NoWeb	60.0	49.6	39.7
	AllWeb	56.9	47.7	38.2
	PPL	55.8	47.4	38.2
	Proposed	**52.1**	**45.2**	**36.8**
Word error rate (in %)	NoWeb	19.8	18.9	17.9
	AllWeb	19.3	19.1	17.9
	PPL	19.1	18.7	17.9
	Proposed	**17.8**	**17.6**	**17.0**
Data selected (in %)	NoWeb	0	0	0
	AllWeb	100	100	100
	PPL	88.5	87.8	87.3
	Proposed	**9.3**	**10**	**8.7**

To get a more complete picture of the relationship between performance and amount of data selected, we also conducted experiments using simulations [9], where we restricted the number of sentences selected by the perplexity ranking baseline to be the same as the number of sentences selected by the proposed method. For these simulations, we generated samples from a reference language model using a random walk procedure. We then compared the performance of data selection using perplexity-based ranking and the R.E. criterion with two metrics. The first metric computes perplexity on a heldout set and the second computes the the relative entropy with respect to the reference model [27]. In both these metrics the R.E.-based criterion outscored perplexity-based selection. In fact, for many cases selecting a random subset of data was found to give better performance using both metrics when compared to the baseline perplexity-based ranking method [9].

Table I presented results on data selection from a corpus of 150 M words. In order to study the performance of this algorithm on larger corpora, we used a larger data set of 850 M words which consisted of the medical domain collection of 320 M words collected from the web and a 525 M word collection published by the University of Washington for the Fisher corpus [28], [29].

We provide comparisons with only the perplexity based rank-and-select scheme, as the LPU and BLEU-based schemes do not scale well to large text collections. More importantly, our results on the 150 M word corpus (see Table I) suggest that the performance of the ASR system is approximately the same when using data selected from any one of the LPU-, BLEU-, or PPL-based data selection schemes.

The results on the 850 M word set, measured in terms of PPL and WER (Table II) follow the same trend as in the 150 M data set. The importance of proper data selection is highlighted by the fact that there was little to no improvement in the unfiltered case (*AllWeb*) by adding the extra data as is, whereas consistent improvements can be seen when the proposed iterative selection algorithm was used. Perplexity reduction in relative terms was 7%, 5%, and 4% for the 10 K, 20 K, and 40 K in-domain

TABLE III
TRANSONICS: NUMBER OF ESTIMATED n-GRAMS WITH WEB-ADAPTED MODELS FOR DIFFERENT NUMBER OF INITIAL SENTENCES FOR THE CASE WITH 40 K IN-DOMAIN SENTENCES. CORPUS SIZE = 850 M

	unigram	bigram	trigram
AllWeb	105K	25.3M	36.2M
PPL	99K	22.1M	32.4M
Proposed	**70K**	**3.2M**	**8.2M**

TABLE IV
TRANSONICS: PERPLEXITY FOR DIFFERENT NUMBER OF INITIAL SENTENCES FOR A CORPUS SIZE OF 850 M AND FIXED VOCABULARY OF 73 K WORDS

	10K	20K	40K
PPL	56.1	48.4	39
Prop	53	45.9	37.8

set, respectively. Corresponding WER improvements in relative terms were 6%, 4%, and 4%, respectively. Table II also shows that the amount of data selected by the R.E.-based data selection scheme was a factor of 9 smaller than the entire collection of 850 M words and still provided improved ASR performance. Reduction in the amount of selected data also translates to sparser language models. As can be seen from Table III, the size of the adapted language model built using the proposed algorithm, measured in terms of number of n-grams is significantly smaller then the number of n-grams in the language models built using the baseline methods.

Table IV illustrates the improvement in perplexity that can be achieved with the proposed scheme when the vocabulary is fixed. The fixed vocabulary results correlate well with the results in presented in Table II, where the vocabulary was varied for optimal performance, suggesting that the perplexity gains are from better selection of training data and not because of changes in vocabulary.

It is interesting to note that for our Transonics experiments, the perplexity improvements correlate surprisingly well with WER improvements. This is in contrast to previous studies in speech recognition [30] where WER improvements did not correlate well with perplexity.

Our results on the medium-vocabulary Transonics task, indicate that with the proposed scheme, we can identify significantly smaller sets of sentences such that the models built from the selected data have a substantially sparser representation and yet perform better (in terms of both perplexity and WER) than models built from the entire corpus. Next, we present results on a large vocabulary task.

B. Large Vocabulary Experiments on TC-STAR

To contrast with the medium vocabulary single decoder Transonics system, we conducted experiments on the IBM LVCSR system used for transcription of European Parliamentary Plenary Speeches (EPPS) [11] as part of the TC-STAR project. We present results on two test sets, namely, the development (Dev06) and evaluation (Eval06) test sets. The Dev06 test set consists of 3 h of data from 42 speakers (mostly non-native speakers). The Eval06 test set comprises of 3 h of data from 41 speakers. The Dev06 and Eval06 test sets cover parliamentary sessions between June and September 2005 and contain approximately 30 K words each.

The baseline system's interpolated language model was built from two in-domain EPPS data sources, namely, the transcripts used for training acoustic models (2 M words) and the Final Text Editions (FTE) (33 M words) and two out-of-domain data sources, the University of Washington's Fisher-style web data corpus (525 M words) and data from the Broadcast News domain (204 M words). The baseline performance of the best ASR system on the Dev06 test set was 11% and, 8.9% on the Eval06 test set. We provide performance comparisons against this baseline by replacing the two out-of-domain data sources in the baseline system with increasing fractions of text selected by the R.E. based data selection method. We used perplexity minimization on the Dev06 set to optimize the amount of data selected. As can be seen from Table V, incorporating the 525 M words mined by our crawling scheme (Section IV) boosted the system performance to 8.4% (6% relative over the baseline). The effectiveness of the data selection scheme is demonstrated by the fact that similar performance gains over the baseline are obtained (8.5% and 8.4% WER) when using 1/7th of the data, i.e., 70 M words or all the data i.e., 525 M words. When the data selected is increased to 1/3rd of the total size i.e., 170 M words, the combined WER reduction on the Dev06 and Eval06 set is same as the WER reduction seen with the entire set. Selecting 1/3rd data was found to give the best performance in terms of Dev06 perplexity. We did not observe any improvement in perplexity of Dev06 set by adding more data. A further reduction in WER was achieved when a third out-of-domain source, the 204 M word Broadcast News corpus was included.

- 1/7th of the selected data yielded a reduction of WER from 11% to 10.6% and 8.9% to 8.3% on the Dev06 and Eval06 test sets, respectively.
- 1/3rd of the selected data yielded a reduction of WER from 11% to 10.3% and 8.9% to 8.3% on the Dev06 and Eval06, respectively.

The ASR system used for transcribing English EPPS speeches in the TC-STAR 2006 evaluation used a system combination approach, the detailed architecture is described in [11]. The best performance was obtained with a system combination using ROVER [33], i.e., 10.4% and 8.3% on the Dev06 and Eval06 test sets. The equivalent system combination result when using the R.E. based data selection scheme that selected 1/3rd of the data yielded 9.8% and 7.9% WER on the Dev06 and Eval06 test sets, respectively, a significant improvement in performance at these low levels of WER. To further understand the significance of the data selected using the proposed scheme, we selected 1/3rd of the data from the same corpus randomly and studied the performance of the ASR system. This yielded a WER of 10.8% on the Dev06 and 8.6% on the Eval06 test sets thereby indicating that the proposed scheme does indeed select data that helps in improving ASR performance in terms of WER.

C. Computation Time and n-Gram Order

The computation time of the proposed R.E. based data selection algorithm depends on the order of the n-gram language model used for data selection (see Appendix). In our experiments, we observed an exponential trend in computation time with increasing n-gram order. Table VI shows the computation time required with higher order language models normalized by

TABLE V
TC-STAR: PERFORMANCE COMPARISON OF THE LANGUAGE MODELS BUILT WITH DIFFERENT FRACTIONS OF DATA BEING SELECTED FOR THE DEV06 AND EVAL06 TEST SETS. THE BASELINE HAD 525 M WORDS OF UNIVERSITY OF WASHINGTON'S FISHER-STYLE WEB DATA [31] AND 204 M WORDS OF BROADCAST NEWS [32] (BN) AS OUT-OF-DOMAIN DATA. THE WER ON DEV06 FOR THE BASELINE WAS 11% AND 8.9% ON EVAL06

Fraction of data selected (words)	Baseline	All (525M)	1/11 (45M)	1/7 (71M)	1/3 (170M)
Perplexity (Dev)	115	94.5	94.5	91.3	88.7
WER (Dev)%	11	10.7	10.9	10.8	10.6
WER (Eval)%	8.9	8.4	8.6	8.5	8.5

TABLE VI
TC-STAR: TIME REQUIRED FOR DATA SELECTION WITH INCREASING ORDER OF THE DATA SELECTION LANGUAGE MODEL NORMALIZED BY THE TIME REQUIRED WITH UNIGRAM LANGUAGE MODEL

	unigram	bigram	trigram	4gram	5gram
TC-STAR	1.0	5.2	22.3	117.0	560.2
Transonics	1.0	3.6	13.0	44.1	180.1

the computation time for a unigram model. A detailed theoretical and experimental analysis of the interplay between the language model order, number of parameters and the computation time has not been carried out at this stage. We intend to undertake this analysis in our future work on data selection.

VI. DISCUSSION AND ANALYSIS OF RESULTS

It is interesting to compare the data selection results between the Transonics and TC-STAR experiments. For Transonics, we used a web corpus of 320 M words (excluding Fisher data). The data selection algorithm was able to achieve better performance than the out-of-domain LM built from the entire 320 M word corpus, while selecting just 1/10th of the data. In contrast to Transonics, in our experiments with the IBM TC-STAR system we selected 1/3rd of the data to get the best performance. We believe that this can be attributed to the fact that we could collect more relevant data for the TC-STAR task which covered European parliamentary speeches as compared to the medical dialogue domain Transonics task.

More insights into these results can be gained by comparisons with the performance of the ROVER-based TC-STAR system. First, the 525 M word collection generated using the scheme presented here gave an improvement of 0.5% compared to the baseline which used two out-of-domain sources of over 700 M words. The baseline WER can be achieved with just 70 M words selected from an out-of-domain source (instead of 700 M words). Second, careful data selection can yield the same gains as those obtained from a system combination approach.

VII. CONCLUSION

A. Summary of Contributions

In this paper, we presented a novel scheme for selecting *relevant* subsets of sentences from large collections of text acquired from the web. Our results indicate that with this scheme, we can identify significantly smaller sets of sentences such that the models built from the selected data have a substantially sparser representation and yet perform better (in terms of both perplexity and WER) than models built from the entire corpus. On our medical domain task which had sparse in-domain data (200 K words), we were able to achieve around 4% relative improvement in WER with a factor of 7 reduction in language model parameters while selecting a set of sentences 1/10th the size of the original corpus. For the TC-STAR task where the in-domain resources were much larger (50 M words), we achieved 6% relative WER improvement by using just 1/3rd of the data. Although most of our results in this paper were on data acquired from the web, the proposed method can easily be used for adaptation of domain specific models from other large generic corpora.

B. Scope of This Work

The research effort presented in this paper is directed towards selecting relevant domain specific data from large collections of generic text. We make no assumptions on how the data were collected or what specific web crawling and querying techniques are used. The methods we have developed can be seen as supplementing the research efforts by the machine translation community on identifying web resources [3], [34] or using web counts [2] for language modeling. We also believe that this work can augment topic based LM adaptation techniques. Topic based LM adaptation schemes typically use LSA [35] or variants [36] to automatically split the available training text across multiple topics. This allows for better modeling of each individual topic in the in-domain collection. The tradeoff is that since the available text is split across topics, each individual model is trained on less data. We believe that this problem can be addressed by selecting data for each topic from a large generic corpora using the proposed data selection algorithm.

C. Directions for Future Work

The effect of varying data granularity has not been studied in this work. We have used sentence level selection, but the selection process can also be naturally extended to groups of sentences, fixed number of words, paragraphs or even entire documents. Selection of data in smaller chunks has the potential to select data better suited to the task but may result in over-fitting to the existing in-domain distribution. In such a case the adaptation model will provide little extra information to the existing model. We plan to study the effect of this tradeoff between data novelty and match to in-domain model on the LM performance, for different levels of selection granularity. We are also looking into extending the algorithm to work directly on collections of n-gram counts. One motivation for research in this direction is that Google has released aggregate unigram to 5-gram counts for their web snapshot [37].

The proposed method can be combined with rank-and-select schemes described in Section II. We are exploring the use of ranking to reorder the data such that the sequential selection

process gives better results with fewer number of randomized searches.

The current framework relies on multiple traversals of data in random sequences to identify the relevant subset. An online single-pass version of the algorithm would be of interest in cases where the text data is available as a continuous stream (one such source is RSS feeds from blogs and news sites). If updates from the stream sources are frequent, iterating through the entire text collection is not feasible. One of the ideas we are investigating to make the selection process single-pass is to use multiple instances of the algorithm with different initial in-domain models generated by bagging. Voting across these multiple instances can be then used to select data. We are also investigating how to select sentences with a probability proportional to the relative entropy gain instead of the threshold based approach currently being used.

APPENDIX
GENERALIZATION TO BACK-OFF n-GRAMS

For a given context $h = w_1, \ldots, w_{n-1}$, let $S(h)$ be the set of words for which the probability estimate $p(w|h)$ is explicitly defined in the back-off n-gram model. Consider the n-gram probabilities $p(w|h)$ which lie in the complement set $S^c(h)$. The probability of these n-grams can be computed in terms of the probability of the back-off n-gram with history $h' = w_2, \ldots, w_{n-1}$ as

$$p(w|h) = b(h) * p(w|h')$$

where $b(h)$ is called the back-off weight. Given that $\sum_{w \in V} p(w|h) = 1$ it is easy to see that

$$b(h) = \frac{1 - \sum_{w \in S} p(w|h)}{1 - \sum_{w \in S} p(w|h')}.$$

The primary problem in extending the data selection algorithm presented in Section III to back-off n-grams is the increase in computational complexity of calculating the relative entropy change resulting from changes in the back-off parameters. To keep the complexity tractable we developed a data selection scheme which enforces the back-off structure of the in-domain model on the n-gram model built from the selected data. Note that the assumption that the two models share the same back-off structure, does not limit the selection of data to n-gram histories seen in the in-domain data. By restricting the back-off structure for the model built from selected data, we fix whether we will update an n-gram estimate or modify the corresponding back-off weights. Other methods which can reduce the complexity include treating the model built from selected data as a non back-off ML model. We have not experimented with these alternate strategies.

We first describe a scheme for the fast computation of R.E. between two back-off n-gram language models which share the same back-off structure. We use the generalized derivation from [27] adapted to the case where the two language models have the same back-off structure. To keep the presentation of the algorithm simple, we will use the entropy model described in [9]. This can be changed to the skew divergence model described for the unigram case described in Section III by adjusting the counts to include the in-domain model probability.

A. Fast Computation of Relative Entropy

We define the following symbols for the purpose of describing the R.E. computation.

w	The current word.
h	The history $w_1, \ldots, w_{n-1}$.
h'	The back-off history $w_2, \ldots, w_{n-1}$.
$b^p(h)$	The back-off weight for the p distribution for history h.
$b^q(h)$	The back-off weight for the q distribution for history h.
V	The vocabulary of the language model.

The information theoretic measure of relative entropy rate [38] can be used to compare discrete Markovian distributions such as n-gram language models. Given two n-gram language models $p(w|h)$ and $q(w|h)$, the relative entropy rate for n-gram of size n is defined as

$$R(n) = \sum_{h \in H} p(h) \sum_{w \in V} p(w|h) \ln \frac{p(w|h)}{q(w|h)} \tag{5}$$

where H is the set of all possible histories.

In the rest of this discussion, we will refer to relative entropy rate as just relative entropy. Let us denote the conditional relative entropy [38] between the two n-gram distributions p and q for the history h with $D(h)$. We have

$$D(h) = \sum_{w \in V} p(w|h) \ln \frac{p(w|h)}{q(w|h)}. \tag{6}$$

We now divide the set of all possible histories (H) for n-gram size n into H_e for all h which exist as $n-1$ gram in the p or the q distribution. The complement set (H_e^c) will contain histories with a back-off weight of 1. H_e^c corresponds to histories not seen in either language model. The R.E. for n-gram of size n, $R(n)$ can be expressed as

$$\begin{aligned} R(n) &= \sum_{h \in H} p(h) D(h) \\ &= \sum_{h \in H_e} p(h) D(h) + \sum_{h \in H_e^c} p(h) D(h) \\ &= \sum_{h \in H} p(h) D(h') + \sum_{h \in H_e} p(h) D(h) \\ &\quad - \sum_{h \subset nH_e} p(h) D(h'). \end{aligned}$$

Since $h = w_1 \cdot h'$ and $D(h) = D(h')$ for $h \in H_e^c$, we can marginalize with respect to w_1

$$\begin{aligned} R(n) &= \sum_{w}\sum_{h'} p(h)D(h') + \sum_{h\in H_e} p(h)D(h) \\ &\quad - \sum_{h\in H_e} p(h)D(h') \\ &= \sum_{h'} D(h') \sum_{w} p(h) + \sum_{h\in H_e} p(h)\left(D(h) - D(h')\right) \\ &= \sum_{h'} D(h')p(h') + \sum_{h\in H_e} p(h)\left(D(h) - D(h')\right) \\ &= R(n-1) + \sum_{h\in H_e} p(h)\left(D(h) - D(h')\right). \end{aligned}$$

In [27], $D(h)$ is split into four terms depending on whether $w|h$ is explicitly defined in the p or the q distribution. When the two LMs have the same back-off structure, we need to consider only two terms in the expansion of $D(h)$. We call these terms $T_1(h)$ and $T_4(h)$, to use the same notation as the derivation in [27]. $T_1(h)$ corresponds to terms $p(w|h)$ and $q(w|h)$ which exist as explicit n-grams ($w \in S(h)$) and $T_4(h)$ corresponds to $p(w|h)$ and $q(w|h)$ which back-off ($w \in S^c(h)$). We have

$$D(h) = T_1(h) + T_4(h) \tag{7}$$

$$T_1(h) = \sum_{w\in S(h)} p(w|h) \ln \frac{p(w|h)}{q(w|h)}$$

$$T_4(h) = \sum_{w\in S^c(h)} b^p(h)p(w|h') \ln \frac{b^p(h)p(w|h')}{b^q(h)q(w|h')}. \tag{8}$$

Thus, we are able to express $D(h)$, in terms of the n-grams explicitly defined in the LM. [27] provides more details on the derivation for $T_1(h)$ and $T_4(h)$.

We have used the tree-based representation of back-off n-gram models to derive the efficient computation scheme described above. An alternative approach for deriving the same relative entropy expressions presented above would be to consider n-gram back-off language models as a special case of probabilistic finite state grammars (PFSGs) [39], [40].

B. Incremental Updates on an n-Gram Model

We now consider the calculation of incremental changes in R.E. between an in-domain n-gram back-off model p and an ML model q built from selected data. We are interested in finding out an efficient way to compute the change in R.E. when a sentence is added to the selected data set, thus changing the model q. Extending the notation from Section III let us define $C(hw)$ as the count of the word w seen with context h and $C(h)$ as the count for context h in the selected set (ML estimate $q(w|h) = C(hw)/C(h)$). We use $c(hw)$ and $c(h)$ to denote the counts in the current sentence. We assume that the model q has the same back-off structure as the model p. Thus, we can divide $D(h)$ into just $T_1(h)$ and $T_4(h)$ depending on whether w is explicitly defined with context h in the model. For $T_1(h)$, we have

$$\begin{aligned} T_1(h) &= \sum_{w\in S(h)} p(w|h)\ln p(w|h) - \sum_{w\in S(h)} p(w|h)\ln q(w|h) \\ &= \sum_{w\in S(h)} p(w|h)\ln p(w|h) - \sum_{w\in S(h)} p(w|h)\ln \frac{C(hw)}{C(h)}. \end{aligned}$$

After addition of a sentence the updated value of $T_1(h)$ is given by

$$\begin{aligned} T_1^+(h) &= \sum_{w\in S(h)} p(w|h)\ln p(w|h) \\ &\quad - \sum_{w\in S(h)} p(w|h)\ln \frac{C(hw)+c(hw)}{C(h)+c(h)} \\ &= T_1(h) + \ln\frac{C(h)+c(h)}{C(h)} \sum_{w\in S(h)} p(w|h) \\ &\quad - \sum_{w\in S(h),c(hw)\neq 0} p(w|h)\ln\frac{C(hw)+c(hw)}{C(hw)}. \end{aligned}$$

Thus, the change in $T_1(h)$

$$\begin{aligned} \delta T_1(h) &= \ln\frac{C(h)+c(h)}{C(h)} \sum_{w\in S(h)} p(w|h) \\ &\quad - \sum_{w\in S(h),c(hw)\neq 0} p(w|h)\ln\frac{C(hw)+c(hw)}{C(hw)}. \end{aligned}$$

The term $\sum_{w\in S(h)} p(w|h)$ can be precomputed since it is not a function of the word counts in the selected set.

We now consider $T_4(h)$ which we further split into two parts

$$\begin{aligned} T_4(h) &= \sum_{w\in S^c(h)} p(w|h)\ln p(w|h) \\ &\quad - \sum_{w\in S^c(h)} b^p(h)p(w|h')\ln b^q(h)q(w|h') \\ &= \sum_{w\in S^c(h)} p(w|h)\ln p(w|h) \\ &\quad - b^p(h) \underbrace{\sum_{w\in S^c(h)} p(w|h')\ln b^q(h)}_{T_{4A}(h)} \\ &\quad - b^p(h) \underbrace{\sum_{w\in S^c(h)} p(w|h')\ln q(w|h')}_{T_{4B}(h)}. \end{aligned}$$

Computing the change in $T_{4A}(h)(\delta T_{4A}(h))$ requires computation of change in $b^q(h)$

$$T_{4A}(h) = \left(1 - \sum_{w\in S(h)} p(w|h')\right)\ln b^q(h)$$

$$\delta T_{4A}(h) = \left(1 - \sum_{w\in S(h)} p(w|h')\right)\delta \ln b^q(h).$$

As for the $T_1(h)$ case, $\sum_{w\in S(h)} p(w|h')$ can be precomputed.

The expression for $b^q(h)$ is given by

$$b^q(h) = \frac{1-\sum_{w\in S(h)} \frac{C(hw)}{C(h)}}{1-\sum_{w\in S(h)} \frac{C(h'w)}{C(h')}} = \frac{C(h')}{C(h)} \frac{C(h)-\sum_{w\in S(h)} C(hw)}{C(h')-\sum_{w\in S(h)} C(h'w)}.$$

The expression for $b^q(h)$ after the addition of a new sentence is given by

$$b^{q+}(h) = \frac{1-\sum_{w\in S(h)} \frac{C(hw)+c(hw)}{C(h)+c(h)}}{1-\sum_{w\in S(h)} \frac{C(h'w)+c(h'w)}{C(h')+c(h')}} = \frac{C(h)+c(h)-\sum_{w\in S(h)} (C(hw)+c(hw))}{C(h')+c(h')-\sum_{w\in S(h)} (C(h'w)+c(h'w))} \times \frac{C(h')+c(h')}{C(h)+c(h)}.$$

Computation of change in $\ln b^q(h) (\delta \ln b^q(h))$ is not required for the case where $c(h') = 0$. With zero counts for h' we also have zero counts for longer contexts which extend h'. Thus, with $c(h') = 0$ we have $c(hw) = c(h) = c(h'w) = 0$. Hence, $b^{q+}(h) = b^q(h)$, which implies $\delta \ln b^q(h) = 0$.

For the case where $c(h) = 0$ and $c(h') \neq 0$, the computation of $\delta \ln b^q(h)$ is simplified. We have

$$\delta \ln b^q(h) = \ln \frac{C(h')-\sum_{w\in S(h)} C(h'w)}{C(h')+c(h')-\sum_{w\in S(h)} (C(h'w)+c(h'w))} \times \frac{C(h')+c(h')}{C(h')}.$$

$T_{4B}(h)$ can be expressed as

$$\begin{aligned} T_{4B}(h) &= \sum_{w\in S^c(h)} p(w|h') \ln q(w|h') \\ &= -D(h') - \sum_{w\in S(h)} p(w|h') \ln q(w|h') + \sum_{w\in V} p(w|h') \ln p(w|h') \\ &= -D(h') - \sum_{w\in S(h)} p(w|h') \ln \frac{C(h'w)}{C(h')} + \sum_{w\in V} p(w|h') \ln p(w|h'). \end{aligned}$$

For $T_{4B}(h)$, the updated value after the addition of a new sentence can be expressed as

$$T^+_{4B}(h) = -D^+(h') - \sum_{w\in S(h)} p(w|h') \ln \frac{C(h'w)+c(h'w)}{C(h')+c(h')}$$

and thus the change in $T_{4B}(h)$, $\delta T_{4B}(h)$ can be expressed as

$$\delta T_{4B}(h) = -\delta D(h') + \ln \frac{C(h')+c(h')}{C(h')} \sum_{w\in S(h)} p(w|h') - \sum_{w\in S(h), c(h'w)\neq 0} p(w|h') \ln \frac{C(h'w)+c(h'w)}{C(h'w)}.$$

The total number of computations grows linearly with the total number of n-grams in the language model which grows exponentially with the order of the model. For initialization, we use a unigram model initialized with a random subset of data to seed data selection (Section III). The data selected with the unigram model is then used to initialize the counts for the q model.

REFERENCES

[1] M. Lapata and F. Keller, "Web-based models for natural language processing," *ACM Trans. Speech Lang. Process.*, vol. 2, pp. 1–31, 2005.

[2] F. Keller, M. Lapata, and O. Ourioupina, "Using the web to overcome data sparseness," in *Proc. EMNLP*, 2002, pp. 230–237.

[3] P. Resnik and N. A. Smith, "The web as a parallel corpus," *Comput. Linguist.*, vol. 29, pp. 349–380, 2003.

[4] D. Graff, "English Gigaword," LDC Catalog ID: LDC2003T05, 2003.

[5] T. Ng, M. Ostendorf, M.-Y. Hwang, M. Siu, I. Bulyko, and X. Lei, "Web-data augmented language model for Mandarin speech recognition," in *Proc. ICASSP*, 2005, pp. 589–592.

[6] R. Sarikaya, A. Gravano, and Y. Gao, "Rapid language model development using external resources for new spoken dialog domains," in *Proc. ICASSP*, 2005, pp. 573–576.

[7] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using EM," *J. Mach. Learn.*, vol. 39, pp. 103–134, 2000.

[8] X. Zhu, "Semi-supervised learning literature survey," Univ. of Wisconsin-Madison, Comput. Sci., Tech. Rep. 1530, Dec. 2005 [Online]. Available: http://www.cs.wisc.edu/jerryzhu/pub/ssl_survey.pdf.

[9] A. Sethy, P. G. Georgiou, and S. Narayanan, "Text data acquisition for domain-specific language models," in *Proc. EMNLP*, 2006, pp. 382–389.

[10] S. Narayanan, S. Ananthakrishnan, R. Belvin, E. Ettaile, S. Ganjavi, P. Georgiou, C. Hein, S. Kadambe, K. Knight, D. Marcu, H. Neely, N. Srinivasamurthy, D. Traum, and D. Wang, "Transonics: A speech to speech system for English-Persian interactions," in *Proc. IEEE ASRU*, 2003, pp. 1–6.

[11] B. Ramabhadran, O. Siohan, L. Mangu, G. Zweig, M. Westphal, H. Schulz, and A. Soneiro, "The IBM 2006 speech transcription system for European parliamentary speeches," in *Proc. ICSLP*, 2006, pp. IV33–IV36.

[12] T. Misu and T. Kawahara, "A bootstrapping approach for developing language model of new spoken dialogue systems by selecting web texts," in *Proc. ICSLP*, 2006, pp. 9–12.

[13] A. Sethy, P. Georgiou, and S. Narayanan, "Building topic specific language models from web-data using competitive models," in *Proc. Eurospeech*, 2005, pp. 1293–1296.

[14] K. Weilhammer, M. N. Stuttlem, and S. Young, "Bootstrapping language models for dialogue systems," in *Proc. ICSLP*, 2006, pp. 1482–1485.

[15] B. Liu, Y. Dai, X. Li, W. S. Lee, and P. Yu, "Building text classifiers using positive and unlabeled examples," in *Proc. ICDM*, 2003, pp. 179–189.

[16] A. Stolcke, "Entropy-based pruning of backoff language models," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, 1998, pp. 270–274.

[17] L. Lee, "Measures of distributional similarity," in *Proc. ACL*, 1999, pp. 25–32.

[18] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," in *Proc. ACL*, 1996, pp. 310–318.

[19] J. Reynar and A. Ratnaparkhi, "A maximum entropy approach to identifying sentence boundaries," in *Proc. 5th Conf. Appl. Natural Lang. Process.*, 1997, pp. 16–19.

[20] "TC-STAR: Technology and Corpora for Speech to Speech Translation." [Online]. Available: http://www.tc-star.org

[21] B. Pellom, "SONIC: The University of Colorado Continuous Speech Recognizer," Univ. of Colorado, Boulder, Tech. Rep. TR-CSLR-2001-01, Jan. 2001.

[22] D. B. Paul and J. M. Baker, "The design for the wall street journal-based csr corpus," in *Proc. Workshop on Speech and Natural Lang.*, 1992, pp. 889–902.

[23] C. Cieri, D. Miller, and K. Walker, "The Fisher Corpus: A resource for the next generations of speech-to-text," in *Proc. LREC*, 2004, pp. 69–71.

[24] D. Graff, K. Walker, and A. Canavan, "Switchboard-2 Phase II," LDC Catalog ID: LDC99S79, 1999.
[25] K. Seymore, S. Chen, S.-J. Doh, E. Gouvea, B. Raj, M. Ravishankar, R. Rosenfeld, M. Siegler, R. Stern, and E. Thayer, "The 1997 CMU Sphinx-3 english broadcast news transcription system," in *Proc. 1998 DARPA Speech Recognition Workshop*, 1998, pp. 55–89.
[26] E. Ettelaie, S. Gandhe, P. Georgiou, K. Knight, D. Marcu, S. Narayanan, D. Traum, and R. Belvin, "Transonics: A practical speech-to-speech translator for English–Farsi medical dialogues," in *Proc. ACL*, 2005, pp. 89–92.
[27] A. Sethy, B. Ramabhadran, and S. Narayanan, "Measuring convergence in language model estimation using relative entropy," in *Proc. ICSLP*, 2004, pp. 1057–1060.
[28] I. Bulyko, M. Ostendorf, and A. Stolcke, "Getting more mileage from web text sources for conversational speech language modeling using class-dependent mixtures," in *Proc. HLT*, 2003, pp. 7–9.
[29] O. Cetin and A. Stolcke, "Language Modeling in the ICSI-SRI Spring 2005 Meeting Speech Recognition Evaluation," ICSI, Tech. Rep. TR-05-006, July 2005.
[30] R. Iyer, M. Ostendorf, and M. Meteer, "Analyzing and predicting language model improvements," in *Proc. ASRU*, 1997, pp. 254–261.
[31] "Fisher-Related Conversational Style Web Collection," [Online]. Available: http://ssli.ee.washington.edu/projects/ears/WebData/web_data_collection.html.
[32] S. Chen, B. Kingsbury, L. Mangu, D. Povey, G. Saon, H. Soltau, and G. Zweig, "Advances in speech transcription at IBM under the DARPA EARS program," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 5, pp. 1596–1608, Sep. 2006.
[33] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus among words: Lattice-based word error minimization," in *Proc. Eurospeech*, 1999, pp. 495–498.
[34] F. Huang, Y. Zhang, and S. Vogel, "Mining key phrase translations from web corpora," in *Proc. EMNLP*, 2005, pp. 483–490.
[35] J. Bellegarda, "Large vocabulary speech recognition with multispan statistical language models," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 1, pp. 76–84, Jan. 2000.
[36] B. J. Hsu and J. Glass, "Style and topic language model adaptation using HMM-LDA," in *Proc. EMNLP*, 2006, pp. 373–381.
[37] T. Brants and A. Franz, "Web 1T 5-Gram Version 1," LDC Catalog ID: LDC2006T13, 2006.
[38] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley-Interscience, Aug. 1991.
[39] M. Mohri, "Finite-state transducers in language and speech processing," *Comput. Linguist.*, vol. 23, pp. 269–311, 1997.
[40] R. C. Carrasco, "Accurate computation of the relative entropy between stochastic regular grammars," *RAIRO (Theoretical Informatics and Applications)*, vol. 31, pp. 437–444, 1997.

Abhinav Sethy received the Ph.D. degree from the University of Southern California, Los Angeles, in 2007, where the main focus of his research was in the field of acoustic and language modeling for large vocabulary speech recognition.

He joined the IBM T. J. Watson Research Center, Yorktown Heights, NY, after completing the Ph.D. degree, where his research focuses on the sharing of resources between languages and domains.

Panayiotis G. Georgiou (M'02) received the B.A. and M.Eng. degrees (with Honors) from Cambridge University (Pembroke College), U.K., in 1996 and the M.Sc. and Ph.D. degrees from the University of Southern California (USC), Los Angeles, in 1998 and 2002, respectively.

From 1992–1996, he was awarded a Commonwealth scholarship from Cambridge-Commonwealth Trust. Since 2003, he has been a member of the Speech Analysis and Interpretation Lab, USC, first as a Research Associate and currently as a Research Assistant Professor. His interests span the fields of human social and cognitive signal processing. He has worked on and published over 30 papers in the fields of statistical signal processing, alpha stable distributions, speech and multimodal signal processing and interfaces, speech translation, language modeling, immersive sound processing, sound source localization, and speaker identification. His current focus is on multimodal cognitive environments and speech-to-speech translation.

Bhuvana Ramabhadran is the Manager of the Speech Transcription and Synthesis Research Group at the IBM T. J. Watson Center, Yorktown Heights, NY. Upon joining IBM in 1995, she made significant contributions to the ViaVoice line of products focusing on acoustic modeling including acoustics-based baseform determination, factor analysis applied to covariance modeling, and regression models for Gaussian likelihood computation. She has served as the Principal Investigator of two major international projects: the NSF-sponsored MALACH Project, developing algorithms for transcription of elderly, accented speech from Holocaust survivors, and the EU-sponsored TC-STAR Project, developing algorithms for recognition of EU parliamentary speeches. She was the publications chair of the 2000 ICME Conference, Organized the HLT-NAACL 2004 Workshop on Interdisciplinary Approaches to Speech Indexing and Retrieval, and a 2007 Special Session on Speech Transcription and Machine Translation at the 2007 ICASSP in Honolulu, HI. Her research interests include speech recognition algorithms, statistical signal processing, pattern recognition, and biomedical engineering.

Shrikanth Narayanan (S'88–M'95–SM'02) received the Ph.D. degree from the University of California at Los Angeles in 1995.

He was formerly with AT&T Bell Laboratories and AT&T Research, first as a Senior Member, and later as a Principal member, of its Technical Staff, from 1995 to 2000. He is currently the Andrew J. Viterbi Professor of Engineering at the University of Southern California (USC), Los Angeles, and holds appointments jointly as Professor of electrical engineering and Professor in computer science, linguistics and psychology. At USC, he is a member of the Signal and Image Processing Institute and was a Research Area Director of the Integrated Media Systems Center, an NSF Engineering Research Center. He has published over 300 papers and has 15 granted/pending U.S. patents.

Dr. Narayanan has been an Editor for the Computer Speech and Language Journal since 2007 and is an Associate Editor for the *IEEE Signal Processing Magazine* and *IEEE Transactions on Multimedia*. He was also an Associate Editor of the IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING from 2000 to 2004. He served on the Speech Processing technical committee from 2003 to 2007 and Multimedia Signal Processing technical committees from 2004 to 2008 of the IEEE Signal Processing Society and the Speech Communication committee of the Acoustical Society of America from 2003 to present. He is a Fellow of the Acoustical Society of America and a member of Tau-Beta-Pi, Phi Kappa Phi, and Eta-Kappa-Nu. He is a recipient of an NSF CAREER Award, USC Engineering Junior Research Award, USC Electrical Engineering Northrop Grumman Research Award, a Provost fellowship from the USC Center for Interdisciplinary research, a Mellon Award for Excellence in Mentoring, and he is a recipient of a 2005 Best Paper Award from the IEEE Signal Processing Society. Papers that he coauthored with his students have won best student paper awards at ICSLP 2002; the International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2005; International Workshop on Multimedia Signal processing (MMSP) 2006; and MMSP 2007.