Class-dependent Interpolation for Estimating Language Models from Multiple Text Sources

Ivan Bulyko, Mari Ostendorf
{bulyko,mo}@ee.washington.edu
Dept of EE, University of Washington
Seattle WA, 98195-2500

Andreas Stolcke
stolcke@speech.sri.com
SRI International
Menlo Park, CA 94025

UWEE Technical Report
Number UWEETR-2003-0000
March 2003

Department of Electrical Engineering
University of Washington
Box 352500
Seattle, Washington 98195-2500
PHN: (206) 543-2150
FAX: (206) 543-3842
URL: http://www.ee.washington.edu

Class-dependent Interpolation for Estimating Language Models from Multiple Text Sources

Ivan Bulyko, Mari Ostendorf
{bulyko,mo}@ee.washington.edu
Dept of EE, University of Washington
Seattle WA, 98195-2500

Andreas Stolcke
stolcke@speech.sri.com
SRI International
Menlo Park, CA 94025

University of Washington, Dept. of EE, UWEETR-2003-0000

March 2003

Abstract

Sources of training data suitable for language modeling of conversational speech are limited. In this paper, we show how training data can be supplemented with text from the web filtered to match the style and/or topic of the target recognition task, but also that it is possible to get bigger performance gains from the data by using class-dependent interpolation of N-grams.

1 Introduction

Language models constitute one of the key components in modern speech recognition systems. Training an N-gram language model, the most commonly used type of model, requires large quantities of text that is matched to the target recognition task both in terms of style and topic. In tasks involving conversational speech the ideal training material, i.e. transcripts of conversational speech, is costly to produce, which limits the amount of training data currently available.

Methods have been developed for the purpose of language model adaptation, i.e. the adaptation of an existing model to new topics, domains, or tasks for which little or no training material may be available. Since out-of-domain data can contain relevant as well as irrelevant information, various methods are used to identify the most relevant portions of the out-of-domain data prior to combination. Past work on pre-selection has been based on word frequency counts [17], probability (or perplexity) of word or part-of-speech sequences [8], latent semantic analysis [1], and information retrieval techniques [12, 8]. Perplexity-based clustering has also been used for defining topic-specific subsets of in-domain data [6, 4, 13], and test set perplexity has been used to prune documents from a training corpus [10]. The most common method for using the additional text sources is to train separate language models on a small amount of in-domain and large amounts of out-of-domain data and to combine them by interpolation, also referred to as mixtures of language models. The technique was reported by IBM in 1995 [11], and has been used by many sites since then. An alternative approach involves decomposition of the language model into a class n-gram for interpolation [7, 16], allowing content words to be interpolated with different weights than filled pauses, for example, which gives an improvement over standard mixture modeling for conversational speech.

Recently researchers have turned to the World Wide Web as an additional source of training data for language modeling. For "just-in-time" language modeling [2], adaptation data is obtained by submitting words from initial hypotheses of user utterances as queries to a web search engine. Their queries, however, treated words as individual tokens and ignored function words. Such a search strategy typically generates text of a non-conversational style, hence not ideally suited for ASR. In [24], instead of downloading the actual web pages, the authors retrieved N-gram counts provided by the search engine. Such an approach generates valuable statistics but limits the set of N-grams to ones occurring in the baseline model.

In this paper, we present an approach to extracting additional training data from the web by searching for text that is better matched to a conversational speaking style. We also show how we can make better use of this new data by applying class-dependent interpolation.

2 Collecting Text from the Web

The amount of text available on the web is enormous (over 3 billion web pages are indexed via Google alone) and continues to grow. Most of the text on the web is non-conversational, but there is a fair amount of chat-like material that is similar to conversational speech though often omitting disfluencies. This was our primary target when extracting data from the web. Queries submitted to Google were composed of N-grams that occur most frequently in the switchboard training corpus, e.g. "I never thought I would", "I would think so", etc. We were searching for the exact match to one or more of these N-grams within the text of the web pages. Web pages returned by Google for the most part consisted of conversational-style phrases like "we were friends but we don't actually have a relationship" and "well I actually I I really haven't seen her for years."

We used a slightly different search strategy when collecting topic-specific data. First we extended the baseline vocabulary with words from a small in-domain training corpus [18], and then we used N-grams with these new words in our web queries, e.g. "wireless mikes like", "I know that recognizer" for a meeting transcription task [14]. Web pages returned by Google mostly contained technical material related to topics similar to what was discussed in the meetings, e.g. "we were inspired by the weighted count scheme...", "for our experiments we used the Bellman-Ford algorithm...", etc.

The retrieved web pages were filtered before their content could be used for language modeling. First we stripped the HTML tags and ignored any pages with a very high OOV rate. We then piped the text through a maximum entropy sentence boundary detector [15] and performed text normalization using NSW tools [19].

3 Class-dependent Mixture of LMs

Linear interpolation is a standard approach to combining language models, where the probability of a word w_i given history h is computed as a linear combination of the corresponding N-gram probabilities from S different models:

$$p(w_i|h) = \sum_{s \in S} \lambda_s p_s(w_i|h).$$

Depending on how much adaptation data is available it may be beneficial to estimate a larger number of mixture weights λ_s (more than one per data source) in order to handle source mismatch, specifically letting the mixture weight depend on the context h:

$$p(w_i|h) = \sum_{s \in S} \lambda_s(h) p_s(w_i|h).$$

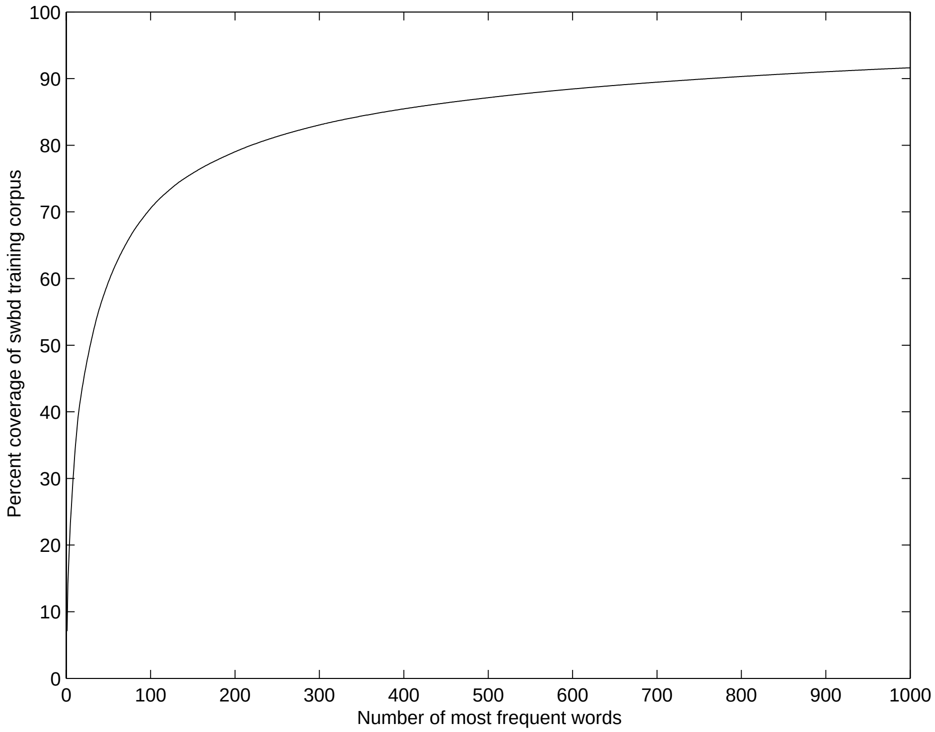

Figure 1: Percent coverage of switchboard training data tokens as a function of the vocabulary size.

One approach is to use a mixture weight corresponding to the source posterior probability $\lambda_s(h) = p(s|h)$ [23]. Here, we instead choose to let the weight vary as a function of the previous word class, i.e. $\lambda_s(h) = \lambda_s(c(w_{i-1}))$. The classes $c(w_{i-1})$ include part-of-speech tags and the 100 most frequent words which form their own individual classes. Such a scheme can generalize across domains by tapping into the syntactic structure (POS tags), already shown to be useful for cross-domain language modeling [7], and at the same time target conversational speech since the top 100 words cover 70% of tokens in Switchboard training corpus. Beyond the top 100 words, additional words give relatively small increments in the corpus coverage, as illustrated in Figure 1.

Combining several N-grams can produce a model with a very large number of parameters, which is costly in decoding. In such cases N-grams are typically pruned. Here we use entropy-based pruning [20] after mixing unpruned models. In experiments comparing standard mixtures to class-dependent interpolation, all models use the same pruning parameters (i.e. entropy gain of 10^{-8}), and we reduce the model aggressively to about 15% of its original size. In the experiments on the effect of pruning, the threshold is varied to obtain several models corresponding to a wide range of sizes.

4 Experiments

4.1 Experiment Paradigm

We evaluated our work on two tasks: 1) Switchboard [5], specifically the HUB5 eval 2001 set having a total of 60K words spoken by 120 speakers, and 2) an ICSI Meeting recorder [14] eval set having a total of 44K words spoken by 25 speakers. Both sets featured spontaneous conversational speech. There were 45K words of held-out data for each task, used for estimating mixture weights and pruning.

Text corpora of conversational telephone speech (CTS) available for training language models consisted of Switchboard, Callhome English, and Switchboard-cellular, a total of 3 million words. In addition to that we used 150 million words of Broadcast News (BN) transcripts, and we collected 191 million words of "conversational" text from the web. For the Meetings task, there were 200K words of meeting transcripts available for training, and we collected 28 million words of "topic-related" text from the web. We also collected 66 million words of text from random web pages in order to assess the importance of content filtering.

The experiments were conducted using the SRI large vocabulary speech recognizer [22] in the N-best rescoring mode. A baseline bigram language model was used to generate N-best lists, which were then rescored with various trigram models. All models used in HUB5 experiments, including the baseline, had identical vocabularies (36546 words). All Meetings results were obtained using a recognizer with this vocabulary augmented with 413 new words from the Meetings speech training data and related text sources [18]. Estimation of language models was accomplished by means of the SRI language modeling toolkit [21], which we extended to allow computation of class-dependent mixtures. All models implemented the modified Knesser-Ney discounting scheme [3].

4.2 Perplexity and WER results

Table 1 shows perplexity numbers and word error rates (WER) that we achieved on the HUB5 test set comparing performance of the class-based mixture against standard (i.e. class-independent) interpolation. The class-based mixture gave better results in all cases except when only CTS sources were used. This may be due to the fact that these sources are similar to each other, whereas the benefits of class-based mixture are more evident in cases where data sources are more diverse. We also obtained lower WER by using the web data instead of BN, which indicates that the web data is better matched to our task (i.e. it is more "conversational").

If training data is completely arbitrary, as shown by two examples of using a 66M-word corpus collected from random web pages, then its benefits to the recognition task are very minimal, if any. In fact, when combining the random data with the CTS and BN sources by means of a standard mixture we observe a degradation in performance compared to using just CTS and BN. There is, however, no degradation if we use class-based mixture to combine the data sources, suggesting that class-based mixtures may be more robust to mismatch in training data than the traditional class-independent interpolation.

Increasing the amount of web training data from 61M to 191M gave relatively small performance gains. We "trimmed" the 191M-word web corpus down to 61M words by choosing documents with lowest perplexity according to the combined CTS model, yielding the "Web2" data source. The model that used Web2 gave the same WER as the one trained with the original 61M web corpus. It could be that the web text obtained with "Google" filtering is fairly homogeneous, so little is gained by further perplexity filtering, which is supported by the fact that the variance in perplexity numbers computed on the "conversational" web data using the combined CTS model is less than half of that computed on the random web data.

Our results on the Meeting test set are shown in Table 2, where the baseline model was trained on CTS and BN sources. As in the HUB5 experiments, the class-based mixture consistently outperformed the standard interpolation. We were able to achieve lower WER by using the web data instead of the meeting transcripts, but the best results are obtained by using all data sources.

It is evident from Tables 1 and 2 that in the majority of experiments class-based mixtures yield lower perplexity of the test set than the corresponding standard mixtures. In several cases, however, WER reductions were accompanied with an increase in perplexity of the test set. When we take all HUB5 experiments listed above into account, the correlation between test set perplexity and WER appears to be very strong (0.96). We also found that 3-gram hit ratio (i.e. percentage of 3-grams in the test data that had explicit probability estimates in the language model) has a negative correlation of -0.84 with WER.

Table 1: HUB5 (eval 2001) perplexity and WER results using standard and class-based mixtures.

LM Data Sources	Std. mix		Class mix	
	PPL	WER	PPL	WER
Baseline CTS	96.0	38.9%	96.7	38.9%
+ 150M BN	87.4	37.9%	87.3	37.8%
+ 66M Web (Random)	91.1	38.6%	91.3	38.3%
+ 61M Web	84.1	37.7%	84.5	37.6%
+ 191M Web	83.0	37.6%	82.4	37.4%
+ 150M BN + 66M Web(Rnd)	87.9	38.1%	87.3	37.8%
+ 150M BN + 61M Web	83.9	37.7%	83.5	37.3%
+ 150M BN + 191M Web	83.4	37.5%	82.3	37.2%
+ 150M BN + 61M Web2	83.4	37.7%	83.5	37.3%

Table 2: Meetings results (perplexity and WER).

LM Data Sources	Std. mix		Class mix	
	PPL	WER	PPL	WER
Baseline	121.7	38.2%	-	-
+ 0.2M Meetings	104.0	37.2%	103.2	36.9%
+ 28M Web (Topic)	108.4	36.9%	106.3	36.7%
+ Meetings + Web (Topic)	98.8	36.2%	94.7	35.9%

4.3 Class Assignments

We tried five different class assignments for the class-based mixture on the HUB5 set; the results are shown in Table 3. Using different data to choose the top 100 words changes the list by only 12 words and does not impact performance. Using automatically derived classes instead of part-of-speech tags does not lead to performance gain, nor does it lead to degradation as long as we allocate individual classes for the top 100 words. This result has important implications for porting to new languages: automatic class mapping can make class-based mixtures feasible for other languages where part-of-speech tags are difficult to derive.

4.4 Pruning Language Models

Very large language models may be too demanding on computational resources. Here we investigate effects of pruning models on WER in the context of the HUB5 task. We conducted a series of experiments using

Table 3: HUB5 WER using class mixtures of CTS, BN and 61M Web data sources with different class mappings.

Class mapping	WER
35 POS + 100 top words from SWBD	37.3%
35 POS + 100 top words from SWBD & SWBD-cell	37.3%
35 automatic classes + 100 top words from SWBD	37.3%
20 automatic classes + 80 top words from SWBD + POS for bottom 25% of words	37.4%
135 automatic classes	37.5%

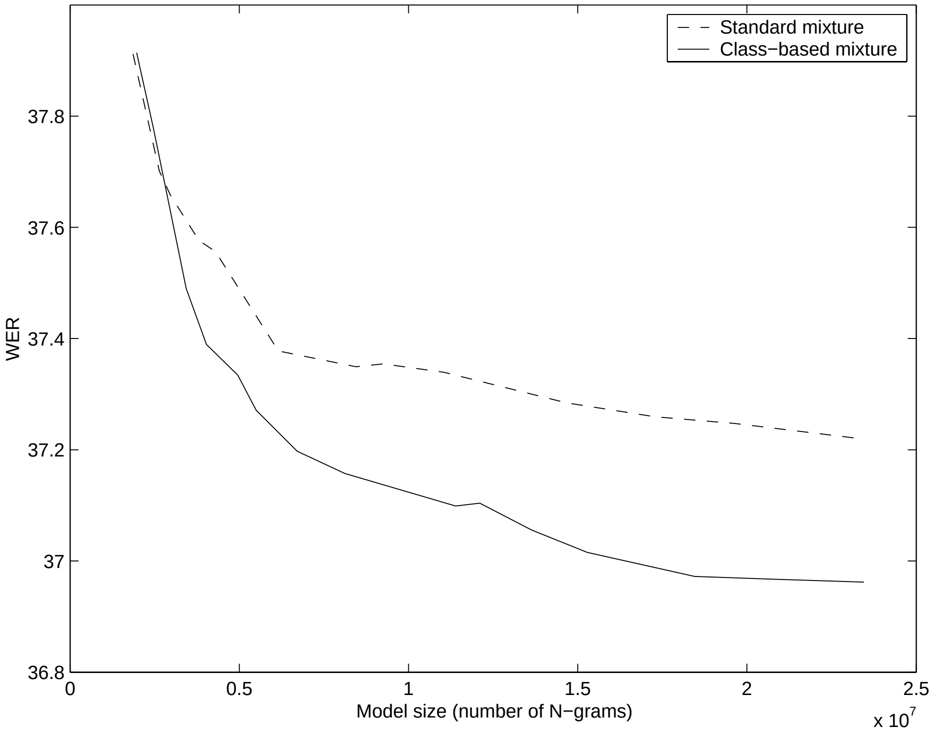

Figure 2: The effect of pruning: WER as a function of model's size.

standard and class mixtures of CTS, BN and 61M Web data sources and pruning the models at various thresholds of entropy gain (see [20] for details on pruning algorithm).

Entropy-based pruning does not necessarily produce exactly the same number of N-grams when a given threshold is applied to different models. As a result, models used in Section 4.2, while being comparable, did not have identical sizes. Here, by varying the pruning threshold we can determine how model size affects performance and compare the standard and class-based mixtures more thoroughly.

Figure 2 shows the non-linear relationship between WER and the model's size (total number of N-grams) for the standard and class-based mixture. It is evident that beyond 5M N-grams (which corresponds to our normal pruning level of 10^{-8} relative increase in entropy and used for results in Section 4.2) gains due to additional N-grams diminish rapidly. However, it may still be cost effective to retain up to 10M N-grams, particularly for the class-based mixture. One can also see that class-based mixture outperformed the standard mixture regardless of the number of parameters used, with the exception of a small number of cases where models were very heavily pruned and the difference in performance between the two approaches is insignificant.

We analyzed three characteristics of the model: 1) increase in model's entropy relative to the unpruned model (i.e. the pruning threshold), 2) trigram hit rate on the test set (i.e. percentage of 3-grams in the test data that had explicit probability estimates in the language model), and 3) perplexity of the test set. Figure 3 illustrates how these three characteristics are affected by the model's size and how well they correlate with WER. The correlation coefficients listed in Table 4 show near linear dependencies between WER and the above three measures with the perplexity having the strongest correlation (0.99). On the log scale, model size is also well correlated with WER. The perplexity result, in particular, is not consistent with earlier analyses of language models trained with out-of-domain data [9], which may be because the web sources are reasonably well matched to conversational speech. Plots in Figure 3 and correlation numbers in Table 4 are based on 13 data points for the standard mixture and 15 data points for the class-based mixture.

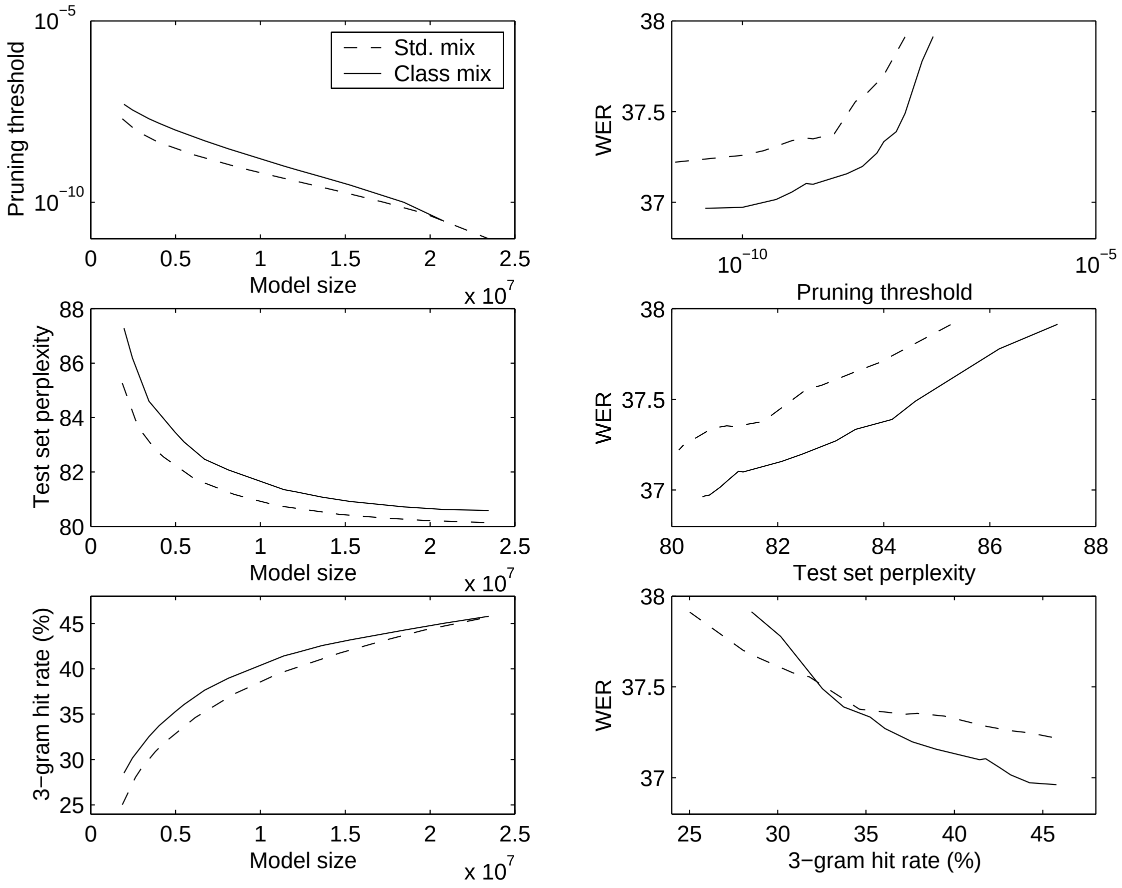

Figure 3: Model's entropy (in log domain), 3-gram hit rate, and test set perplexity affected by model's size and projected onto WER.

Table 4: Correlation between various model characteristics and WER on the HUB5 task using a standard and a class-based mixtures of CTS, BN and 61M Web data sources.

Model characteristics	Correlation with WER	
	Std. mix	Class mix
Model's size (number of N-grams)	-0.84	-0.83
log_{10} of model's size	-0.96	-0.95
Pruning threshold	0.95	0.97
Perplexity of the test set	0.99	0.99
3-gram hit rate on the test set	-0.96	-0.96

5 Conclusions

In summary, we have shown that, if filtered, web text can be successfully used for training language models of conversational speech, outperforming some other out-of-domain (BN) and small domain-specific (Meetings) sources of data. We have also found that by combining LMs from different domains with class-dependent interpolation (particularly when each of the top 100 words forms its own class), we achieve lower WER than if we use the standard approach where mixture weights depend only on the data source. Recognition experiments show a significant reduction in WER (1.3-2.3% absolute) due to additional training data and class-based interpolation. The class-based mixture consistently outperformed the traditional mixture as we compared the two types of models at various sizes. The experiments have also provided evidence of a very high degree of correlation between model perplexity on the test data and WER, though this is in contradiction to other reported results and needs further investigation.

References

[1] J. Bellegarda. Exploiting both local and global constraints for multispan statistical language modeling. In *Proc. ICASSP*, pages II:677–680, 1998.

[2] A. Berger and R. Miller. Just-in-time language modeling. In *Proc. ICASSP*, pages II:705–708, 1998.

[3] S. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. *Computer Speech and Language*, 13(4):359–394, 1999.

[4] P. Clarkson and A. Robinson. Language model adaptation using mixtures and an exponentially decaying cache. In *Proc. ICASSP*, pages II:799–802, 1997.

[5] J. Godfrey, E. Holliman, and J. McDaniel. Switchboard: telephone speech corpus for research and development. In *Proc. ICASSP*, pages I:517–520, 1992.

[6] R. Iyer and M. Ostendorf. Modeling long range dependencies in languages. In *Proc. ICSLP*, pages 236–239, 1996.

[7] R. Iyer and M. Ostendorf. Transforming out-of-domain estimates to improve in-domain language models. In *Proc. Eurospeech*, volume 4, pages 1975–1978, 1997.

[8] R. Iyer and M. Ostendorf. Relevance weighting for combining multi-domain data for n-gram language modeling. *Computer Speech and Language*, 13(3):267–282, 1999.

[9] R. Iyer, M. Ostendorf, and M. Meteer. Analyzing and predicting language model improvements. In *IEEE Workshop on Speech Recognition and Understanding Proceedings*, pages 254–261, 1997.

[10] D. Klakow. Selecting articles from the language model training corpus. In *Proc. ICASSP*, pages III:1695–1698, 2000.

[11] F. Liu et al. IBM Switchboard progress and evaluation site report. In *LVCSR Workshop*, Gaithersburg, MD, 1995. National Institute of Standards and Technology.

[12] M. Mahajan, D. Beeferman, and D. Huang. Improved topic-dependent language modeling using information retrieval techniques. In *Proc. ICASSP*, pages I:541–544, 1999.

[13] S. Martin et al. Adaptive topic-dependent language modeling using word-based varigrams. In *Proc. Eurospeech*, pages 3:1447–1450, 1997.

[14] N. Morgan et al. The meeting project at ICSI. In *Proc. Conf. on Human Language Technology*, pages 246–252, 2001.

[15] A. Ratnaparkhi. A maximum entropy part-of-speech tagger. In *Proc. Empirical Methods in Natural Language Processing Conference*, pages 133–141, 1996.

[16] K. Ries. A class based approach to domain adaptation and constraint integration for empirical m-gram models. In *Proc. Eurospeech*, pages 4:1983–1986, 1997.

[17] A. Rudnicky. Language modeling with limited domain data. In *Proc. ARPA Spoken Language Technology Workshop*, pages 66–69, 1995.

[18] S. Schwarm and M. Ostendorf. Text normalization with varied data sources for conversational speech language modeling. In *Proc. ICASSP*, pages I:789–792, 2002.

[19] R. Sproat et al. Normalization of non-standard words. *Computer Speech and Language*, 15(3):287–333, 2001.

[20] A. Stolcke. Entropy-based pruning of backoff language models. In *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, pages 270–274, 1998.

[21] A. Stolcke. SRILM – an extensible language modeling toolkit. In *Proc. Intl. Conf. on Spoken Language Processing*, 2002.

[22] A. Stolcke et al. The SRI March 2000 Hub-5 conversational speech transcription system. In *Proc. NIST Speech Transcription Workshop*, 2000.

[23] M. Weintraub et al. LM95 Project Report: Fast training and portability. Technical Report 1, 1996.

[24] X. Zhu and R. Rosenfeld. Improving trigram language modeling with the world wide web. In *Proc. ICASSP*, pages I:533–536, 2001.