Using Out-of-Domain Data to Improve In-Domain Language Models

Rukmini Iyer, Mari Ostendorf, *Member, IEEE,* and Herb Gish, *Member, IEEE*

***Abstract*— Standard statistical language modeling techniques suffer from sparse-data problems when applied to real tasks in speech recognition, where large amounts of domain-dependent text are not available. In this correspondence, we investigate new approaches to improve sparse application-specific language models by combining domain-dependent and out-of-domain data, including a back-off scheme that effectively leads to context-dependent multiple interpolation weights, and a likelihood-based similarity weighting scheme to discriminatively use data to train a task-specific language model. Experiments with both approaches on a spontaneous speech recognition task (switchboard), lead to reduced word error rate over a domain-specific n-gram language model, giving a larger gain than that obtained with previous brute-force data combination approaches.**

***Index Terms*—Lanugage modeling, out-of-domain training.**

I. Introduction

STATISTICAL language models that model the probability of different word sequences play an important role in state-of-the-art speech recognizers. The most commonly used statistical language modeling technique, also referred to as *n-gram language modeling*, considers the word sequence $w_1, w_2, \cdots, w_T$ to be a Markov process with probability

$$P(w_1, w_2, \cdots, w_T) = \prod_{i=1}^{T+1} P(w_i \mid w_{i-1}, \cdots, w_{i-n+1}) \quad (1)$$

where w_0 and w_{T+1} are sentence boundary markers and n is typically restricted to 2 or 3, a bigram or trigram language model, respectively. The n-gram models are very successful in research domains where large amounts of text data are available for parameter estimation, but they suffer from severe sparse-data problems in many real tasks where little domain-specific data is available for training language models. Different approaches that have been developed to deal with the issue of sparse data include using less specific class language models [1] or models using domain-specific techniques [2]. Unfortunately, class models have not shown performance gains and task-dependent techniques are not easily portable to a new domain.

Another solution to this sparse-data problem involves using large amounts of data from other tasks/domains to improve the in-domain language model, motivated in particular by the increasing amount of data available in the form of newspaper text, transcribed television speech, and web hypertext documents. Many of the current approaches to using out-of-domain information involve simply adding the available data without discriminating for relevance to the given task, or building n-gram language models independently on the different domains and using an optimized interpolation weight to combine the models. In this correspondence, we investigate two alternative approaches to using out-of-domain information: i) a back-off scheme to combine out-of-domain n-grams with the given task-domain language model, and ii) a likelihood-based similarity weighting scheme to focus the available language training data to the specific task domain. Both approaches are domain-independent (i.e., portable) and can easily leverage extensions of n-gram language modeling. In Section II, we describe the two approaches for combining information from different domains in more detail. Section III describes the experimental setup and outlines perplexity and recognition results obtained.

Manuscript received August 6, 1996. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. L. Niles.

R. Iyer and M. Ostendorf are with the College of Engineering, Boston University, Boston, MA 02215 USA (e-mail: mo@raven.bu.edu).

H. Gish is with Bolt, Beranek, and Newman Systems and Technology Department, Cambridge, MA 02138 USA.

Publisher Item Identifier S 1070-9908(97)05979-8.

II. Approaches

A. Combining Models

Models estimated on different domains can be combined using a single interpolation weight, namely

$$P(w_i \mid w_{i-1}) = \lambda P_I(w_i \mid w_{i-1}) + (1-\lambda) P_O(w_i \mid w_{i-1}) \quad (2)$$

where $P_I()$ is the in-domain language model and $P_O()$ is the out-of-domain language model.[1] Both language models need some smoothing scheme to account for unseen n-grams; in all our experiments the Witten–Bell smoothing technique [5] is used. The interpolation weight, λ, can either be estimated using maximum likelihood of a held-out in-domain data set, or heuristically optimized to minimize recognition error on a development test set.

Recently, Besling and Meier [3] proposed a back-off scheme that combines the two language models $P_I()$ and $P_O()$, and this approach is used in [4] where $P_O()$ is a class grammar. Here, we use a modified version of this approach, in which the Witten–Bell smoothing technique is used to estimate context-dependent interpolation weights. Specifically, (2) becomes

$$P_I(w_i \mid w_{i-1}) = \lambda_I(w_{i-1}) P_I^{ML}(w_i \mid w_{i-1}) + (1 - \lambda_I(w_{i-1})) P_O(w_i \mid w_{i-1}) \quad (3)$$

[1] For notational simplicity in the discussion, we use the bigram representation; however, all experiments in this paper use trigram models.

where

$$\lambda_I(w_{i-1}) = \frac{c_I(w_{i-1})}{c_I(w_{i-1}) + r_I(w_{i-1})}. \quad (4)$$

$c_I(w_{i-1})$ is the in-domain count of word w_{i-1}, $r_I(w_{i-1})$ is the in-domain count of unique words following w_{i-1}, $P_I^{ML}(w_i \mid w_{i-1})$ is the conditional relative frequency estimate of bigram (w_i, w_{i-1}), and the out-of-domain bigram estimate, $P_O(w_i \mid w_{i-1})$, is given by

$$P_O(w_i \mid w_{i-1}) = \lambda_O(w_{i-1}) P_O^{ML}(w_i \mid w_{i-1}) + (1 - \lambda_O(w_{i-1})) P_I(w_i). \quad (5)$$

Equations (3)–(5) combine the information from the two domains in order of specificity and relevance, using context-dependent data-driven interpolation weights, so this variable back-off algorithm will be referred to as the *multiple interpolation weight* scheme.

B. Combining Counts

Instead of combining models, one might estimate a single model based on the combined counts of different domains. A brute-force approach is to use the counts from both data sets as is, i.e., estimate the count of a bigram (w_i, w_{i-1}) as

$$C_{bf}(w_i, w_{i-1}) = c_I(w_i, w_{i-1}) + c_O(w_i, w_{i-1}). \quad (6)$$

There are two issues that arise with weighted training: estimating weights to reflect domain similarity and estimating the n-gram model parameters using these weighted counts.

The similarity weight used here is an approximate posterior probability of the target domain given the data sample. To estimate robust weights, the weighting is done at the document level, e.g., at the article level for newspaper text and at the show level for a transcribed television speech corpus. Given two n-gram models, for the in-domain data and the out-of-domain data respectively, the similarity weight for a document W^i is given as

$$P(I \mid W^i) = \frac{P_I(W^i)P(I)}{P_I(W^i)P(I) + P_O(W^i)P(O)} \quad (7)$$

where I represents the task domain and O represents the out-of-domain class. In our experiments, the two classes are assumed to be equally likely and

$$P_k(W^i) = \left(\prod_{j=1}^{n_i} P_k(w_j \mid w_{j-1}) \right)^{\frac{1}{n_i}} \quad (8)$$

where n_i represents the number of words in W^i and $k \in \{I, O\}$. The factor $\frac{1}{n_i}$ is used to average the probability of the data sample and, in our experiments, was observed to provide improved weighting over a true posterior probability.

Parameter estimation using weighted counts, in the context of the Witten–Bell back-off technique, has previously been developed in [6]. The bigram probability for the combined language model is

$$P(w_i \mid w_{i-1}) = (1 - \lambda(w_{i-1}))P^{ML}(w_i \mid w_{i-1}) + \lambda(w_{i-1})P(w_i) \quad (9)$$

TABLE I
PERPLEXITY AND RECOGNITION RESULTS ON THE DEV95 TEST SET USING THE BRUTE-FORCE (BF) APPROACH, THE SINGLE INTERPOLATION WEIGHT (SI), MULTIPLE INTERPOLATION WEIGHTS (MI) IN A BACK-OFF SCHEME, AND THE SIMILARITY WEIGHTING (WT) SCHEME, COMPARED TO THE BASELINE IN-DOMAIN LANGUAGE MODEL

LM Training	Technique	Perplexity	Word error rate (%)
SWBD	Baseline	91	44.0
SWBD+CNN	BF	102	43.4
SWBD+CNN	SI	79	43.1
SWBD+CNN	MI	120	42.9
SWBD+CNN	WT	100	42.9

where the unigram back-off mass is

$$\lambda(w_{i-1}) = \frac{\sum_q \frac{C_{wt}(w_q, w_{i-1})}{C_{bf}(w_q, w_{i-1})}}{\sum_q C_{wt}(w_q) + \sum_q \frac{C_{wt}(w_q, w_{i-1})}{C_{bf}(w_q, w_{i-1})}} \quad (10)$$

and the combined count of a bigram (w_i, w_{i-1}) with the weighted documents is given as

$$C_{wt}(w_i, w_{i-1}) = c_I(w_i, w_{i-1}) + \sum_j P(I \mid W^j) c_O^j(w_i, w_{i-1}) \quad (11)$$

where $c_O^j(w_i, w_{i-1})$ is the bigram count in the jth document.

III. EXPERIMENTS

A. Paradigm

Perplexity and recognition experiments were run on the switchboard (SWBD) task [7], transcribing spontaneous, telephone conversations. The training data included 2.1 million words of in-domain text from switchboard, 118 million words of transcribed television speech from Cable News Network (CNN), and a subset of the North American Business (NAB) newspaper text data. The CNN data included spontaneous conversational speech from talk shows, as well as read speech in the form of news and voice-overs.

Results were obtained using the N-best rescoring formalism [8] with the N-best hypotheses generated by the BBN Byblos System [9], a speaker-independent hidden Markov model (HMM) system. More specifically, the top N sentence hypotheses ($N = 100$) are rescored by the language model, and a weighted combination of the HMM and new language model scores is used to rerank the hypotheses. The top ranking hypothesis is used as the recognized output. The weights for recombination are estimated on development test sets (1995 and 1996 internal BBN test sets, referred to as dev95 and dev96, respectively) and held fixed for the evaluation test set (1996 official switchboard evaluation test set, referred to as eval96).

B. Results

Table I compares the model and data combination techniques described in Section II, and provides the results obtained using only a switchboard trigram model as a baseline. The CNN data provide a reduction in word error rate using

TABLE II
RECOGNITION RESULTS ON THE DEV96 AND EVAL96 TEST SETS, COMPARING THE BEST CASE MI AND WT SCHEMES TO THE BASELINE LANGUAGE MODEL

LM Training	Technique	Dev (WER %)	Eval (WER %)
SWBD	Baseline	41.1	44.6
SWBD+CNN	MI	40.5	44.2
SWBD+CNN	WT	39.9	43.8

the brute-force (BF) technique of (6). However, we observe an additional gain using the likelihood-based similarity weighting (WT) scheme to obtain our best performance of 42.9% as compared to 44.0%. We repeated the above experiment with the NAB corpus and observed a smaller (1.4%) gain using the BF approach, but not additional improvement from the WT scheme. This may not be surprising, given that the NAB corpus has very little similarity to the switchboard domain, but the corpus does contain n-grams that could improve the trigram hit rate in the switchboard task. The CNN trigram model reduces both perplexity as well as word error rate when interpolated with the in-domain switchboard trigram model. The back-off scheme (MI) provides a small gain over the standard technique of using a single interpolation weight (SI), both of which outperform the BF data combination approach. Note that the best word accuracy results are obtained with models that actually increase perplexity.

We repeated the best-case experiments with the CNN data—using the back-off scheme and the weighting approach—on dev96 and eval96 test sets. Table II gives the recognition performance with score-combination weights optimized on the dev96 test set and held fixed for the eval96 test set, showing a small advantage to the weighting approach.

In summary, we find that both back-off and similarity weighting techniques give improved recognition performance over more simplistic methods of using out-of-domain data. Moreover, the fact that perplexity increases are associated with the better recognition results suggests that test set likelihood may not be a good criterion for estimating model interpolation weights.

REFERENCES

[1] P. Brown *et al.*, "Class-based n-gram language models of natural language," *Computat. Linguist.*, vol. 18, pp. 467–479, 1992.
[2] R. Rohlicek, Y. Chow, and S. Roucos, "Statistical language modeling using a small corpus from an application domain," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, New York, NY, 1988, vol. 1, pp. 267–270.
[3] S. Besling and H.-G. Meier, "Language model speaker adaptation," in *Proc. Europ. Conf. Speech Communications and Technology*, Sept. 1995, vol. 3, pp. 1755–1759.
[4] T. Niesler and P. Woodland, "Combination of word-based and category-based language models," in *Proc. Int. Conf. Spoken Language Processing*, Oct. 1996, vol. 1, pp. 220–223.
[5] P. Placeway, R. Schwartz, P. Fung, and L. Nguyen, "Estimation of powerful LM from small and large corpora," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, Apr. 1993, vol. 2, pp. 33–36.
[6] R. Iyer, "Language modeling with sentence-level mixtures," Masters thesis, Boston Univ., Boston, MA (anonymous ftp to raven.bu.edu, pub/reports directory), 1994.
[7] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "Switchboard: Telephone speech corpus for research and development," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, Mar. 1992, vol. 1, pp. 517–520.
[8] M. Ostendorf *et al.*, "Integration of diverse recognition methodologies through reevaluation of n-best sentence hypotheses," in *Proc. ARPA Workshop Speech and Natural Language*, Feb. 1991, pp. 83–87.
[9] L. Nguyen *et al*., "The 1994 BBN/BYBLOS speech recognition system," in *Proc. ARPA Workshop Spoken Language Technology*, Mar. 1994, pp. 77–81.