A PAC-Bayesian Approach to Minimum Perplexity Language Modeling

Sujeeth Bharadwaj
University of Illinois
405 N. Mathews Ave.
Urbana, IL 61801, USA
sbhara3@illinois.edu

Mark Hasegawa-Johnson
University of Illinois
405 N. Mathews Ave.
Urbana, IL 61801, USA
jhasegaw@illinois.edu

Abstract

Despite the overwhelming use of statistical language models in speech recognition, machine translation, and several other domains, few high probability guarantees exist on their generalization error. In this paper, we bound the test set perplexity of two popular language models – the n-gram model and class-based n-grams – using PAC-Bayesian theorems for unsupervised learning. We extend the bound to sequence clustering, wherein classes represent longer context such as phrases. The new bound is dominated by the maximum number of sequences represented by each cluster, which is polynomial in the vocabulary size. We show that we can still encourage small sample generalization by sparsifying the cluster assignment probabilities. We incorporate our bound into an efficient HMM-based sequence clustering algorithm and validate the theory with empirical results on the resource management corpus.

1 Introduction

The ability to predict unseen events from a few training examples is the holy grail of statistical language modeling (SLM). Although the final test for any language model is its contribution to the performance of a real system, task-independent metrics such as perplexity are popular for evaluating the general quality of a model. Standard algorithms therefore attempt to minimize perplexity on some previously unobserved test set, assumed to be drawn from the same distribution as the training set. This begets the question of how the test set perplexity is related to training set perplexity – every paper on SLM has an answer, with varying levels of theoretical and empirical justification.

The problem of data sparsity and generalization can be traced back to at least as early as Good (1953), and possibly Laplace, who recognizes that the maximum likelihood (ML) estimate of event frequencies (n-grams) cannot handle unseen events. Smoothing techniques such as the add-one estimator (Lidstone, 1920) and the Good-Turing estimator (Good, 1953) assign a non-zero probability to events that have never been observed in the training set. Recently, Ohannessian and Dahleh (2012) strengthened the theory by showing that Good-Turing estimation is consistent when the data generating process is heavy-tailed. In the context of this paper, smoothing was perhaps the first attempt to bound generalization error, in that it successfully guarantees a finite test set perplexity.

It is evident that smoothing of the n-gram estimate alone is not sufficient. Techniques that incorporate lower and higher order n-grams, such as Katz (1987) smoothing, Jelinek-Mercer (1980) interpolation, and Kneser-Ney (1995) smoothing, have become standard (Rosenfeld, 2000). Chen and Goodman (1999) provide a thorough empirical comparison of smoothing methods and uncover useful relationships between the test set cross-entropy (log perplexity) and the size of the training set, model order, etc. A Bayesian interpretation further explains why some of the techniques (don't) work. Teh (2006) discusses fundamental limitations of the Dirichlet process (Mackay and Peto, 1995) and proposes the hierarchical Pitman-Yor language model as a better way of generating the heavy-tailed (power law) distributions exhibited in natural language.

Instead of directly modeling a heavy-tailed distribution over words, class-based models address data sparsity by estimating n-grams over clusters of words. Intuitively, clustering is a transformation of the event space from the space of word n-grams, in which most events are rare, to the space of class n-grams,

which is more densely measured and therefore requires fewer training examples. Brown et al. (1992) show that the clustering function that maximizes the training data likelihood must also maximize mutual information between adjacent clusters; although several useful clustering algorithms are based on this principle, no provable guarantees currently exist. Moreover, word transitions are never completely captured by the underlying class transitions, and some tradeoff between accurate estimation of frequent events (word n-grams) and generalization to unseen events (class n-grams) is desired – class-based models are therefore often interpolated with word n-grams using some of the previously described Bayesian methods (Rosenfeld, 2000).

Our survey of SLM techniques and their treatment of generalization error has been rather brief and certainly not comprehensive. We focus primarily on n-grams and related models since they have dominated SLM over the last several decades (Rosenfeld, 2000), and therefore serve as a good starting point for further analysis. The existing literature suggests that apart from empirical validation and intuition, no provable guarantees exist on the generalization error of language models. Bayesian techniques work well only to the extent the prior assumptions are valid; in this paper, we present theoretical guarantees that hold irrespective of the correctness of the prior.

Model selection approaches such as the Akaike Information Criterion (AIC) (Akaike, 1973) and its variants (Burnham and Anderson, 2002) quantify the tradeoff between complexity and goodness of fit. In the context of a language model, it can be shown that test set cross entropy is approximately the training set cross entropy plus the number of model parameters. Unfortunately, such bounds are loose and do not provide significant algorithmic insight – at best, they recommend the smallest model that works well on the training set. Chen (2009) obtained a very accurate relationship for exponential language models by estimating the test set performance with linear regression. Although empirical, his approximation leads to better models based on $l_1 + l_2^2$ regularization. Exponential models are often motivated with the minimum discrimination information (MDI) principle, which roughly states that of all distributions satisfying a particular set of features, the exponential family is the centroid (minimizes distortion relative to the farthest possible true distribution) (Rosenfeld, 1996). This does not bound the generalization error in the manner we wish to, but it is nevertheless a useful property that complements Chen's observations.

In this paper, we strive for the best of both worlds – we present PAC-Bayesian theory as a powerful tool for deriving high probability guarantees as well as efficient and well-motivated algorithms. In the next section, we state some useful PAC-Bayesian theorems. In Section 3, we present our main results. We apply the PAC-Bayesian bounds to n-grams, class-based n-grams, and also sequence clustering, where classes represent longer context such as phrases. We show that for sequence clustering, the bound is dominated by the maximum number of sequences represented by each cluster, and consequently requires many more training examples than a class-based model over words. We address this issue by sparsifying the cluster assignment probabilities using the l_α norm, $0 < \alpha < 1$, an effective proxy for the intractable l_0 norm. In Section 4, we show how our bound can be incorporated into an HMM-based clustering algorithm. In Section 5, we validate the theory presented in this paper with some empirical results on the resource management corpus.

2 PAC-Bayesian Bounds

PAC-Bayesian theory is a useful framework for combining frequentist bounds with the notion of a prior. Probably approximately correct (PAC) learning bounds the worst case generalization error of the best hypothesis selected from a hypothesis space – and therefore treats all hypotheses uniformly (Valiant, 1984). PAC-Bayesian bounds, however, place a prior over the hypothesis space while making no assumptions on the data generating distribution (McAllester, 1998). Thus, PAC-Bayesian bounds can both 1) incorporate prior information, and 2) provide frequentist guarantees on the expected performance. They have been successfully applied to classification settings such as the support vector machine (SVM) (McAllester, 2003; Langford, 2005), yielding significantly tighter bounds. Seldin and Tishby (2010) extend the framework to include unsupervised learning tasks such as density estimation and clustering. Since statistical language modeling at its core is a discrete density estimation problem, we focus on the bounds developed by Seldin and Tishby (2010) and summarize key results in the following subsection.

2.1 Unsupervised Learning

Given a d-dimensional product space $\mathcal{X}^{(1)} \times ... \times \mathcal{X}^{(d)}$ and a collection of N samples, S, independent and identically distributed (i.i.d.) according to some unknown distribution $p(x_1, ..., x_d)$ over the product space, we want to estimate $p(x_1, ..., x_d)$ with some model $q(x_1, ..., x_d)$. In the case of clustering (e.g. class-based models), we make the following assumption on $q(x_1, ..., x_d)$ [Note: we make no assumptions on the true distribution $p(x_1, ..., x_d)$]:

$$q(x_1, ..., x_d) = \sum_{c_1,...,c_d} q(c_1, ..., c_d) \prod_{i=1}^{d} q(x_i|c_i) \tag{1}$$

where $c_i = h_i(x_i)$ for some clustering function $h_i : \mathcal{X}^{(i)} \mapsto \mathcal{C}^{(i)}$. We refer to them collectively as a clustering function h, $h = \{h_i\}_{i=1}^{d}$; hence $h : \mathcal{X}^{(1)} \times ... \times \mathcal{X}^{(d)} \mapsto \mathcal{C}^{(1)} \times ... \times \mathcal{C}^{(d)}$. We assume that the original space $\mathcal{X}^{(1)} \times ... \times \mathcal{X}^{(d)}$ has finite cardinality, with $n_i = |\mathcal{X}^{(i)}|$, and likewise for the clustered space $\mathcal{C}^{(1)} \times ... \times \mathcal{C}^{(d)}$, where $m_i = |\mathcal{C}^{(i)}|$ is the number of clusters. We define a hypothesis space, $\mathcal{H}$, to be the space of all possible clustering functions $h \, \epsilon \, \mathcal{H}$.

For $h \, \epsilon \, \mathcal{H}$, we define the distributions $p_h(c_1, ..., c_d) = \sum_{x_1,...,x_d} p(x_1, ..., x_d) \prod_{i=1}^{d} \delta(h_i(x_i) = c_i)$ and $\hat{p}_h(c_1, ..., c_d) = \sum_{x_1,...,x_d} \hat{p}(x_1, ..., x_d) \prod_{i=1}^{d} \delta(h_i(x_i) = c_i)$, where $p(x_1, ..., x_d)$ is the unknown true distribution, and $\hat{p}(x_1, ..., x_d)$ is the empirical (maximum likelihood) estimate. The delta function, $\delta(arg)$, takes a value of 1 only when arg is true, and 0 otherwise. We can extend to the original space with the model assumption in Equation (1). For example, $p_h(x_1, ..., x_d) = \sum_{c_1,...,c_d} p_h(c_1, ..., c_d) \prod_{i=1}^{d} q(x_i|c_i)$.

The key difference between PAC learning and the PAC-Bayesian framework is the following notion of a random predictor, which is a distribution $\mathcal{Q}(h)$, learnt over the hypothesis space $\mathcal{H}$. Inference works as follows: for a new sample $(x_1, ..., x_d)$, we first draw a hypothesis h from $\mathcal{H}$ at random according to the distribution $\mathcal{Q}(h)$. We then return $q(x_1, ..., x_d)$ according to the model described by Equation (1) and the clustering function h. The PAC-Bayesian framework therefore allows for a second level of averaging over $\mathcal{Q}$, and we can define the induced distributions: $p_{\mathcal{Q}}(c_1, ..., c_d) = \sum_h \mathcal{Q}(h) p_h(c_1, ..., c_d)$ and $\hat{p}_{\mathcal{Q}}(c_1, ..., c_d) = \sum_h \mathcal{Q}(h) \hat{p}_h(c_1, ..., c_d)$. Again, we can extend to the original space with $p_{\mathcal{Q}}(x_1, ..., x_d)$ and $\hat{p}_{\mathcal{Q}}(x_1, ..., x_d)$ using the model assumption in Equation (1). Note that $p_{\mathcal{Q}}(x_1, ..., x_d)$ is unknown since $p(x_1, ..., x_d)$ is unknown; but the goal is to bound some notion of generalization error, such as the KL-divergence $\mathbb{KL}(\hat{p}_{\mathcal{Q}}(x_1, ..., x_d)||p_{\mathcal{Q}}(x_1, ..., x_d))$.

The **Change of Measure Inequality (CMI)** (Seldin and Tishby, 2010) is central to almost every PAC-Bayesian bound, so we briefly state it here. For any measurable function $\phi(h)$ on $\mathcal{H}$ and for any distributions $\mathcal{Q}(h)$ and $\mathcal{P}(h)$:

$$\mathbb{E}_{\mathcal{Q}(h)}[\phi(h)] \leq \mathbb{KL}(\mathcal{Q}||\mathcal{P}) + \ln \mathbb{E}_{\mathcal{P}(h)} \left[e^{\phi(h)} \right] \tag{2}$$

where $\mathbb{KL}(\mathcal{Q}||\mathcal{P}) = \mathbb{E}_{\mathcal{Q}(h)} \left[\ln \frac{\mathcal{Q}(h)}{\mathcal{P}(h)} \right]$ is the KL-divergence between $\mathcal{Q}$ and $\mathcal{P}$. The proof is fairly straightforward and is a direct consequence of rewriting $\phi(h)$ as $\ln \left(e^{\phi(h)} \frac{\mathcal{Q}(h)}{\mathcal{P}(h)} \frac{\mathcal{P}(h)}{\mathcal{Q}(h)} \right)$.

Seldin and Tishby (2010) apply the CMI with $\phi(h) = N \cdot \mathbb{KL}(\hat{p}_h(x_1, ..., x_d)||p_h(x_1, ..., x_d))$ and simplify the KL-divergence term by recognizing that 1) $\{q(c_i|x_i)\}_{i=1}^{d}$ defines a distribution over all possible clusterings, and hence $\mathcal{Q} = \{q(c_i|x_i)\}_{i=1}^{d}$; and 2) a specific $\mathcal{P}$, which they call the prior, can be defined without making any assumptions on the true distribution $p(x_1, ..., x_d)$. Note that $\mathcal{P}$ is not a prior in the Bayesian sense: 1) it indicates preference on the structure of the hypothesis, not an assumption on the data generating distribution, although the latter could be a consequence of the former; 2) the bound holds regardless of $\mathcal{P}$; and 3) the bound holds regardless of $\mathcal{Q}$, which is not necessarily the Bayes posterior.

The following prior on $\mathcal{H}$ makes no assumptions on $p(x_1, ..., x_d)$. We present a simplified version of the prior developed by Seldin and Tishby (2010):

$$\mathcal{P}(h) \geq \frac{1}{\exp \left[\sum_{i=1}^{d} m_i \ln n_i + n_i \ln m_i \right]} \tag{3}$$

The prior is based on a combinatorial argument. In order to select a clustering function h_i for some i, we first need to pick a cardinality profile (number of elements per cluster) for the m_i clusters; there are $n_i^{m_i}$ such profiles, hence the first term in the sum. Next, given a cardinality profile, we need to bound the number of ways in which each of the n_i elements can be assigned to the clusters given their sizes; there are at most $m_i^{n_i}$ possibilities, hence the second term in the sum. The CMI with $\phi(h) = N \cdot \mathbb{KL}(\hat{p}_h(x_1, ..., x_d)||p_h(x_1, ..., x_d))$, our modified prior, and a few information theoretic results lead to the following bound.

PAC-Bayesian Clustering: For any distribution p over $\mathcal{X}^{(1)} \times ... \times \mathcal{X}^{(d)}$ and an i.i.d. sample S of size N according to p, with probability at least $1-\delta$, for all distributions of cluster functions $\mathcal{Q} = \{q(c_i|x_i)\}_{i=1}^d$, the following holds:

$$\mathbb{KL}(\hat{p}_{\mathcal{Q}}(x_1, ..., x_d)||p_{\mathcal{Q}}(x_1, ..., x_d)) \leq \frac{\sum_{i=1}^d n_i \ln m_i + K_1}{N} \tag{4}$$

where $K_1 = \sum_{i=1}^d m_i \ln n_i + (M-1)\ln(N+1) + \ln \frac{d+1}{\delta}$, and $M = \prod_{i=1}^d m_i$. Although this shows convergence, in applications such as language modeling, we are interested in directly bounding the test set perplexity or cross-entropy. Seldin and Tishby (2010) smooth $\hat{p}_{\mathcal{Q}}(x_1, ..., x_d)$ to bound $\mathbb{E}_{p(x_1,...,x_d)}[-\ln \hat{p}_{\mathcal{Q}(x_1,...,x_d)}]$ and provide the following useful result based on Equation (4).

Bound on Cross-Entropy: For any probability measure p over $\mathcal{X}^{(1)} \times ... \times \mathcal{X}^{(d)}$ and an i.i.d. sample S of size N according to p, with probability $1-\delta$ for all distributions of cluster functions $\mathcal{Q} = \{q(c_i|x_i)\}_{i=1}^d$:

$$\mathbb{E}_{p(x_1,...,x_d)}[-\ln \hat{p}_{\mathcal{Q}}(x_1, ..., x_d)] \leq -I(\hat{p}_{\mathcal{Q}}(c_1, ..., c_d)) + \ln(M)\sqrt{\frac{\sum_{i=1}^d n_i \ln m_i + K_1}{2N}} + K_2 \tag{5}$$

where $\hat{p}_{\mathcal{Q}}(x_1, ..., x_d)$ is now the *smoothed* empirical estimate induced by $\mathcal{Q}$, $I(\hat{p}_{\mathcal{Q}}(c_1, ..., c_d)) = \sum_{i=1}^d H(\hat{p}_{\mathcal{Q}}(c_i)) - H(\hat{p}_{\mathcal{Q}}(c_1, ..., c_d))$ is the multi-information of the clustering, M and K_1 are as defined in Equation (4), and K_2 is an additional term, $K_2 \geq I(\hat{p}_{\mathcal{Q}}(c_1, ..., c_d))$, and the bound is non-negative.

3 Language Models

Since language modeling is yet another density estimation problem in which we want to minimize the test set perplexity, the bound in Equation (5) readily applies to both word n-grams and class-based n-grams. Note that the bounds are on cross-entropy, which is log perplexity, but we use the two terms almost interchangeably. We are now interested in estimating the unknown true distribution $p(v_1, ..., v_n)$ over the space $\mathcal{V}^n$, where $\mathcal{V}$ is some vocabulary consisting of $V = |\mathcal{V}|$ words. The degenerate case, $d = 1$, $\mathcal{X}^{(1)} = \mathcal{V}^n$, is the case of word n-grams and results in a bound that is dominated by $n_1 = |\mathcal{X}^{(1)}| = V^n$. This suggests that the number of training samples, N, must be on the same order as V^n for the bound (and hence the estimate) to be meaningful.

It is also clear why class-based models are favored whenever they work. In this case, $d = n$, $\mathcal{X}^{(i)} = \mathcal{V}$ for all $1 \leq i \leq d$, and the bound in Equation (5) reduces to something linear in V (since $\forall i$, $n_i = |\mathcal{X}^{(i)}| = V$). Moreover, the clustering function is the same for all i – that is, word clusters do not depend on the position in the n-gram. Assuming K word clusters, the number of training examples, N, only needs to be on the order of $K^n + nV$, achieving effective small sample generalization especially when $K << V$. In the following subsections, we extend the bound to sequences and present a unique approach to regularize the bound.

3.1 Sequence Clustering

We have discussed two extreme cases, namely $d = 1$ and $d = n$, that correspond to word n-grams and class-based n-grams, respectively. In practice, they are often interpolated to retain the advantages of both, as shown in the following model:

$$q(v_1, ..., v_n) = \alpha q(v_1, ..., v_n) + (1-\alpha) \sum_{c_1,...,c_n} q(c_1, ..., c_n) \prod_{i=1}^n q(v_i|c_i) \tag{6}$$

for some $0 < \alpha < 1$. A Bayesian interpretation of the above model is to select between the n-gram and the class-based model with probabilities α and $1 - \alpha$, respectively. In other words, for each n-gram $(v_1, ..., v_n)$, we simply flip an α-biased coin to decide on one of the two models. In this paper, we interpolate across the entire spectrum, $1 \leq d \leq n$, instead of just the extreme cases – that is, we capture clusters over not just words, but also sequences of words (phrases). Previous results by Deligne and Bimbot (1995), Ries et al. (1996), and Justo and Torres (2007) indicate that clustering over phrases is practically useful and leads to significant improvements.

Suppose our goal is to estimate the probability of a trigram, for example, "the cat sat." In the case of $d = 1$, we directly estimate the joint probability $p(the, cat, sat)$. In the standard class-based model, where $d = 3$, we estimate with the model $p(the, cat, sat) = \sum_{c_1,c_2,c_3} p(c_1, c_2, c_3)p(the|c_1)p(cat|c_2)p(sat|c_3)$. The intermediate cases, such as $d = 2$ in this example, are often neglected. The theory we subsequently develop interpolates over all four segmentations, including the missing ones: $p(the, cat, sat) = \sum_{c_1,c_2} p(c_1, c_2)p(the\ cat|c_1)p(sat|c_2)$ as well as $p(the, cat, sat) = \sum_{c_1,c_2} p(c_1, c_2)p(the|c_1)p(cat\ sat|c_2)$.

In general, an n-gram has 2^{n-1} possible segmentations, as illustrated in the previous example. Suppose $f \in \mathcal{F}$ is a particular segmentation from the space of all possible segmentations, and we explicitly define it as the following mapping:

$$f : \mathcal{V}^n \mapsto \mathcal{X}^{(1)} \times ... \times \mathcal{X}^{(d)} \tag{7}$$

where $1 \leq d \leq n$ and f is simply a segmentation that does not modify the joint distribution; that is, $p(v_1, ..., v_n) = p(x_1, ..., x_d)$. If f is fixed *a priori*, we can immediately apply the bounds derived in Equation (5) over the segmented space $\mathcal{X}^{(1)} \times ... \times \mathcal{X}^{(d)}$. This is the case where we decide on a model, such as the standard class-based model ($d = n$), and simply use it.

An extension to the case of interpolated models is straightforward. We modify the hypothesis space $\mathcal{H}$ to not only include all possible clusterings, but also all possible segmentations. The new random prediction $\mathcal{Q}$ over $\mathcal{H}$ works as follows: given an n-gram $(v_1, ..., v_n)$, draw a segmentation $f \in \mathcal{F}$ according to the distribution $\pi = (\pi_1, ..., \pi_{2^{n-1}})$, where the segmentations are indexed by $j = 1, ..., 2^{n-1}$ (the ordering does not matter), and π_j is the probability of drawing segmentation j; pick a clustering as in the random classifier described in Equation (5) for the new segmented space; and estimate $q(v_1, ..., v_n)$ according to the model described by the previous steps. The bound, in terms of π, is given below.

PAC-Bayes Sequence Clustering: For any probability measure p over $\mathcal{V}^n$, and an i.i.d. sample S of size N drawn according to p, with probability $1 - \delta$ for all distributions of segmentations π and for all distributions of cluster functions $\mathcal{Q}$:

$$\mathbb{E}_{p(v_1,...,v_n)}[-\ln \hat{p}_{\mathcal{Q}}(v_1, ..., v_n)] \leq \sum_{j=1}^{2^{n-1}} \left(K_3(j) + \ln(M(j)) \sqrt{\frac{\sum_{i=1}^{d(j)} V^{a_i(j)} \ln m_i(j) + K_1(j)}{2N}} \right) \pi_j$$
$$K_3(j) = -I(\hat{p}_{\mathcal{Q}}(c_1, ..., c_{d(j)})) + K_2(j) \tag{8}$$

where $\forall j\ \forall i,\ 1 \leq a_i(j) \leq n$, and $\forall j,\ \sum_{i=1}^{d(j)} a_i(j) = n$, and $V^{a_i(j)}$ simply replaces n_i in Equation (5) for a given j. The term $K_2(j)$ is from Equation (5). Note that all terms such as $m_i(j)$, the number of clusters corresponding to the space, their product $M(j)$, and additional terms $K_1(j)$, $K_2(j)$ now depend on the segmentation j since $X^{(i)}$ and $d(j)$ depend on j.

We can favor certain segmentations (e.g. those that require few training examples), but note that the bound above is true regardless of the distribution over possible segmentations, π. Also, the bound is dominated by the exponent $a_i(j)$ and the constraint $\sum_{i=1}^{d(j)} a_i(j) = n$. Hence, the bound is polynomial in V for all segmentations except the standard class-based setting where $d(j) = n$, in which case $\forall i,\ a_i(j) = 1$. For example, if $d(j) = n - 1$ for some segmentation j, there exists some i such that $a_i(j) = 2$ and hence represents clusters of bigrams. If $d(j) = n - 2$, there exists some segmentation j, and a space i such that $a_i(j) = 3$, and so on until $d(j) = 1$, and this is the case of word n-grams where $a_1(j) = n$.

3.2 Bound Minimization

Imposing the restriction $\forall j \; \forall i, \; a_i(j) = 1$ is simple, and although it can guarantee the small-sample benefits of a standard class-based model, it is not a useful strategy for incorporating the constraint. Since $a_i(j)$ corresponds to the original space $\mathcal{X}^{(i)}$ for a given j, restricting $a_i(j)$ would restrict $\mathcal{X}^{(i)}$ to an $a \; priori$, fixed set of V elements. To learn the best possible set of V elements, however, we need to minimize the *effective* size of $\mathcal{X}^{(i)}$. For example, suppose we are estimating trigrams over $\mathcal{V}^3$ using the following segmentation: $\mathcal{X}^{(1)} = \mathcal{V}$ and $\mathcal{X}^{(2)} = \mathcal{V}^2$ – i.e. a bigram over clusters of words and clusters of word bigrams. The unconstrained bound is dominated by $\mathcal{X}^{(2)}$. We can restrict the *effective* size of $\mathcal{X}^{(2)}$ by assigning zero probability to the vast majority of its elements, by constraining the hypothesis space to consider only cluster assignment functions $q(x_i|c_i)$ in which $n_2 << V^2$ of the elements have nonzero probability. Thus, every word sequence in $\mathcal{V}^d$ can be generated by the $d = n$ segmentation, but every other segmentation is constrained to generate at most a subset of $\mathcal{V}^d$ with nonzero probability.

We achieve this by imposing the restriction on the random predictor $\mathcal{Q}$. By Bayes rule, $q(c_i|x_i) = \frac{q(x_i|c_i)q(c_i)}{q(x_i)}$ and we can alternatively define $\mathcal{Q}$ as $\mathcal{Q} = \{q(c_i), q(x_i), q(x_i|c_i)\}_{i=1}^{d}$. Our goal is to learn a $\mathcal{Q}$ that minimizes the RHS of Equation (5), which includes maximizing the multi-information term, as well as constraining n_i. As expected, $q(x_i)$ controls the absolute size of $\mathcal{X}^{(i)}$ and $q(x_i|c_i)$ controls the effective size based on the clustering. The dominant term in all of our bounds is n_i (or a_i, with $n_i = V^{a_i}$), which results from the second term in the prior defined in Equation (3), since it bounds the number of ways in which the n_i items can be assigned to the m_i clusters. Alternatively, we can represent this quantity with an upper bound, $\left(\sum_{c_i} \|q(x_i|c_i)\|_0\right) \ln m_i$. We can write $q(x_i) = \sum_{c_i} q(x_i|c_i)q(c_i)$, and $n_i = \|q(x_i)\|_0 = \|\sum_{c_i} q(x_i|c_i)q(c_i)\|_0$; by the triangle inequality and scale invariance of the l_0 norm, this is less than or equal to $\sum_{c_i} \|q(x_i|c_i)\|_0$. We therefore limit the upper bound, $\sum_{c_i} \|q(x_i|c_i)\|_0$, by sparsifying $q(x_i|c_i)$ for every cluster c_i.

The Optimization Problem: Given some segmentation, we want to find a random predictor $\mathcal{Q}$ – a class-based model over the fixed segmentation – such that the bound in Equation (5) is minimized, which is given by the following optimization problem:

$$\begin{aligned} \underset{\mathcal{Q}}{\text{maximize}} \quad & I(\hat{p}_{\mathcal{Q}}(c_1, ..., c_d)) \\ \text{subject to} \quad & \|q(x_i|c_i)\|_0 \leq V, \; \forall \; c_i \, \epsilon \, \mathcal{C}^{(i)}, i = 1, \ldots, d \end{aligned} \tag{9}$$

Since such optimization problems are known to be NP-complete, we use a computationally tractable proxy. The standard practice is to use the l_1 norm instead of the l_0 norm; although non-convex, we resort to the l_α norm, $0 < \alpha < 1$, since $q(x_i|c_i)$ is a probability vector with a fixed l_1 norm. We therefore solve the following problem:

$$\begin{aligned} \underset{\mathcal{Q}}{\text{maximize}} \quad & I(\hat{p}_{\mathcal{Q}}(c_1, ..., c_d)) \\ \text{subject to} \quad & \|q(x_i|c_i)\|_\alpha \leq V, \; \forall \; c_i \, \epsilon \, \mathcal{C}^{(i)}, i = 1, \ldots, d \end{aligned} \tag{10}$$

We have shown that one way to regularize the bound for a non-trivial sequence clustering problem, regardless of whether the segmentation is fixed or if we are interpolating across all segmentations, is to sparsify the cluster assignment probabilities for every cluster. There are many ways to sparsify a probability vector (Pilanci et al., 2012; Kyrillidis et al., 2013), and we select the l_α norm, $0 < \alpha < 1$, for its simplicity and success in other applications (Chartrand and Staneva, 2008). Our approach guarantees manageable bounds on the test set cross-entropy for a general class of SLMs, without making any assumptions on the true distribution $p(v_1, ..., v_n)$.

The Bayesian Connection A Bayesian interpretation of our regularization provides additional insight into other successful models, such as the hierarchical Pitman-Yor language model (HPYLM). In our approach, we impose the restriction $\|q(x_i|c_i)\|_\alpha \leq V$, $0 < \alpha < 1$, for every cluster c_i. It can be shown that this is equivalent to a sub-exponential prior on $q(x_i|c_i)$ (Hastie et al., 2009). Since $q(x_i) = \sum_{c_i} q(x_i|c_i)q(c_i)$ and we make the assumption that $q(x_i|c_i)$ is sub-exponential for every c_i, we are consequently assuming that $q(x_i)$ is also sub-exponential. Although the PAC-Bayesian bounds hold regardless of the true distribution, our regularization technique implicitly assumes that it is heavy-tailed.

The key to HPYLM's success within the Bayesian setting is a better prior that matches the heavy-tailed distribution of natural language (Teh, 2006) – the regularization approach developed in this paper reassuringly corresponds to the assumption that the true distribution is heavy-tailed (sub-exponential). On the other hand, it may be possible to derive provable guarantees for HPYLM within the context of our clustering model. The main difference between HPYLM and the less successful Dirichlet process (DP) is the Chinese restaurant process, which assigns new tables (clusters) to customers (samples) much more aggressively in the former model than in the latter (Teh, 2006). HPYLM therefore has far fewer customers (samples) per table (cluster) than DP, resulting in significantly sparser $q(x_i|c_i)$.

4 An Efficient HMM Algorithm

The hidden Markov model (HMM) is a popular tool for modeling sequences and has been used in several speech and language clustering tasks (Rabiner, 1989; Smyth, 1997; Li and Biswas, 1999). Over its rich history, several techniques, including regularization and sparsification of the HMM parameters, have been developed (Bicego et al., 2007; Bharadwaj et al., 2013). The goal of this section is to show how our bound easily fits into a well-established model such as the HMM.

We can rewrite the standard class-based model by making a Markov assumption on $q(c_1, ..., c_n)$:

$$q(x_1, ..., x_d, c_1, ..., c_d) = \prod_{i=1}^{d} q(x_i|c_i)q(c_i|c_{i-1}) \tag{11}$$

where $\{x_i\}_{i=1}^{d}$ is some segmentation of $(v_1, ..., v_n) \ \epsilon \ \mathcal{V}^n$. The HMM literature refers to c_i as the hidden state, $q(x_i|c_i)$ as the observation probability, and $q(c_i|c_{i-1})$ as the state transition probability (Rabiner, 1989). If we consider each state of the HMM to be a cluster, then as before, $q(c_i|x_i) = q(x_i|c_i)\frac{q(c_i)}{q(x_i)}$ is a distribution over all possible clustering functions. To solve the optimization problem described in Equation (10), we need to maximize the multi-information $I(q(c_1, ..., c_n))$ while satisfying the constraint $\|q(x_i|c_i)\|_\alpha \leq V$. We can rewrite the constrained optimization problem as an unconstrained problem using a Lagrangian, and solve for $q(x_i|c_i)$ with an l_α regularized version of the expectation maximization (EM) algorithm, similar to Bharadwaj et al. (2013).

To maximize the multi-information term $I(q(c_1, ..., c_d))$ in Equation (10), we sparsify the state transition probabilities $q(c_i|c_{i-1})$. This provably works when we use l_α regularization, $0 < \alpha < 1$ for sparsifying $q(c_i|c_{i-1})$. The Renyi α-entropy of a random variable with some probability distribution q is defined to be $H_\alpha(q) = \frac{\alpha}{1-\alpha} \log \|q\|_\alpha$ and there are two useful results we use (Principe, 2010): 1) $\lim_{\alpha \to 1} H_\alpha(q) = H(q)$, where $H(q)$ is the Shannon entropy; and 2) $H_\alpha(q)$ is non-increasing in α. Thus, for $\alpha < 1$, $H_\alpha(q)$ is an upper bound on the Shannon entropy. Since l_α regularization minimizes the Renyi α-entropy, which for $0 < \alpha < 1$ is an upper bound on the Shannon entropy, it effectively maximizes the mutual information between c_i and c_{i-1}, given that $I(\hat{q}_\mathcal{Q}(c_i, c_{i-1})) = H(\hat{q}_\mathcal{Q}(c_i)) - H(\hat{q}_\mathcal{Q}(c_i|c_{i-1}))$.

Thus, we have shown that at least in the context of clustering, sparsifying both the observation probabilities and the state transition probabilities of an HMM using the l_α prior directly minimizes generalization error.

5 Experiments

We test our approach on a subset of the resource management (RM) corpus (Price et al., 1993), which consists of naval commands that span approximately $V = 1000$ words. First, we show that l_α regularization works. Figure 1 shows the estimated test set cross-entropy of an unregularized HMM and of an l_α-regularized HMM as a function of the number of training sentences. We vary the training set size from 10 to 2000 sentences and test the models on 800 sentences; Figure 1 reports the average cross-entropy on brackets of training sizes – 10-100, 110-200, and so on. The l_α-regularized HMM requires additional tunable parameters such as the value of α. To simplify the search on a separate 300 sentence development set, we make a (rather restrictive) assumption that α for both the transition and observation probabilities is the same, and that α is independent of the size of the training set. Our solutions are therefore not optimal, but adequate to demonstrate our claims. To ensure that the cross-entropy is bounded, we smooth all

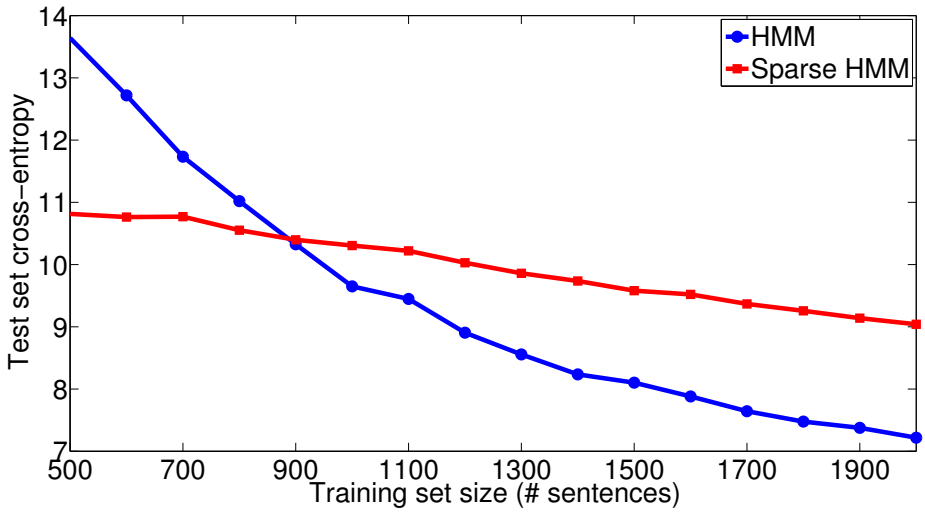

Figure 1: Test set cross-entropy of HMM vs l_α-regularized (sparse) HMM as a function of the number of training sentences

estimates with add-one smoothing. For small training datasets, the unregularized HMM learns models that assign near-zero likelihood to some of the test sentences; hence, we only present results for training set sizes greater than 500 sentences.

Like many other model selection results, Figure 1 suggests that model sparsity is essential when training datasets are small. In this example, about 900 sentences are required for the unregularized HMM to outperform the sparse HMM. In the context of the theory developed in earlier sections, it was shown that test set cross-entropy is proportional to $\frac{n_i}{N}$, where N is the number of training examples. In practical settings, N is fixed; hence, the only strategy for minimizing cross-entropy is to minimize n_i. Figure 1 confirms that l_α regularization successfully sparsifies $q(x_i|c_i)$, the observation probabilities of the HMM, thereby minimizing n_i.

We also compare how the test set cross-entropy improves as a function of the training set size for four different models: 1) a baseline bigram model estimated over words; 2) a baseline class-based model using Brown's algorithm (Brown et al., 1992) with $K = 20$ clusters, learnt over the entire dataset so that it is also representative of knowledge-based approaches in which the true clusters are known $a\ priori$; 3) l_α-regularized HMM with 20 ergodic states; and 4) a special case of 3) in which the state transitions are constrained to artificially form $m_1 = 10$ word clusters (10 states) and $m_2 = 5$ clusters that represent word bigrams (10 states, where the 5 clusters are modeled with 2 left-to-right states each); therefore, the model represents an interpolation between the standard class-based model and word bigrams, but is of the exact same complexity as 2) and 3).

Figure 2 shows the estimated test set cross-entropy for each of the four models. The values of α used in our experiments are $\alpha = 0.7$ for the words only case and $\alpha = 0.9$ for sequences. It is clear from Figure 2 that l_α regularization helps even in the case of a standard class-based model, the bound for which is already linear in V. With fewer than 100 sentences, l_α regularization can both learn the clusters and estimate their transitions reasonably well, and surpasses Brown for training set sizes of $N \geq 800$ sentences. Brown's algorithm in 2) finds clusters such that pairwise mutual information terms are maximized; in 3), we not only maximize the mutual information, but we also reduce the effective V by ensuring that each cluster (or state) specializes and represents as few words as possible. As the number of training examples increases, estimates of class transitions indeed improve, but the class-based assumption itself becomes too restrictive. In 4), which represents an interpolated model, we see the tradeoff achieved by incorporating sequences: for small training sets, the model achieves better generalization than word bigrams, but is worse than the class-based model; and for larger training sets, the interpolated model learns better representations of high frequency events and outperforms the class-based models represented by 2) and 3).

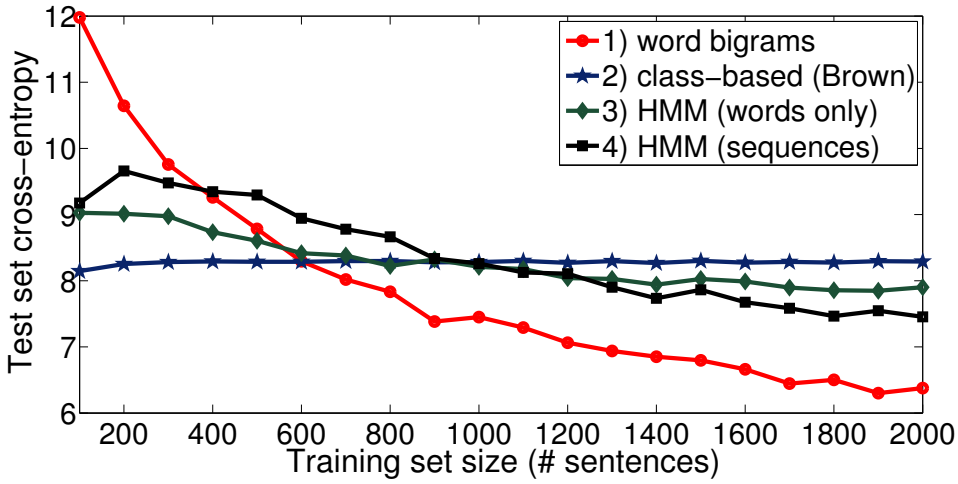

Figure 2: Test set cross-entropy as a function of the number of training sentences for the four settings

The value of α in 3) is 0.7, whereas α in 4) is 0.9; this seems counter-intuitive at first, but note that a smaller α does not necessarily imply sparser observation probabilities; however, it implies a heavier distribution in a Bayesian setting. A Bayesian interpretation therefore suggests that in 4), the model itself is better equipped to cope with heavy tails, whereas a more aggressive α is required in 3).

6 Conclusion

By defining a random clustering model (a model in which there is a distribution over possible cluster assignments, e.g. an HMM), it is possible to specialize published PAC-Bayesian cross-entropy bounds to the cases of n-gram and class-based n-gram estimation. A distribution over segmentations allows derivation of a cross-entropy bound on sequence clustering algorithms, which can be made useful by sparsifying the sequence cluster observation probabilities. An efficient l_α regularization technique can be used to maximize sparsity, thereby minimizing the test set cross-entropy.

7 Acknowledgements

We are grateful to the SST Group at Illinois and the anonymous reviewers for valuable feedback. Thanks also to Jitendra Ajmera, Om Deshmukh, and Ashish Verma for their contributions to the clustering algorithm. This work was supported by the NSF CDI Program Grant Number BCS 0941268 and ARO W9111NF-09-1-0383; the opinions expressed in this work are those of the authors and do not necessarily reflect the views of the funding agencies.

References

Hirotugu Akaike. 1973. Information theory and an extension of the maximum likelihood principle. In *Proceedings of the Second International Symposium on Information Theory*, pages 267–281.

Sujeeth Bharadwaj, Mark Hasegawa-Johnson, , Jitendra Ajmera, Om Deshmukh, and Ashish Verma. 2013. Sparse hidden Markov models for purer clusters. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 3098–3102.

Manuele Bicego, Marco Cristani, and Vittorio Murino. 2007. Sparseness achievement in hidden Markov models. In *Proceedings of of the International Conference on Inage Analysis and Processing*, pages 67–72.

Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jenifer C. Lai, and Robert L. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Kenneth P. Burnham and David R. Anderson. 2002. *Model selection and multi-model inference: a practical information-theoretic approach*. Springer.

Rick Chartrand and Valentina Staneva. 2008. Restricted isometry properties and nonconvex compressive sensing. *Inverse Problems*, 24:1–14.

Stanley F. Chen and Josha Goodman. 1999. An empirical study of smoothing techniques for language modeling. *Computer Speech and Language*, 13:359–394.

Stanley F. Chen. 2009. Performance prediction for exponential language models. In *Proceedings of NAACL HTL*.

Sabine Deligne and Frederic Bimbot. 1995. Language modeling by variable length sequences: theoretical formulation and evaluation of multigrams. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 169–172.

I.J. Good. 1953. The population frequencies of species and the estimation of population parameters. *Biometrica*, 40(3):237–264.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

Frederick Jelinek and Robert L. Mercer. 1980. Interpolated estimation of Markov source parameters from sparse data. In *Proceedings of Workshop on Pattern Recognition in Practice*, pages 381–397.

Raquel Justo and M. Ines Torres. 2007. Different approaches to class-based language models using word segments. *Computer Recognition Systems 2, Advances in Soft Computing*, 45:421–428.

Slava M. Katz. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(3):400–401.

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 181–184.

Anastasios Kyrillidis, Stephen Becker, Volkan Cevher, and Christoph Koch. 2013. Sparse projections onto the simplex. *JMLR: Workshop and Conference Proceedings, Proceedings of the 30th International Conference on Machine Learning*, 28(2):235–243.

John Langford. 2005. Tutorial on practical prediction theory for classification. *The Journal of Machine Learning Research*, 6:273–306.

Cen Li and Gautam Biswas. 1999. Clustering sequence data using hidden Markov model representation. In *Proceedings of the SPIE '99 Conference on Data Mining and Knowledge Discovery*, pages 14–21.

G.J. Lidstone. 1920. Note on the general case of the Bayes-Laplace formula for inductive or $a\ posteriori$ probabilities. *Transactions of the Faculty of Actuaries*, 8:182–192.

David J.C. Mackay and Linda C. Bauman Peto. 1995. A hierarchical Dirichlet language model. *Natural Language Engineering*, 1(03):289–308.

David McAllester. 1998. Some PAC-Bayesian theorems. In *COLT' 98 Proceedings of the eleventh annual conference on Computational Learning Theory*, pages 230–234.

David McAllester. 2003. Simplified PAC-Bayesian margin bounds. In *COLT' 03 Proceedings of the sixteenth annual conference on Computational Learning Theory*, pages 202–215.

Mesrob I. Ohannessian and Munther A. Dahleh. 2012. Rare probability estimation under regularly varying heavy tails. *JMLR: Workshop and Conference Proceedings, 25th Annual Conference on Learning Theory*, 23(21):1–24.

Mert Pilanci, Laurent El Ghaoui, and Venkat Chandrasekaran. 2012. Recovery of sparse probability measures via convex programming. In *Advances in Neural Information Processing Systems (NIPS)*, volume 25, pages 2429–2437.

P. Price, W.M. Fisher, J. Bernstein, and D.S. Pallett. 1993. Resource Management RM 2.0. In *Linguistic Data Consortium, Philadelphia*.

Jose C. Principe. 2010. *Information Theoretic Learning*. Springer.

Lawrence R. Rabiner. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Klaus Ries, Finn Dag Buo, and Alex Waibel. 1996. Class phrase models for language modeling. In *ICSLP '96 Proceedings of the Fourth International Conference on Spoken Language*, pages 398–401.

Ronald Rosenfeld. 1996. A maximum entropy approach to adaptive statistical language modeling. *Computer, Speech, and Language*, 10:187–228.

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

Yevgeny Seldin and Naftali Tishby. 2010. PAC-Bayesian analysis of co-clustering and beyond. *The Journal of Machine Learning Research*, 11:3595–3646.

Padhraic Smyth. 1997. Clustering sequences with hidden Markov models. In *Advances in Neural Information Processing (NIPS)*, volume 9, pages 648–654.

Yee Whye Teh. 2006. A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 985–992.

L.G. Valiant. 1984. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142.