On-line Language Model Biasing for Multi-Pass Automatic Speech Recognition

Sankaranarayanan Ananthakrishnan, Stavros Tsakalidis, Rohit Prasad and Prem Natarajan

Raytheon BBN Technologies
10 Moulton Street
Cambridge, MA 02138, U.S.A.
{sanantha,stsakali,rprasad,prem}@bbn.com

Abstract

The language model (LM) is a critical component in statistical automatic speech recognition (ASR) systems, serving to establish a probability distribution over the hypothesis space. In typical use, the LM is trained off-line and remains static at run-time. While cache LMs, dialogue/style adaptation, and information retrieval-based biasing provide some ability for modifying the LM at run-time, they are limited in scope, susceptible to recognition error, place restrictions on the training data and/or test sets, or cannot be implemented for on-line, interactive systems. In this paper, we describe a novel LM biasing method suitable for multi-pass ASR systems. We use k-best lists from the initial recognition pass to obtain a confidence-weighted biasing of the LM training corpus. The latter is used to train a LM biased to the test input. The biased LM is used in the second pass to obtain refined hypotheses either by re-decoding or by re-ranking the k-best list. We sketch an on-line implementation of this scheme that lends itself to integration within low-latency systems. The proposed method is robust to recognition error, and operates on individual utterances without the need for dialogue context. The biased LMs provide significant reduction in perplexity and consistent improvement in word error rate (WER) over unbiased, state-of-the-art, large-vocabulary baseline ASR systems. On the Farsi and English test sets, we obtained relative reductions in perplexity of 24.5% and 31.6%, respectively. Additionally, relative reductions of 1.6% and 1.8% in WER were obtained for large-vocabulary Farsi and English ASR, respectively.

Index Terms: speech recognition, language model biasing, multipass ASR, k-best list rescoring

1. Introduction

Automatic speech recognition (ASR) systems typically use *acoustic models* (AMs) to obtain the likelihood of phonemes and words given features derived from the speech signal, and a *language model* (LM) that establishes a prior distribution over candidate transcriptions. The LM generally has a n-gram structure, where the probability of the current word given its immediate, fixed-length history is estimated from a large corpus of relevant text.

Most ASR systems, especially those that are employed in on-line applications, use a static LM. Likelihoods for candidate hypotheses are obtained from n-gram probabilities that do not vary at run-time. This can cause recognition errors if acoustically confusable constructs also occur with low probability in the static LM. The ability to boost probabilities of potentially relevant n-grams beyond their actual frequency of occurrence in the training data can therefore be useful for ASR.

1.1. Previous Work

Some previous work exists on run-time adaptation of ASR LMs. Cache LMs [1, 2] were proposed to exploit the "bursty" nature of words in spoken language to dynamically adapt n-gram probabilities using the most recently hypothesized words; however, they are limited by (a) the absence of a reliable estimation procedure for probabilities of words currently in the cache and (b) lack of robustness to recognition error.

Mixture LMs can be used in conjunction with dialogue context to bias relevant n-gram probabilities at run-time. In this scenario, the LM training corpus is assumed to consist of several sub-corpora, each drawn from different domains, topics, and genres (collectively referred to as *style*). Style-specific LMs are estimated from each sub-corpus, and constitute the mixture components. The probability of an arbitrary n-gram according to the mixture LM is obtained as a linear combination of its probabilities in each component. These coefficients, also known as *interpolation weights*, can be estimated at run-time. Clarkson et al. [2] use the "recognition history" to estimate these interpolation weights using expectation-maximization. Stallard et al. [3] use a term-frequency representation of the dialogue history up to the current instance to estimate the posterior probability of the dialogue originating from each mixture component (cluster), and use these as interpolation weights. The principal disadvantages of this approach are (a) the reliance on training data being partitioned into distinct styles, and (b) the assumption that a test set is derived from a coherent dialogue with a specific style.

Perhaps most relevant to the present discussion is previous work on unsupervised LM adaptation using information retrieval (IR) techniques [4, 5]. Queries are constructed from initial recognition hypotheses using keywords (content words) relevant to the specific story. IR is used to retrieve a set of *relevant* documents from a large collection of text. The retrieved text is used to adapt the model by interpolation with the baseline (static) LM. While this method has been shown to work well, it cannot be integrated within on-line systems with low latency requirements. Moreover, IR operates at the document level, and may therefore select large quantities of irrelevant text, which can degrade the adapted model.

1.2. Proposed Approach

In this paper, we develop a novel LM biasing technique suited for multi-pass ASR systems. The idea is to obtain a word confidence-weighted biasing of the LM training corpus using k-best lists from an initial recognition pass. In other words, each sentence in the LM training corpus is assigned a weight proportional to its "similarity" to the corresponding k-best list.

Similarity is evaluated using sparse vector representations of the k-best lists and LM data. The resulting LM assigns higher probabilities to n-grams potentially relevant to the test input. The biased LM can be used to re-decode the speech input, or re-score the k-best list generated using the static LM. In its basic form, this technique is primarily suited for off-line, multi-pass ASR configurations. However, motivated by simple ideas in linear algebra, we also outline an efficient implementation that transfers much of the computational load off-line to the training phase.

The proposed method offers three principal advantages over other dynamic LM adaptation techniques: (a) increased error-robustness stemming from the use of k-best lists to obtain a confidence-weighted similarity measure; (b) increased generative power due to fine-grained sentence-level biasing of the LM corpus; and (c) ability for biased LMs to be integrated within on-line, low-latency ASR applications.

The remainder of this paper is organized as follows. In Section 2, we introduce the proposed LM biasing technique, and provide the sketch for a computationally efficient implementation. We give a detailed description of the architecture of our state-of-the-art ASR system in Section 3. This section also describes how the Farsi and English large-vocabulary systems were trained. We describe our experimental setup and results of LM biasing in Section 4. Finally, we summarize our findings and discuss potential extensions and improvements in Section 5.

2. Language Model Biasing

In traditional LM estimation, n-gram counts are evaluated assuming unit weight for each sentence in the training corpus. Our approach to LM biasing involves re-distributing these weights to favor sentences that are "similar" to the test input. The proposed approach represents the first-pass ASR k-best list and the LM training sentences in a sparse vector space defined by the vocabulary. Thus, n-grams containing words that occur frequently in the input K-best list will be assigned high probability by the biased LM, and vice-versa.

2.1. Similarity Measure

Let $V = \{v_1, \ldots, v_M\}$ represent the vocabulary of the ASR system. Let $\mathbf{u}$ represent the first-pass k-best list in a M-dimensional vector space, whose m^{th} dimension u_m represents the count of vocabulary item v_m in the list. In other words, we interpret the k-best list as a "document", with $\mathbf{u}$ serving as its *term-frequency* representation. Constructing the term-frequency representation from a k-best list rather than from a single best hypothesis imparts error-robustness to the LM biasing procedure. This is because the majority of the term-frequency vector mass is concentrated around recognized words that appear repeatedly across the list, i.e. high-confidence hypotheses. Analogously, $\mathbf{v}_j$ is the M-dimensional term-frequency representation of j^{th} sentence in the LM training corpus. Traditionally, the cosine similarity measure is used to evaluate the likeness of two term-frequency representations, as shown in Equation 2.1:

$$\omega_j = \frac{1}{\|\mathbf{u}\|\|\mathbf{v}_j\|}\mathbf{u}^T\mathbf{v}_j \tag{2.1}$$

In the above equation, we ensure that both $\mathbf{u}$ and $\mathbf{v}_j$ are normalized to unit L_2-norm. This is a standard procedure that prevents over- or under-estimation of "document" similarity due to sentence length mismatch.

The biased LM is estimated by weighting n-gram counts collected from the j^{th} sentence in the training corpus with the corresponding similarity ω_j. The two main advantages of this biasing scheme over traditional LM adaptation/interpolation techniques are: (a) the LM training data do not have to be separated by source, and (b) it provides a fine-grained weighting of the LM corpus.

2.2. On-line Implementation

While feasible for off-line evaluations, the biasing technique described above is computationally intensive because: (a) LM corpora usually consist of hundreds of thousands or millions of sentences; ω_j must be evaluated at run-time for each of them, and (b) the entire LM must be re-estimated at run-time from n-gram counts weighted by sentence-level k-best similarity.

In order to alleviate the run-time complexity of on-line LM biasing, we present an efficient method for obtaining *biased counts* of an arbitrary n-gram t. We define $\mathbf{c}_t = \left[c_t^1, \ldots, c_t^K\right]^T$ to be the indicator-count vector where c_t^j is the unbiased count of t in sentence j. Let $\omega = [\omega_1, \ldots, \omega_K]^T$ be the vector that represents similarity between the k-best list and each of the K target sentences. Then, the biased count of this n-gram, denoted by $\mathcal{C}^*(t)$, is given by Equation 2.2:

$$\begin{aligned}\mathcal{C}^*(t) &= \mathbf{c}_t^T\omega \\ &= \sum_{j=1}^{K}\frac{1}{\|\mathbf{u}\|\|\mathbf{v}_j\|}c_t^j\mathbf{u}^\mathbf{T}\mathbf{v}_j \\ &= \frac{1}{\|\mathbf{u}\|}\mathbf{u}^T\sum_{j=1}^{K}\frac{1}{\|\mathbf{v}_j\|}c_t^j\mathbf{v}_j \\ &= \frac{1}{\|\mathbf{u}\|}\mathbf{u}^T\mathbf{b}_t\end{aligned} \tag{2.2}$$

In the above equation, $\mathbf{b}_t$ is independent of the input k-best list represented by $\mathbf{u}$. We can therefore pre-compute it off-line during the training phase. At run-time, the biased count of any n-gram can be obtained via a simple dot product. This adds very little on-line time complexity because $\mathbf{u}$ is a sparse vector. The fact that $\mathbf{b}_t$, the weighted sum of many sparse vectors, is technically a dense vector may raise concern about the high space complexity of the proposed approach. In practice, it is not necessary to store and process dense, high-dimensional vectors for each n-gram, because the mass of $\mathbf{b}_t$ is usually concentrated around a small number of words that frequently co-occur with n-gram t. Thus, it can be "sparsified", often with little or no loss of information, greatly reducing storage and processing requirements.

The above formulation allows us to evaluate biased counts and probabilities *on demand* for specific n-grams without re-estimating the entire LM. It also eliminates the need to compute, at run-time, similarity weights for each LM training sentence, which may number in the millions. Typically, relatively few n-gram probabilities need be looked up for rescoring a k-best list. Therefore, this implementation can be used for interactive, on-line systems with low latency requirement.

3. Baseline ASR System

We built high-performance baseline ASR systems for two languages commissioned under the DARPA Transtac speech-to-speech translation initiative, viz. Farsi and English.

3.1. Data Description

Acoustic training data for these languages include 1.5-way (simple answers to questions) and 2-way (full dialog) collections in the force protection domain. Recognition experiments for reporting WER were performed on a validation test set. An additional held-out set was used for development. These sets were randomly selected from the Transtac data.

Training data for the Farsi LM consisted of acoustic transcriptions and Web-harvested data. English LMs were estimated from a smaller text corpus consisting of only domain-relevant acoustic transcriptions and Farsi-to-English translations. In both cases, a baseline trigram LM was trained from the corresponding text corpus. The training text was first filtered to remove duplicate sentences. n-gram counts were obtained from the filtered corpus assuming unit weight for each sentence. Witten-Bell discounting [6] was used for parameter smoothing (we made this choice due to an implementation constraint for the biased LM; in principle, we could use Kneser-Ney smoothing [7] as well). Table 1 summarizes the corpora used in building the acoustic and language models for these languages.

Component	Farsi ASR	English ASR
Acoustic Model	83 hrs	214 hrs
Language Model	46.1M wds	13.5M wds
Development set	3.5 hrs (33k wds)	2.6 hrs (32k wds)
Test set	3.4 hrs (29k wds)	1.3 hrs (16k wds)

Table 1: Training, held-out, and test sets for Farsi and English ASR. Number of LM training sentences: 1.8M (Farsi), 1.2M (English).

3.2. Recognizer Architecture

Recognition was based on the BBN Byblos ASR system that models speech as the output of context dependent phonetic Hidden Markov Models (HMMs). We used a perceptual linear prediction (PLP) front-end, that computes 14 cepstral coefficients and normalized energy for each frame of speech. Mean and variance normalization were applied to the cepstra on a conversation basis, to reduce variability due to the channel/speaker. The acoustic models were estimated in the MPE framework [8].

The BBN system uses a multipass decoding strategy in which models of increasing complexity are used in successive passes in order to refine the recognition hypotheses. Baseline recognition was performed using our multi-pass decoder [9]. The forward pass is a fast-match search that uses a State Tied Mixture (STM) model, and an approximate bigram LM to produce a word lattice. The backward pass is a time-synchronous beam search, employing an approximate trigram language model and within-word quinphone State clustered tied mixture (SCTM) HMMs. The backward pass uses the word lattice and associated scores from the forward pass to perform a detailed search. The output of the backward pass is a word lattice. The decoding is completed with a rescoring pass, operating on the lattice. This makes use of between-word quinphone SCTM acoustic and the baseline LM. The top scoring hypothesis represents the recognition output; a k-best list is also produced. Finally, the decoder uses a low-latency online speaker adaptation procedure where the adaptation statistics are updated continuously [10]. For decoding, we used a 32k word lexicon for Farsi and a 35k word lexicon for English.

4. Experimental Results

We evaluated the proposed LM biasing technique in a k-best list rescoring framework for both Farsi and English. We evaluated the predictive power of the biased LMs using perplexity, and studied their discrimination ability through WER analysis.

4.1. Rescoring k-best Lists

The baseline ASR described in Section 3 was used to generate k-best lists ($k = 75$) from every speech utterance in the Farsi/English development and test sets. For each utterance, we evaluated biasing weights for the LM corpus as a function of the cosine similarity between TF representations of the corresponding k-best list and the training sentences. These TF vectors were based on the vocabulary of all unigrams, bigrams, and trigrams in the k-best list. The frequency of each n-gram was weighted by its length n. Thus, for a given number of occurrences, higher order n-grams received a larger proportion of the total "mass" of the TF representation.

Smoothing techniques such as Witten-Bell and Kneser-Ney redistribute probability mass from maximum-likelihood estimates of seen n-grams across unseen n-grams. The probability mass "stolen" from observed higher-order n-grams is usually a function of the number of unique contexts in which the current word was seen. The latter is unaffected by sentence-level biasing weights. Therefore, too much probability mass will be taken away from seen n-grams if the cosine similarity, which has a range of $[0.0, 1.0]$ is used to weight n-gram counts. In order to obtain a more "sane" smoothed LM, we linearly interpolated sentence-level cosine similarity over the closed intervals $[0.0, 10.0]$ for Farsi and $[0.0, 5.0]$ for English to obtain the final set of biasing weights.

Witten-Bell smoothed trigram LMs were trained for each k-best list in the held-out development and test sets. Baseline LM probabilities in each k-best list were replaced by likelihoods evaluated from the corresponding biased LM. We performed two types of rescoring: (a) using baseline and biased LM likelihoods only, ignoring acoustic scores and other features, and (b) incorporating all features, optimizing feature weights for minimum WER on the development set; the optimized weights were then applied to k-best lists on the test set. In both cases, we optimized n-gram cutoff frequencies of the biased LMs as well as mixing weights with the baseline LM for minimum WER on the development set.

Table 2 summarizes the WER performance on the corresponding test sets in the baseline and biased conditions. We see consistent reduction in test-set WER across languages when biased LMs are used to rescore k-best lists. On the Farsi and English test sets using all k-best list features for re-ranking (bold numbers in Table 2), we obtain relative WER improvements of 1.6% and 1.8%, respectively. For comparison, the lower bounds on k-best list WER (oracle error rates) on the Farsi and English test sets were determined to be 12.2% and 7.3%, respectively.

4.2. Perplexity Analysis

As an additional comparison metric, we also evaluated test set perplexity of the baseline and biased LMs. Table 3 summarizes baseline and biased LM perplexity on the development and test sets for both languages. As expected, the biased LMs exhibit significantly reduced perplexity on these sets across languages. Perplexity on the Farsi development/test sets is significantly higher than that on the corresponding English sets. This

Eval set	WER (LM-only)		WER (Full)	
	Static	**Biased**	**Static**	**Biased**
Farsi dev.	34.2	33.9	31.3	30.9
Farsi test	26.6	26.4	**25.1**	**24.7**
English dev.	16.5	16.0	14.3	14.2
English test	13.3	13.0	**11.2**	**11.0**

Table 2: Word error rate (WER) performance on Farsi and English development/test sets with static and biased LMs. The *WER (LM-only)* column shows error rate when k-best lists are rescored based solely on LM likelihood; *WER (Full)* indicates error rate when all features (including acoustic scores) are used for rescoring.

may be attributed to the agglutinative nature of Farsi, which has a more complex morphology than English.

As demonstrated by significant reduction in perplexity, the proposed biasing technique greatly improves the generative power of the adapted LMs with respect to the corresponding spoken utterance. However, the differences between competing hypotheses in a k-best list, for example, can be very subtle. Thus, even though the k-best oracle error rates are significantly lower than the one-best WER, the biased LMs are only able to recover a fraction of this error. Our analysis reinforces the observation that perplexity reduction is not necessarily a good predictor of WER improvement.

5. Conclusion and Future Directions

State-of-the-art ASR systems, especially those designed for interactive applications, almost always use LMs that are fixed at run-time. Existing methods for LM adaptation typically utilize a window of recently hypothesized words to boost probabilities of n-grams that are more likely given this context. This window may be restricted to the current utterance (e.g. cache LMs), or may be a function of the entire "dialogue" to date. Cache LMs are prone to recognition error, and do not provide a robust estimation procedure for n-gram probabilities of cached words. Dialogue-based adaptation does not bias the LM specifically for the current utterance, and is not useful for test sets consisting of unrelated speech utterances. Moreover, it relies on the LM training data being partitioned into different styles. Information retrieval-based LM adaptation has been shown to work well on large-vocabulary off-line recognition tasks, but is difficult to implement for low-latency ASR applications.

In this paper, we proposed a novel LM biasing technique based on the likeness between k-best lists generated by a baseline ASR system (using a static LM), and the LM training corpus. Likeness is measured by cosine similarity between sparse vector space (term-frequency) representations of the k-best lists and training sentences. We adopted a fine-grained approach, weighting individual sentences, and using the entire, weighted corpus to estimate a biased LM. We illustrated an on-line implementation that greatly improves the time and space efficiency of the proposed method by pre-computing and "sparsifying" sums of term-frequency vectors off-line during the training phase. Thus, our approach can be integrated within online, low-latency ASR systems. We showed that biased LMs yield significant reductions in test set perplexity. We also obtained consistent improvements in WER across multiple languages over strong large-vocabulary baseline systems.

The proposed approach of obtaining utterance-specific LMs using biasing weights greatly improves their generative power with respect to the corresponding utterance. However, a discriminative approach is required to differentiate the often small differences between competing hypotheses in k-best lists or lattices. In the future, we plan to explore maximum-entropy approaches to LM biasing based on lexical triggers obtained from recognition hypotheses.

PPL eval set	Static	Biased	Reduction
Farsi dev.	1082.4	859.7	20.6%
Farsi test.	1365.7	1031.3	24.5%
English dev.	106.9	72.2	32.5%
English test	93.9	64.2	31.6%

Table 3: Perplexity of development and test-set references with static and biased LMs for Farsi and English.

6. References

[1] F. Jelinek, B. Merialdo, S. Roukos, and M. Strauss, "A dynamic language model for speech recognition," in *Proceedings of the workshop on Speech and Natural Language*. Stroudsburg, PA, USA: Association for Computational Linguistics, 1991, pp. 293–295. [Online]. Available: http://dx.doi.org/10.3115/112405.112464

[2] P. Clarkson and A. Robinson, "Language model adaptation using mixtures and an exponentially decaying cache," in *Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '97)-Volume 2*. Washington, DC, USA: IEEE Computer Society, 1997, pp. 799–802. [Online]. Available: http://portal.acm.org/citation.cfm?id=839274.839354

[3] D. Stallard, S. Tsakalidis, and S. Saleem, "Incremental dialog clustering for speech-to-speech translation," in *Proceedings of Interspeech*, Brighton, UK, 2009, pp. 428–431.

[4] L. Chen, J.-L. Gauvain, L. Lamel, G. Adda, and M. Adda, "Using information retrieval methods for language model adaptation," in *Proceedings of Eurospeech*, Aalborg, Denmark, 2001, pp. 255–258.

[5] L. Chen, J.-L. Gauvain, L. Lamel, and G. Adda, "Unsupervised language model adaptation for broadcast news," in *Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '97)*, Hong Kong, 2003, pp. 220–223.

[6] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," in *Proceedings of the 34th annual meeting on Association for Computational Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 1996, pp. 310–318. [Online]. Available: http://dx.doi.org/10.3115/981863.981904

[7] R. Kneser and H. Ney, "Improved backing-off for m-gram language modeling," in *Proceedings of the 1995 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '95)*, Detroit, MI, 1995, pp. 181–184.

[8] D. Povey and P. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proceedings of the 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '02)*, Orlando, FL, 2002, pp. 105–108.

[9] L. Nguyen and R. Schwartz, "Efficient 2-pass n-best decoder," in *Proceedings of Eurospeech*, Rhodes, Greece, 1997, pp. 167–170.

[10] D. Liu, D. Kiecza, A. Srivastava, and F. Kubala, "Online speaker adaptation and tracking for real-time speech recognition," in *Proceedings of Interspeech*, Lisbon, Portugal, 2005, pp. 281–284.