Factored Language Models and Generalized Parallel Backoff

Jeff A. Bilmes **Katrin Kirchhoff**
SSLI-LAB, University of Washington, Dept. of Electrical Engineering
{bilmes,katrin}@ssli.ee.washington.edu

Abstract

We introduce factored language models (FLMs) and generalized parallel backoff (GPB). An FLM represents words as bundles of features (e.g., morphological classes, stems, data-driven clusters, etc.), and induces a probability model covering sequences of bundles rather than just words. GPB extends standard backoff to general conditional probability tables where variables might be heterogeneous types, where no obvious natural (temporal) backoff order exists, and where multiple dynamic backoff strategies are allowed. These methodologies were implemented during the JHU 2002 workshop as extensions to the SRI language modeling toolkit. This paper provides initial perplexity results on both CallHome Arabic and on Penn Treebank Wall Street Journal articles. Significantly, FLMs with GPB can produce bigrams with significantly lower perplexity, sometimes lower than highly-optimized baseline trigrams. In a multi-pass speech recognition context, where bigrams are used to create first-pass bigram lattices or N-best lists, these results are highly relevant.

1 Introduction

The art of statistical language modeling (LM) is to create probability models over words and sentences that trade-off statistical prediction with parameter variance. The field is both diverse and intricate (Rosenfeld, 2000; Chen and Goodman, 1998; Jelinek, 1997; Ney et al., 1994), with many different forms of LMs including maximum-entropy, whole-sentence, adaptive and cache-based, to name a small few. Many models are simply smoothed conditional probability distributions for a word given its preceding history, typically the two preceding words.

In this work, we introduce two new methods for language modeling: *factored language model* (FLM) and *generalized parallel backoff* (GPB). An FLM considers a word as a bundle of features, and GPB is a technique that generalized backoff to arbitrary conditional probability tables. While these techniques can be considered in isolation, the two methods seem particularly suited to each other — in particular, the method of GPB can greatly facilitate the production of FLMs with better performance.

2 Factored Language Models

In a *factored language model*, a word is viewed as a vector of k factors, so that $w_t \equiv \{f_t^1, f_t^2, \ldots, f_t^K\}$. Factors can be anything, including morphological classes, stems, roots, and other such features in highly inflected languages (e.g., Arabic, German, Finnish, etc.), or data-driven word classes or semantic features useful for sparsely inflected languages (e.g., English). Clearly, a two-factor FLM generalizes standard class-based language models, where one factor is the word class and the other is words themselves. An FLM is a model over factors, i.e., $p(f_t^{1:K} | f_{t-1:t-n}^{1:K})$, that can be factored as a product of probabilities of the form $p(f|f_1, f_2, \ldots, f_N)$. Our task is twofold: 1) find an appropriate set of factors, and 2) induce an appropriate statistical model over those factors (i.e., the structure learning problem in graphical models (Bilmes, 2003; Friedman and Koller, 2001)).

3 Generalized Parallel Backoff

An individual FLM probability model can be seen as a directed graphical model over a set of $N + 1$ random variables, with child variable F and N parent variables F_1 through F_N (if factors are words, then $F = W_t$ and $F_i = W_{t-i}$). Two features make an FLM distinct from a standard language model: 1) the variables $\{F, F_1, \ldots, F_N\}$ can be heterogeneous (e.g., words, word clusters, morphological classes, etc.); and 2) there is no obvious natural (e.g., temporal) backoff order as in standard word-based language models. With word-only models, backoff proceeds by dropping first the oldest word, then the next oldest, and so on until only the unigram remains. In $p(f|f_1, f_2, \ldots, f_N)$, however, many of the parent variables might be the same age. Even if the variables have differing seniorities, it is not necessarily best to drop the oldest variable first.

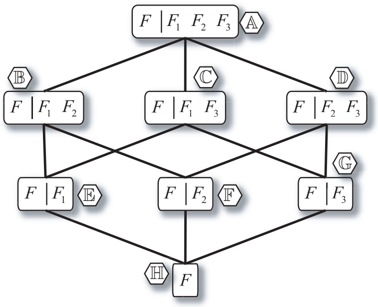

Figure 1: A backoff graph for F with three parent variables F_1, F_2, F_3. The graph shows all possible single-step backoff paths, where exactly one variable is dropped per backoff step. The SRILM-FLM extensions, however, also support multi-level backoff.

We introduce the notion of a *backoff graph* (Figure 1) to depict this issue, which shows the various *backoff paths* from the all-parents case (top graph node) to the unigram (bottom graph node). Many possible backoff paths could be taken. For example, when all variables are words, the path $\mathbb{A}-\mathbb{B}-\mathbb{E}-\mathbb{H}$ corresponds to trigram with standard oldest-first backoff order. The path $\mathbb{A}-\mathbb{D}-\mathbb{G}-\mathbb{H}$ is a reverse-time backoff model. This can be seen as a generalization of lattice-based language modeling (Dupont and Rosenfeld, 1997) where factors consist of words and hierarchically derived word classes.

In our GPB procedure, either a single distinct path is chosen for each gram or multiple parallel paths are used simultaneously. In either case, the set of backoff path(s) that are chosen are determined dynamically (at "run-time") based on the current values of the variables. For example, a path might consist of nodes $\mathbb{A}-(\mathbb{BCD})-(\mathbb{EF})-\mathbb{G}$ where node $\mathbb{A}$ backs off in parallel to the three nodes $\mathbb{BCD}$, node $\mathbb{B}$ backs off to nodes $(\mathbb{EF})$, $\mathbb{C}$ backs off to $(\mathbb{E})$, and $\mathbb{D}$ backs off to $(\mathbb{F})$.

This can be seen as a generalization of the standard backoff equation. In the two parents case, this becomes:

$$p_{GBO}(f|f_1,f_2) = \begin{cases} d_{N(f,f_1,f_2)}\, p_{ML}(f|f_1,f_2) & \text{if } N(f,f_1,f_2) > \tau \\ \alpha(f_1,f_2)\, g(f,f_1,f_2) & \text{otherwise} \end{cases}$$

where $d_{N(f,f_1,f_2)}$ is a standard discount (determining the smoothing method), p_{ML} is the maximum likelihood distribution, $\alpha(f_1,f_2)$ are backoff weights, and $g(f,f_1,f_2)$ is an arbitrary non-negative *backoff function* of its three factor arguments. Standard backoff occurs with $g(f,f_1,f_2) = p_{BO}(f|f_1)$, but the GPB procedures can be obtained by using different g-functions. For example, $g(f,f_1,f_2) = p_{BO}(f|f_2)$ corresponds to a different backoff path, and parallel backoff is obtained by using an appropriate g (see below). As long as g is non-negative, the backoff weights are defined as follows:

$$\alpha(f_1,f_2) = \frac{1 - \sum_{f:N(f,f_1,f_2)>\tau} d_{N(f,f_1,f_2)}\, p_{ML}(f|f_1,f_2)}{\sum_{f:N(f,f_1,f_2)<=\tau} g(f,f_1,f_2)}$$

This equation is non-standard only in the denominator, where one may no longer sum over the factors f only with counts greater than τ. This is because g is not necessarily a distribution (i.e., does not sum to unity). Therefore, backoff weight computation can indeed be more expensive for certain g functions, but this appears not to be prohibitive as demonstrated in the next few sections.

Table 1: CallHome Arabic Results.

LM	parents	backoff function/path(s)	ppl
3-gram	w_1, w_2	- / temporal [2, 1]	173
FLM 3-gram	w_1, w_2, m_1, s_1	- / [2, 1, 4, 3]	178
GPB-FLM 3-gram	w_1, w_2, m_1, s_1	g_1 / [2, 1, (3, 4), 3, 4]	166
2-gram	w_1	- / temporal [1]	175
FLM 2-gram	w_1, m_1	- / [2, 1]	173
FLM 2-gram	w_1, m_1, s_1	- / [1, 2, 3]	179
GPB-FLM 2-gram	w_1, m_1, s_1	g_1 / [1, (2, 3), 2, 3]	167

4 SRILM-FLM extensions

During the recent 2002 JHU workshop (Kirchhoff et al., 2003), significant extensions were made to the SRI language modeling toolkit (Stolcke, 2002) to support arbitrary FLMs and GPB procedures. This uses a graphical-model like specification language, and where many different backoff functions (19 in total) were implemented. Other features include: 1) all SRILM smoothing methods at every node in a backoff graph; 2) graph level skipping; and 3) up to 32 possible parents (e.g., 33-gram). Two of the backoff functions are (in the three parents case):

$$g(f,f_1,f_2,f_3) = p_{GBO}(f|f_{\ell_1}, f_{\ell_2})$$

where

$$(\ell_1,\ell_2) = \underset{(m_1,m_2)\in\{(1,2),(1,3),(2,3)\}}{\text{argmax}} p_{GBO}(f|f_{m_1}, f_{m_2})$$

(call this g_1) or alternatively, where

$$(\ell_1,\ell_2) = \underset{(m_1,m_2)\in\{(1,2),(1,3),(2,3)\}}{\text{argmax}} \frac{N(f, f_{m_1}, f_{m_2})}{|\{f : N(f, f_{m_1}, f_{m_2}) > 0\}|}$$

(call this g_2) where $N()$ is the count function. Implemented backoff functions include maximum/min (normalized) counts/backoff probabilities, products, sums, mins, maxs, (weighted) averages, and geometric means.

5 Results

GPB-FLMs were applied to two corpora and their perplexity was compared with standard optimized vanilla bi- and trigram language models. In the following, we consider as a "bigram" a language model with a temporal history that includes information from no longer than one previous time-step into the past. Therefore, if factors are deterministically derivable from words, a "bigram" might include both the previous words and previous factors as a history. From a decoding state-space perspective, any such bigram would be relatively cheap.

In CallHome-Arabic, words are accompanied with deterministically derived factors: morphological class (M),

Table 2: Penn Treebank WSJ Results.

LM	parents	Backoff function/path(s)	ppl (±std. dev.)
3-gram	w_1, w_2	- / temporal $[2, 1]$	258(±1.2)
2-gram	w_1	- / temporal $[1]$	320(±1.3)
GPB-FLM 2-gram A	w_1, d_1, t_1	g_2 / $[(1, 2, 3), (1, 2), (2, 3), (3, 1), 1, 2, 3]$	266(±1.1)
GPB-FLM 2-gram B	w_1, d_1, f_1	g_2 / $[2, 1]$	276(±1.3)
GPB-FLM 2-gram C	w_1, d_1, c_1	g_2/ $[1, (2, 3), 2, 3]$	275(±1.2)

stems (S), roots (R), and patterns (P). Training data consisted of official training portions of the LDC CallHome ECA corpus plus the CallHome ECA supplement (100 conversations). For testing we used the official 1996 evaluation set. Results are given in Table 1 and show perplexity for: 1) the baseline 3-gram; 2) a FLM 3-gram using morphs and stems; 3) a GPB-FLM 3-gram using morphs, stems and backoff function g_1; 4) the baseline 2-gram; 5) an FLM 2-gram using morphs; 6) an FLM 2-gram using morphs and stems; and 7) an GPB-FLM 2-gram using morphs and stems. Backoff path(s) are depicted by listing the parent number(s) in backoff order. As can be seen, the FLM alone might increase perplexity, but the GPB-FLM decreases it. Also, it is possible to obtain a 2-gram with lower perplexity than the optimized baseline 3-gram.

The Wall Street Journal (WSJ) data is from the Penn Treebank 2 tagged ('88-'89) WSJ collection. Word and POS tag information (T_t) was extracted. The sentence order was randomized to produce 5-fold cross-validation results using (4/5)/(1/5) training/testing sizes. Other factors included the use of a simple deterministic tagger obtained by mapping a word to its most frequent tag (F_t), and word classes obtained using SRILM's `ngram-class` tool with 50 (C_t) and 500 (D_t) classes. Results are given in Table 2. The table shows the baseline 3-gram and 2-gram perplexities, and three GPB-FLMs. Model A uses the true by-hand tag information from the Treebank. To simulate conditions during first-pass decoding, Model B shows the results using the most frequent tag, and Model C uses only the two data-driven word classes. As can be seen, the bigram perplexities are significantly reduced relative to the baseline, almost matching that of the baseline trigram. Note that none of these reduced perplexity bigrams were possible without using one of the novel backoff functions.

6 Discussion

The improved perplexity bigram results mentioned above should ideally be part of a first-pass recognition step of a multi-pass speech recognition system. With a bigram, the decoder search space is not large, so any appreciable LM perplexity reductions should yield comparable word error reductions for a fixed set of acoustic scores *in a first-pass*. For N-best or lattice generation, the oracle error should similarly improve. The use of an FLM with GPB in such a first pass, however, requires a decoder that supports such language models. Therefore, FLMs with GPB will be incorporated into GMTK (Bilmes, 2002), a general purpose graphical model toolkit for speech recognition and language processing. The authors thank Dimitra Vergyri, Andreas Stolcke, and Pat Schone for useful discussions during the JHU'02 workshop.

References

[Bilmes2002] J. Bilmes. 2002. The GMTK documentation. `http://ssli.ee.washington.edu/~bilmes/gmtk`.

[Bilmes2003] J. A. Bilmes. 2003. Graphical models and automatic speech recognition. In R. Rosenfeld, M. Ostendorf, S. Khudanpur, and M. Johnson, editors, *Mathematical Foundations of Speech and Language Processing*. Springer-Verlag, New York.

[Chen and Goodman1998] S. F. Chen and J. Goodman. 1998. An empirical study of smoothing techniques for language modeling. Technical Report Tr-10-98, Center for Research in Computing Technology, Harvard University, Cambridge, Massachusetts, August.

[Dupont and Rosenfeld1997] P. Dupont and R. Rosenfeld. 1997. Lattice based language models. Technical Report CMU-CS-97-173, Carnegie Mellon University, Pittsburgh, PA 15213, September.

[Friedman and Koller2001] N. Friedman and D. Koller. 2001. Learning Bayesian networks from data. In *NIPS 2001 Tutorial Notes*. Neural Information Processing Systems, Vancouver, B.C. Canada.

[Jelinek1997] F. Jelinek. 1997. *Statistical Methods for Speech Recognition*. MIT Press.

[Kirchhoff et al.2003] K. Kirchhoff et al 2003. Novel approaches to arabic speech recognition: Report from the 2002 johns-hopkins summer workshop. In *Proc. IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing*, Hong Kong.

[Ney et al.1994] H. Ney, U. Essen, and R. Kneser. 1994. On structuring probabilistic dependencies in stochastic language modelling. *Computer Speech and Language*, 8:1–38.

[Rosenfeld2000] R. Rosenfeld. 2000. Two decades of statistical language modeling: Where do we go from here? *Proceedings of the IEEE*, 88(8).

[Stolcke2002] A. Stolcke. 2002. SRILM- an extensible language modeling toolkit. In *Proc. Int. Conf. on Spoken Language Processing*, Denver, Colorado, September.