Learning Curved Multinomial Subfamilies for Natural Language Processing and Information Retrieval

Keith Hall KH@CS.BROWN.EDU
Thomas Hofmann TH@CS.BROWN.EDU
Department of Computer Science, Brown University, Box 1910, Providence, RI 02912, USA

Abstract

Many problems in natural language learning and information retrieval involve estimating probabilities in very large discrete state spaces. Dimension reduction as well as clustering techniques in various flavors have been popular choices to deal with the problem of data sparseness. In this paper, we present a general framework for dimension reduction based on curved multinomial subfamilies. The investigated class of models include different geometries as well as various objective functions and algorithms for model fitting. The pursued goal is twofold – to achieve a systematic understanding of the differences and similarities between various models and to empirically investigate their generalization performance on a number of representative data sets.

1. Introduction

Statistical estimation problems in natural language learning and information retrieval often suffer from inherent data sparseness. Although typical text corpora and collections may consist of millions of documents providing large amounts of training data, it is primarily the combinatorial nature of categorical data that challenges machine learning methods. Prominent examples are: (i) Markov chain language models (n-grams) which are an essential part of today's speech recognition systems (Jelinek, 1997), (ii) probabilistic models for syntactic structures like subject-predicate or object-predicate, utilized in the context of parsing (Charniak, 1997), word sense disambiguation, and selectional preferences, and (iii) document-specific language models for information retrieval (Ponte & Croft, 1998; Berger & Lafferty, 1999; Hofmann, 1999b). The common trait of these and other similar problems is the very large state space over which probabilities have to be estimated.

In n-gram language modeling and many natural language processing methods that make use of lexicalization, the number of possible word combinations may exceed any conceivable corpus size. In document-specific language modeling, language models have to be estimated for each individual document, which is a challenge even for simple models based on single word frequencies (unigrams).

This paper pursues two closely related goals: Building upon recent work of Gous (Gous, 1998; Gous, 1999) and our own work (Hofmann, 1999a), it provides a systematic overview and a unifying framework for various methods developed in different research communities, thereby establishing important links and connections. Moreover, we present an experimental comparison of the performance of these methods, not just for a single problem, but for a variety of tasks. By combining theoretical insight about modeling assumptions, objective functions, generalization performance, and optimization strategies with a carefully designed series of experiments, we hope to advance the general methodical knowledge as well as the applicability of the discussed methods.

2. Curved Multinomial Subfamilies

2.1 Problem Setting

We consider the general task of (simultaneously) estimating probability mass functions P_x over some state space $\Omega = \{\omega_1, \ldots, \omega_M\}$ for all elements $x \in \mathcal{X}$ in some (finite or infinite) set $\mathcal{X}$.

Example. *As an illustrative example consider the case of estimating document-specific unigram models: $\mathcal{X}$ is a collection of documents, Ω the vocabulary and each P_x corresponds to a document-specific unigram language model. $P_x(\omega_i)$ denotes the probability that a term ω_i occurs in document x.*

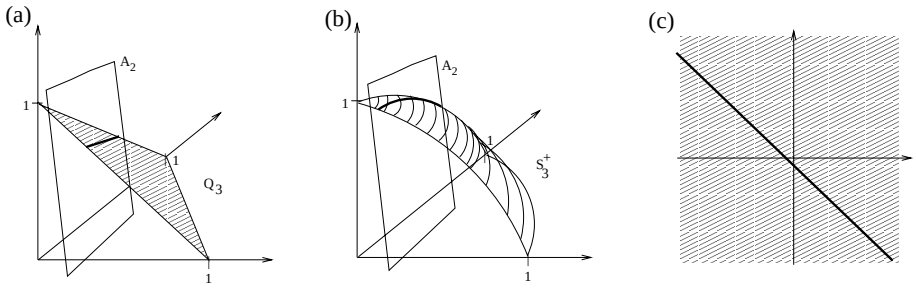

Figure 1. Sketch of (a) the simplex Q_3, (b) S_3^+, the positive part of the 3-sphere, and (c) $\Re^2$, along with intersecting affine spaces. In all three cases the intersection is a one-dimensional subfamily.

In this paper, we take a geometrical point of view and think of each probability mass function P_x as defining a point in $\Re^M$. We can use the Euclidean coordinate system to represent each P_x by a parameter vector θ^x with coefficients $\theta_j^x \equiv P_x(\omega_j)$, where $\theta^x \in Q_M \equiv \{\theta \in \Re^M : \theta \geq 0, \sum_j \theta_j = 1\}$. Q_M denotes the probability simplex, i.e., the convex region spanned by the M canonical basis vectors.

Consider a finite subset $\bar{\mathcal{X}} \subset \mathcal{X}$, $|\bar{\mathcal{X}}| = N$ and assume we generate n_+^x independent samples for each $x \in \bar{\mathcal{X}}$ according to P_x. We introduce count variables n_j^x to refer to the number of times the event ω_j has been observed in the sample for x. Conditioned on the respective sampling sizes the count vectors $n^x = (n_1^x, \ldots, n_M^x)$ have a multinomial distribution with parameters θ^x. The maximum likelihood estimator (MLE) of θ^x is given by $\hat{\theta}_j^x = n_j^x / n_+^x$.

Example. *This sampling model corresponds to the so-called "bag-of-words" view on documents. The counts n_j^x are usually called tf (term-frequency) weights. Using the MLEs $\hat{\theta}^x$, each document is represented as a point on the simplex Q_M.*

In cases, where n_+^x is small compared to M, the MLE has a high variance and will therefore result in a poor estimate. A standard way to overcome this problem is to smooth or shrink the maximum likelihood estimators, e.g., in the simplest case by combining $\hat{\theta}^x$ with the pooled estimator $\hat{\theta}_j^0 = \sum_x n_j^x / L = \sum_x n_+^x \hat{\theta}_j^x / L$, where $L = \sum_x \sum_j n_j^x$ is the total number of samples. More elaborate combination schemes like low order back-off models (Katz, 1987) or deleted interpolation (Bahl et al., 1983) are available, if an additional structure on $\mathcal{X}$ exists.

2.2 Curved Multinomial Subfamilies

Parameterizations of the Multinomial Family So far, we have utilized a parameterization θ of the multinomial family, $P(\omega_j) = \theta_j$ which is usually called the *mean-value* parameterization. Yet, there are two other parameterizations that are relevant in our context. The first one is the (non-minimal) canonical parameterization with $\eta \in \Re^M$, $P(\omega_j) = \exp[\eta_j] / \sum_{j'} \exp[\eta_{j'}]$. The second parameterization is the so-called spherical parameterization (Kass & Vos, 1997) with $\rho \in S_M^+$, the non-negative part S_M^+ of the unit M–sphere S_M, $P(\omega_j) = \rho_j^2$.

What is the rationale behind these transformations? Loosely speaking, the canonical parameterization is important, because log-probabilities of independent events are additive and natural parameters define an affine coordinate system for log-probabilities.[1] The spherical parameterization has the advantage to provide an *isometric embedding* with respect to the Fisher information. Intuitively, the Fisher information defines a metric and thus induces an intrinsic distance function between probability mass functions. In the spherical parameterization, the Fisher distance between two multinomials is proportional to the corresponding arc length on the sphere.[2]

Example. *Revisiting the case of document-specific language models, we might represent a document collection as a "point cloud" in three different ways. In θ–space as points on the simplex Q_M, in η–space as points in $\Re^M$, and in ρ–space as points on the non-negative part of the unit sphere S_M^+.*

The general approach we pursue is to obtain estimates for θ^x by parametric dimension reduction techniques that can be used, e.g., as back-off models in combination with the MLEs. What we mean here by dimension reduction is a many-to-one mapping of points in some

[1] The interested reader is referred to (Murray & Rice, 1993) for an in-depth treatment of this subject.

[2] (Kass & Vos, 1997; Gous, 1999) discuss the spherical parameterization in detail and also point out the relation of Fisher distance to Hellinger distance and to variance stabilizing transformations.

high-dimensional space to points in a low-dimensional (say K dimensional) subspace, where this subspace is chosen in a data-dependent manner. We thus have to address two problems: (i) How do we identify the optimal subspace? And, (ii) how do we project points to this subspace?

More formally stated, our goal is (i) to learn K–dimensional multinomial subfamilies $F_K(\phi)$, parameterized by some parameter ϕ and (ii) to compute for each $x \in \mathcal{X}$ an optimal location τ^x within $F_K(\phi)$. Since the multinomial family is an exponential family, each $F_K(\phi)$ defines a curved exponential subfamily (CESF).

Parameterization for Affine Subspaces The subfamilies we consider are defined by intersection of the simplex, the hyper-sphere, and $\Re^2$, respectively, with affine subspaces (cf. Figure 1). The problem of finding an appropriate representation of subfamilies thus reduces to the problem of parameterizing K–dimensional affine subspaces of $\Re^M$.

It is known that one can define a manifold structure – called a Grassmann manifold – on the set of all K-dimensional affine subspaces of $\Re^M$, which has dimension $(K+1)(M-K)$. Following (Gous, 1998), one (minimal) way to parameterize these subspaces by $\phi = (\alpha, \Lambda)$ is as follows: Define $A_K(\alpha, \Lambda) = \{p \in \Re^M : p = \alpha + \Lambda\tau,\ \tau \in \Re^K\}$, where $\Lambda^T\Lambda = I$, $\alpha^T\Lambda = 0$, and the upper $K \times K$ submatrix of Λ is lower triangular.

Another (redundant) way to specify an affine subspace A_K is by affine combination with $K+1$ points. For our purposes, it is most convenient to consider points ϕ^k on the simplex Q_M. Then we can define

$$A_K(\phi) = \left\{ p \in Q_M : p = \sum_{k=1}^{K+1} \tau_k \phi^k,\ \sum_{k=1}^{K+1} \tau_k = 1 \right\}, \quad (1)$$

where $\phi = (\phi^1, \ldots, \phi^{K+1})$. By simple parameter counting we see that this uses $MK + M - K$ free parameters, K^2 more than the minimal parameterization by (α, Λ).

In the sequel, it will also be important to consider the restricted case of convex subregions $\tilde{A}_K(\phi) \subseteq A_K(\phi)$ where the τ-weights fulfill the additional constraint $\tau_k \geq 0$. Note that although the intersection $F_K(\phi) = A_K(\phi) \cap Q_M$ is convex, this region may not be representable as the convex hull of $K+1$ points.[3]

Affine Subfamilies Let us focus on the most intuitive case of affine subfamilies first. An affine subfamily can be defined as the intersection of the probability simplex (θ-space) with an affine space, i.e., $F_K(\phi) = Q_M \cap A_K(\phi)$. This is sketched for Q_3 in Figure 1 (a).[4]

Spherical Subfamilies In (Gous, 1999), spherical subfamilies have been proposed as an alternative to affine and exponential models. Spherical subfamilies are lower dimensional spheres which are embedded on S_M^+. They can be defined by intersecting the sphere with affine spaces, $F_K(\phi) = S_M^+ \cap A_{K+1}(\phi)$ (cf. Figure 1 (b)). As was claimed in (Gous, 1999), one of the main advantages of spherical subfamilies is that they can model subspaces that – geometrically speaking – are close to many corners of Q_M.

Exponential Subfamilies Flat exponential subfamilies, i.e., subfamilies that are themselves exponential, can be obtained by restricting η to some affine subspace. Flat subfamilies have been investigated in (Aitchison, 1982; Hastie & Little, 1987). We will focus in the sequel on the principle profile method of (Hastie & Little, 1987) in the implementation of (Gous, 1998).[5]

3. Learning Curved Multinomial Subfamilies

3.1 Objective Functions

Log–likelihood As pointed out in the introduction, our goal is to come up with better estimates $\theta(\tau^x, \phi)$ for the unknown parameters θ^x. A natural objective function to use in learning subfamilies as well as in projecting points to a subfamily is the log-likelihood,

$$\mathcal{L}(\tau, \phi) \quad = \quad \sum_{x \in \mathcal{X}} \sum_{j=1}^{M} n_j^x \log \theta_j(\tau^x, \phi)\,. \quad (2)$$

Other Criteria Since it is often difficult to maximize $\mathcal{L}$ directly, one might alternatively think of other objective functions. (Gous, 1999) proposed to minimize the (approximate) χ^2 distance which is known to be locally equivalent to relative entropy,

$$\chi^2(\tau, \phi) = \sum_{x \in \mathcal{X}} n_+^x \sum_j (\hat{\theta}_j^x - \theta_j(\tau^x, \phi))^2 / \hat{\theta}_j^0\,. \quad (3)$$

[3] To give a simple example, cutting a tetrahedron with a plane may result in a region with four corners.

[4] As an aside, we would like to point out that despite the fact that affine spaces $A_K(\phi)$ are flat in the Euclidean geometry, they are not flat in the sense of information geometry, since they are not themselves exponential families. Affine subfamilies in the mean-value parameterization are thus *curved* families.

[5] N. Tisby has pointed out that there is a close relationship with the information bottleneck method (Tishby et al., 1999) which we have not explored here.

This becomes equivalent to correspondence analysis (Greenacre, 1984) in cases where all n_+^i are equal.

In even a cruder approximation, one might ignore the rescaling in the denominator of (3) and just perform a weighted least squares approximation by minimizing

$$SS(\tau, \phi) \quad = \quad \sum_{x \in \mathcal{X}} n_+^x \, \|\hat{\theta}^x - \theta(\tau^x, \phi)\|^2 \, . \qquad (4)$$

In the context of spherical subfamilies, a natural choice is the weighted sum of squared geodesic distances, i.e.,

$$\mathcal{G}(\tau, \phi) = \sum_{x \in \mathcal{X}} n_+^x \arccos^2(\langle \hat{\theta}^x, \theta(\tau^x, \phi) \rangle) \, . \qquad (5)$$

Generalization Performance The key problem in machine learning is to generate models that generalize well on new data. In the setting considered here, we may distinguish two different types of generalization: (I) the predictive performance on additional samples generated for each $x \in \bar{\mathcal{X}}$, or (II) one may generate sample sets for new $x \in \mathcal{X} - \bar{\mathcal{X}}$. In the first case, one will "reuse" both, the ϕ (specifying the family) as well as the τ^x parameters (specifying a member of the family, i.e., a point on the manifold), while in the second case a new τ^x has to be computed for x. In our experiments, we have focused on the first scenario.

Example. *In the first setting one aims at predicting additional words for documents in the given collection. In the second setting one is interested in the performance for new documents x after they have been "folded-in" (i.e., after τ^x has been computed).*

3.2 Optimization Algorithms

Singular Value Decomposition Singular Value Decomposition SVD is a standard technique that can be used to solve least squares problems such as the one in (4). Before applying SVD, one has to shift and rescale the MLEs (cf. (Gous, 1998)), i.e., one performs SVD of the matrix $\tilde{\Theta}$ with rows

$$\tilde{\theta}^x \equiv \sqrt{n_+^x}(\hat{\theta}^x - \hat{\theta}^0), \qquad (6)$$

hence in standard notation $\tilde{\Theta} = USV^T$, where $U \in \Re^{N \times L}$, $S \in \Re^{L \times L}$, $V \in \Re^{M \times L}$, $L = rank(\tilde{\Theta})$. And S is diagonal with ordered singular values, while U and V have orthonormal columns. By keeping only the first $K \leq L$ singular values and the first K columns of V, $\lambda^k = col_k(V)$, $\Lambda = (\lambda^1, \ldots, \lambda^K)$, $\alpha = (I - \Lambda\Lambda^T)\theta^0$, one obtains an optimal (in the weights sum of squares sense) low dimensional affine space $A_K(\phi)$. The low-dimensional estimators are obtained by orthogonal projection of $\hat{\theta}^x$, $\theta^x(\phi) = \alpha + \Lambda\Lambda^T\hat{\theta}^x$.

The SVD can also be applied to minimize the criterion in (3). Essentially, one only needs to appropriately rescale the j-th column before and after performing the SVD by $(\theta_j^0)^{-1/2}$ and $(\theta_j^0)^{1/2}$, respectively.

Iterative SVD-Based Algorithms In (Gous, 1998) a method was proposed to compute optimal subfamilies $F_K(\phi)$ by an iterative scheme which performs an SVD in every step. The idea is to replace the MLEs $\hat{\theta}^x$ by appropriately chosen points $\theta^{x'}$ such that the optimization problem can locally be approximated by a weighted least squares problem in $\theta^{x'}$. We refrain from presenting this method here and refer to (Gous, 1998) for further details. We have used this method to fit the spherical model as well as the exponential family model.[6]

Expectation Maximization Algorithm It is interesting to note that the constrained class of convex subfamilies $\tilde{A}_K(\phi)$ is equivalent to a latent class model which has been discussed under the name of latent class analysis (Gilula & Haberman, 1986) in statistics and was referred to as Probabilistic Latent Semantic Analysis (PLSA) in (Hofmann, 1999a). The key observation is that we may use the following mixture model formulation

$$\theta(\tau^x, \phi) = \sum_{k=1}^{K} \phi^k \tau_k^x \, . \qquad (7)$$

This corresponds to a generative probabilistic model with the following data generation process. For each observation ω_j to be generated for x: (i) pick a value k, $1 \leq k \leq K$, with probability τ_k^x (which can be formally modeled by a latent class variable), (ii) conditioned on the outcome k, generate an observation according to Multinom$(1; \phi^k)$. This probabilistic interpretation of the parameters (ϕ, τ) is possible, since they fulfill the necessary non-negativity and normalization constraints.

The advantage of the mixture model formulation in (7) is that it allows the use of an Expectation Maximization (EM) algorithm (Dempster et al., 1977) for model fitting. For each observation of an ω_j in the sample of x one computes posterior probabilities that the observation was generated from ϕ^k,

$$p_{jk}^x = \frac{\phi_j^k \tau_k^x}{\sum_{k'=1}^{K} \phi_j^{k'} \tau_{k'}^x} \qquad (8)$$

[6] In order to handle zero counts in the exponential model, we have added a (small) additive constant to the counts n_j^x. The value of this constant has been coarsely optimized on hold-out data.

The posterior computed in the E step (8) allow one to calculate the expected sufficient statistics which are then used in the so-called M step

$$\phi_j^k = \frac{\sum_{x\in\mathcal{X}} p_{jk}^x n_j^x}{\sum_{x\in\mathcal{X}}\sum_{j=1}^M p_{jk}^x n_j^x}\,,\; \tau_k^x = \frac{\sum_{j=1}^M p_{jk}^x n_j^x}{n_+^x}\,. \qquad (9)$$

The EM algorithm alternates E and M steps and is known to converge to a local maximum of the log-likelihood in (2).[7]

Tempered EM Algorithm Tempered EM (Hofmann, 1999a) aims at improving the generalization performance of EM based model fitting by introducing an entropic regularization for the posterior probabilities p_{jk}^x. As a starting point, let us derive the E step equation (8) from an optimization principle. It can be shown that $p_j^x = (p_{j1}^x, \ldots, p_{jK}^x)$ maximizes

$$\mathcal{F}_j^x(\beta) = \sum_{k=1}^K p_{jk}^x \left[\log\phi_j^k + \log\tau_k^x - \frac{1}{\beta}\log p_{jk}^x\right] \qquad (10)$$

for $\beta = 1$. By choosing the so-called inverse temperature $\beta < 1$, the entropy contribution in (10) will get more weight and the posterior probabilities will tend to be "smeared out" (closer to the uniform distribution). Then the tempered E step is simply given by

$$p_{jk}^x = \frac{(\phi_j^k\tau_k^x)^\beta}{\sum_{k'=1}^K(\phi_j^{k'}\tau_{k'}^x)^\beta}\,. \qquad (11)$$

In order to determine an optimal choice for the regularization parameter β, we have used hold-out data.[8] As our experiments will demonstrate, the tempered regularization yields substantial improvements in terms of generalization performance.

Multiplicative Parameter Update In (Lee & Seung, 1999), a method for non-negative matrix decomposition has been proposed, which was applied to decompose face images. The utilized objective function is very similar to the log-likelihood in (2) with the only difference that the multinomial sampling model is replaced by a Poisson model. The multinomial model can be obtained from the Poisson model by conditioning on the sample set sizes n_+^x which just leads to a re-scaling of the parameters τ^x. The multiplicative matrix update rule used in (Lee & Seung, 1999) is thus essentially equivalent to the presented EM procedure.

[7] Obviously, there is an identifiability problem. Strictly speaking, the specific convex region spanned by the ϕ^k might not be identifiable. As a consequence, the log-likelihood will have ridges of local maxima and the EM algorithm will converge to a point on a ridge. Notice that the generalization performance for additional samples of $x \in \mathcal{X}$ (type I generalization) is unaffected by this problem.

[8] The typical range for β on the data set used in this paper is $0.55 \leq \beta \leq 0.95$.

4. Experimental Results

4.1 Applications

In our experimental evaluation, we have used three data sets for different problems in natural language learning and information retrieval.

Language Models for Information Retrieval A number of recent information retrieval methods make use of document-specific language models (Ponte & Croft, 1998; Berger & Lafferty, 1999; Hofmann, 1999b). Assume that we have a set of words (vocabulary), $\Omega = \{\omega_1, \ldots, \omega_M\}$ and that for each document $x \in \mathcal{X}$ we want to estimate the probability θ_j^x to observe a particular word ω_j in document x. Using the "bag-of-words" view, we can directly apply the presented dimension reduction framework in order to simultaneously estimate these probabilities for all documents. As was outlined in (Hofmann, 1999b) this approach is closely related to Latent Semantic Analysis (Deerwester et al., 1990), a SVD-based technique that is (up to shifting) equivalent to learning affine families based on (4).

The reason why document-specific language models are so important is that they allow one to compute the probability that a given query was generated from a document. Such schemes have proven to provide a powerful score for document ranking. (Ponte & Croft, 1998; Berger & Lafferty, 1999; Hofmann, 1999b).

Bigram Language Models Low-order Markov model (n–gram models) are by far the most popular class of language models. The key problem in these models is the data sparseness: most combinations of words will typically be never observed in a training corpus. Usually, linear model combination schemes with lower order models are used for smoothing, i.e., trigram estimates are combined with bigrams and bigrams are combined with unigram models. Since, for example, a unigram model can be thought of as a 0–dimensional bigram model, it seems natural to use curved exponential subfamily models as intermediate models. This scheme has been proposed before (Saul & Pereira, 1997) using the convex affine families $\tilde{A}_K(\phi)$ (aggregate Markov model). In our experiments, we restrict attention to the case of low-dimensional estimates for bigram probabilities, although our frame-

	Number of Dimensions											
	10	25	50	75	100	150	200	300	500	1000	2000	opt.
Affine, *SS*	732	674	664	704	715	758	810	865	974	1482	-	**662**
	541	*361*	*250*	*197*	*165*	*128*	*108*	*85*	*63*	*32*	-	
Affine, χ^2	658	586	604	601	623	657	691	753	884	1474	-	**586**
	508	*352*	*263*	*214*	*184*	*146*	*122*	*95*	*68*	*32*	-	
Exponential	2044	2836	4017	5112	5802	6825	7239	7431	7559	8510	-	**2044**
	1752	*1824*	*1544*	*1269*	*950*	*559*	*335*	*149*	*57*	*32*	-	
Spherical	1722	1175	991	942	931	983	1033	1161	1532	2429	-	**919**
	985	*434*	*216*	*144*	*109*	*76*	*61*	*47*	*38*	*36*	-	
Affine, NMF	774	659	599	559	527	506	494	469	441	412	394	**394**
	568	446	349	289	274	255	245	232	201	166	139	
EM (tempered)	637	532	472	446	427	412	408	389	374	368	363	**363**
	535	384	305	240	191	174	158	144	111	91	74	

Table 1. Perplexity results on the Medline1033 data. Numbers in italics are results obtained on the training data. The last column reports the optimal result obtained by picking the best number of dimension K (which might be different from the K values displayed in the table).

	Number of Dimensions										
	10	25	50	75	100	150	200	300	500	1000	opt.
Affine, *SS*	208	192	178	181	181	181	176	177	177	-	**170**
	132	*91*	*65*	*55*	*47*	*38*	*32*	*26*	*21*	-	
Affine, χ^2	285	212	182	162	152	147	147	155	158	-	**145**
	256	*171*	*127*	*92*	*74*	*53*	*41*	*30*	*22*	-	
Exponential	1561	980	597	383	314	293	272	248	229	219	**219**
	1263	*623*	*246*	*108*	*66*	*45*	*35*	*26*	*20*	*19*	
Spherical	527	376	318	291	280	264	257	257	257	-	**257**
	264	*121*	*67*	*49*	*40*	*32*	*29*	*26*	*26*	-	
Affine, NMF	191	154	139	132	125	119	114	107	104	100	**100**
	128	96	84	78	63	55	49	44	39	34	
EM (tempered)	166	140	124	116	110	105	101	96	93	91	**91**
	130	95	68	62	51	44	43	38	33	26	

Table 2. Perplexity results on the Verb1000 data. Numbers in italics are results obtained on the training data. The last column reports the optimal result obtained by picking the best number of dimension K.

work can also be applied in the more general case.

Predicate-Argument Model Lexicalized parsing appropriately augments statistical parsing techniques such as Probabilistic Context Free Grammars (Charniak, 1997). Learning lexical attributes essentially extracts semantic information from the language. Attributes such as predicate-argument structure provide some clues to help disambiguate candidate parses. Unfortunately, estimating distributions over predicate-argument pairs is difficult due to the inherent sparseness of the pairs. In our experiments we consider specifically the predicate-subject pairs.

4.2 The Data Sets

In order to be able to perform a large number of experiments for various models and fitting algorithms, we have chosen three medium-sized data sets in our experimental evaluation.

Medline1033 We have used the Medline1033 test collection which contains 1033 documents from the Medline on-line database. For each document, we have counted the number of occurrences of words, where $M = 1668$ words have been used (those words with at least 10 occurrences in the collection).

Bigram1000 We have collected bigram data from a subset of the Penn Treebank Wall Street Journal corpus, a corpus of Wall Street Journal articles from 1988 and 1989. From this subset, we collected co-occurrence counts for the most frequent 1000 word roots (the WordNet Morphy system was used for morphological reduction). We model the occurrence of a word by the occurrence of the preceding context (in this case, the preceding word).

Verb1000 From a collection of parsed Wall Street Journal articles, we collected subject verb co-occurrence counts for the most frequent 1000 subjects and 1000 verbs. In particular, we identified a head-noun and head-verb form each subject-phrase and

	Number of Dimensions										
	10	25	50	75	100	150	200	300	500	1000	opt.
Affine, SS	119	90	79	74	71	68	67	67	69	-	**67**
	115	*84*	*71*	*64*	*59*	*54*	*51*	*47*	*43*		
Affine, χ^2	186	106	85	78	73	69	68	66	67	-	**66**
	185	*102*	*79*	*71*	*64*	*57*	*52*	*48*	*44*		
Exponential	527	345	267	219	194	158	131	106	92	89	**89**
	511	*317*	*230*	*175*	*145*	*106*	*80*	*57*	*45*	*41*	
Spherical	141	123	122	122	123	123	125	128	135	-	**122**
	134	*111*	*104*	*100*	*97*	*92*	*90*	*89*	*88*		
Affine, NMF	104	83	75	71	69	66	64	63	62	61	**61**
	99	79	66	62	58	57	56	55	55	54	
EM (tempered)	103	81	74	69	67	64	62	61	61	60	**60**
	98	77	68	62	59	55	53	51	50	50	

Table 3. Perplexity results on the Bigram1000 data. Numbers in italics are results obtained on the training data. The last column reports the optimal result obtained by picking the best number of dimension K.

verb-phrase pair and morphologically reduced these words using the WordNet Morphy system. The data set Verb1000 models the occurrence of the verbs given the preceding subject.

4.3 Series of Experiments

We have empirically investigated the following models: (i) affine model with the sum of squared error measure in (4), (ii) affine model with χ^2 fitting criterion in (3), (iii) exponential subfamily model with one-step orthogonal projection, (iv) spherical model with one-step geodesic projection, (v) affine (convex) model trained with the multiplicative matrix update rule and early stopping, and (vi) affine (convex) model trained by tempered EM.

We have used the log-likelihood as an evaluation criterion. Following common practice in language modeling, we report perplexity results:

$$PERP \equiv \exp\left[-\sum_{x\in\tilde{X}}\sum_{j=1}^{M} n_j^x \log\theta_j^x / L\right]. \qquad (12)$$

The data has been split into three sets: (i) a training set consisting of 80% of the observations, (ii) a (hold-out) validation set to determine optimal parameters (such as dimensions, smoothing parameters, number of iterations, optimal β), (iii) a test set on which we report perplexity results. For both, validation and test set, 10% of the data was used.

Perplexity Minimization by Dimension Reduction In the first series of experiment, we have computed (approximately) optimal families $F_K(\phi)$ for various choices of dimensions. Results are reported for $K = 10, 25, 50, 100, 150, 200, 500, 1000, 2000$. Tables 1, 2, and 3 show the results on Medline1033, Verb1000, and Bigram1000, respectively.

	Medline1033	Verb1000	Bigram1000
Baseline	454	91	64
Affine, SS	397	89	61
Affine, χ^2	380	88	61
Spherical	406	90	62
Temp. EM	357	84	59

Table 4. Perplexity results for smoothing the MLE.

Let us compare the SVD–based class of methods first. (i) In accordance with results reported in (Gous, 1998; Gous, 1999), the spherical model shows the best fit on the training data for an intermediate range of dimensions, while the affine model trained with weighted sum of squared error (4) is typically doing best for lower dimensionalities. The exponential model does not fit the training data well. (ii) In term of generalization performance, however, the performance of the spherical model is very poor as can be seen from the large discrepancy between training vs. test set performance. Overall, the affine model based on (3) shows the best generalization performance among the SVD-based approaches.

Comparing SVD–based methods with the iterative optimization techniques based on the mixture model formulation, the latter show substantial and consistent performance gains on all data sets. The regularization in tempered EM proves to be more effective than early stopping. Tempered EM achieves the best results for all data sets and all dimensions.

Low-Dimensional Families as Back-off Models In a second series of experiments, we have evaluated the use of low-dimensional families for the purpose of smoothing the MLE. More precisely, we have investigated a linear interpolation scheme of the type $\bar{\theta}^x = \lambda_1\hat{\theta}^x + \lambda_2\theta(\tau^x,\phi) + \lambda_3\hat{\theta}^0$, where $\lambda_1+\lambda_2+\lambda_3 = 1$.

Table 4 reports the best results achieved with the various dimension reduction methods by optimizing the interpolation parameters on the validation data. The baseline model corresponds to $\lambda_2 = 0$, i.e., a back-off model with $\hat{\theta}^0$ and no dimension reduction model. These results essentially confirm the previous results: likelihood-based approaches outperform SVD-based methods, with tempered EM showing consistently the best performance.

5. Conclusion

We have presented a general framework for probabilistic dimension reduction within the theory of curved multinomial subfamilies. Experimental results have shown that these methods are a flexible and well–suited tool for various applications in natural language learning and information retrieval. Among the discussed geometries and methods, the tempered EM algorithm has consistently outperformed all other techniques in terms of generalization performance.

References

Aitchison, J. (1982). The statistical analysis of compositional data. *Journal of the Royal Statistical Society, Ser. B*, *44*, 137–177.

Bahl, L. R., Jelinek, F., & Mercer, R. L. (1983). A maximum likelihood approach to continuous speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *5*, 179–190.

Berger, A., & Lafferty, J. (1999). Information retrieval as statistical translation. *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.*

Charniak, E. (1997). Statistical techniques for natural language parsing. *AI Magazine*, *18*, 33–44.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, *41*, 391–407.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statist. Soc. B*, *39*, 1–38.

Gilula, Z., & Haberman, S. J. (1986). Canonical analysis of contingency tables by maximum likelihood. *Journal of the American Statistical Association*, *81*, 780–788.

Gous, A. (1998). *Exponential and spherical subfamily models.* Doctoral dissertation, Stanford, Statistics Department.

Gous, A. (1999). Spherical subfamily models. *(submitted for publication).*

Greenacre, M. J. (1984). *Theory and applications of correspondence analysis.* Academic Press.

Hastie, T. J., & Little, F. (1987). Principal profiles. *Proccedings of 19th Symposion on the Interface between Computer Science and Statistics* (pp. 243–249).

Hofmann, T. (1999a). Probabilistic latent semantic analysis. *Proceedings of the 15th Conference on Uncertainty in AI* (pp. 289–296).

Hofmann, T. (1999b). Probabilistic latent semantic indexing. *Proceedings of the 22nd ACM-SIGIR International Conference on Research and Development in Information Retrieval, Berkeley, California* (pp. 50–57).

Jelinek, F. (1997). *Statistical methods for speech recognition.* Cambridge, MA: MIT Press.

Kass, R. E., & Vos, P. W. (1997). *Geometrical foundations of asymptotic inference.* Wiley Series in Probability and Statistics. Wiley.

Katz, S. (1987). Estimation of probabilities for sparse data for the language model component of a speech recogniser. *IEEE Transactions on Acoustics, Speech and Signal Processing*, *35*, 400–401.

Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, 788–791.

Murray, M., & Rice, J. W. (1993). *Differential geometry and statistics.* Chapman & Hall.

Ponte, J., & Croft, W. (1998). A language modeling approach to information retrieval. *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 275–281).

Saul, L., & Pereira, F. (1997). Aggregate and mixed–order Markov models for statistical language processing. *Proceedings of the 2nd International Conference on Empirical Methods in Natural Language Processing* (pp. 81–89).

Tishby, N., Pereira, F., & Bialek, W. (1999). The information bottleneck method. *The 37th Annual Allerton Conference on Communication, Control, and Computing.*