Journal of Machine Learning Research 8 (2007) 2405-2441 Submitted 3/07; Published 10/07

The Locally Weighted Bag of Words Framework for Document Representation

Guy Lebanon LEBANON@STAT.PURDUE.EDU
Yi Mao YMAO@ECE.PURDUE.EDU
Joshua Dillon JVDILLON@ECE.PURDUE.EDU
Department of Statistics and
School of Electrical and Computer Engineering
Purdue University - West Lafayette, IN, USA

Editor: Andrew McCallum

Abstract

The popular bag of words assumption represents a document as a histogram of word occurrences. While computationally efficient, such a representation is unable to maintain any sequential information. We present an effective sequential document representation that goes beyond the bag of words representation and its *n*-gram extensions. This representation uses local smoothing to embed documents as smooth curves in the multinomial simplex thereby preserving valuable sequential information. In contrast to bag of words or *n*-grams, the new representation is able to robustly capture medium and long range sequential trends in the document. We discuss the representation and its geometric properties and demonstrate its applicability for various text processing tasks.

Keywords: text processing, local smoothing

1. Introduction

Modeling text documents is an essential component in a wide variety of text processing applications, including the classification, segmentation, visualization and retrieval of text. A crucial part of the modeling process is choosing an appropriate representation for documents. In this paper we demonstrate a new representation that considers documents as smooth curves in the multinomial simplex. The new representation goes beyond standard alternatives such as the bag of words and *n*-grams and captures sequential content at a certain resolution determined by a given local smoothing operator.

We consider documents as finite sequences of words

$$y = \langle y_1, \ldots, y_N \rangle \qquad y_i \in V \tag{1}$$

where V represents finite vocabulary which for simplicity is assumed to be a set of integers $V = \{1, \ldots, |V|\} = \{1, \ldots, V\}$. The slight abuse of notation of using V once as a set and once as an integer will not cause confusion later on and serves to simplify the notation. Due to the categorical or nominal nature of V, a document should be considered as a categorical valued time series. In typical cases, we have $1 < N \ll V$ which precludes using standard tools from categorical time series analysis. Instead, the standard approach in text processing is to "vectorize" the data by keeping track of occurrences of length-*n* word-patterns irrespective of where they appear in the document. This approach, called the *n*-gram representation, has the benefit of embedding sequential documents in a Euclidean space $\mathbb{R}^{V^n}$ which is a convenient representation, albeit high dimensional, for many machine learning and statistical models. The specific case of $n = 1$, also called bag of words or bow representation, is perhaps the most frequent document representation due to its relative robustness in sparse situations.

Formally, the n-gram approach represents a document $y = \langle y_1, \ldots, y_N \rangle, y_i \in V$ as $x \in \mathbb{R}^{V^n}$, defined by

$$x_{(j_1,\ldots,j_n)} = \frac{1}{N-n+1} \sum_{i=1}^{N-n+1} \delta_{y_i,j_1} \delta_{y_{i+1},j_2} \cdots \delta_{y_{i+n-1},j_n}, \tag{2}$$

where $\delta_{a,b} = 1$ if $a = b$ and 0 otherwise. In the case of 1-gram or bag of words the above representation reduces to

$$x_j = \frac{1}{N} \sum_{i=1}^{N} \delta_{y_i,j}$$

which is simply the relative frequencies of different vocabulary words in the document.

A slightly more general outlook is to consider smoothed versions of (2) in order to avoid the otherwise overwhelmingly sparse frequency vector (since $N \ll V$, only a small subset of the vocabulary appears in any typical document). For example, a smoothed 1-gram representation is

$$x_j = \frac{1}{Z} \sum_{i=1}^{N} (\delta_{y_i,j} + c), \quad c \geq 0 \tag{3}$$

where Z is a constant that ensures normalization $\sum x_j = 1$. The smoothed representation (3) has a Bayesian interpretation as a the maximum posterior estimate for a multinomial model with Dirichlet prior and setting $c = 0$ in (3) reduces it to the standard word histogram or 1-gram. Recent comparative studies of various n-gram smoothing methods in the contexts of language modeling and information retrieval may be found in Chen and Rosenfeld (2000) and Zhai and Lafferty (2001).

Conceptually, we may consider the n-gram representation for $n = N$ in which case the full original sequential information is maintained. In practice, however, n is typically chosen to be much smaller than N, often taking the values 1, 2, or 3. In these cases, frequently occurring word patterns are kept allowing some limited amount of word-sense disambiguation. On the other hand, almost all of the sequential content, including medium and long range sequential trends and position information is lost.

The paper's main contribution is a new sequential representation called locally weighted bag of words or lowbow. This representation, first introduced in Lebanon (2005), generalizes bag of words by considering the collection of local word histograms throughout the document. In contrast to n-grams, which keep track of frequently occurring patterns independent of their positions, lowbow keeps track of changes in the word histogram as it sweeps through the document from beginning to end. The collection of word histograms is equivalent to a smooth curve which facilitates the differential analysis of the document's sequential content. The use of bag of words rather than n-grams with $n > 1$ is made here for simplicity purpose only. The entire lowbow framework may be generalized to define locally weighted n-grams in a straightforward manner.

The next section presents a detailed explanation of the locally weighted bag of words framework. Section 3 describes the mechanics of using the lowbow framework in document modeling. Section 4 discusses the tradeoff in choosing the amount of temporal smoothing through a bias-variance analysis and generalization error bounds. Section 5 outlines several experiments, followed by related work and discussion. Since our presentation makes frequent use of the geometry of the multinomial simplex, which is not common knowledge in the machine learning community, we provide a brief summary of it in Appendix A.

2. Locally Weighted Bag of Words

As mentioned previously, the original word sequence (1) is categorical, high dimensional and sparse. The smoothing method employed by the bag of words representation (3) is categorical in essence rather than temporal since no time information is preserved. In contrast to (3) or its variants, temporal smoothing such as the one used in local regression or kernel density estimation (e.g., Wand and Jones, 1995) is performed across a continuous temporal or spatial dimension. Temporal smoothing has far greater potential than categorical smoothing since a word can be smoothed out to varying degrees depending on the temporal difference between the two document positions. The main idea behind the locally weighted bag of words framework is to use a local smoothing kernel to smooth the original word sequence temporally. In other words, we borrow the presence of a word at a certain location in the document to a neighboring location but discount its contribution depending on the temporal distance between the two locations.

Since temporal smoothing of words results in several words occupying one location we need to consider the following broader definition of a document.

Definition 1 *A document x of length N is a function $x : \{1,\dots,N\} \times V \to [0,1]$ such that*

$$\sum_{j\in V} x(i,j) = 1 \quad \forall i \in \{1,\dots,N\}.$$

The set of documents (of all lengths) is denoted by $\mathfrak{X}$.

For a document $x \in \mathfrak{X}$ the value $x(i,j)$ represent the weight of the word $j \in V$ at location i. Since the weights sum to one at any location we can consider Definition 1 as providing a local word histogram or distribution associated with each document position. The standard way to represent a word sequence as a document in $\mathfrak{X}$ is to have each location host the appropriate single word with constant weight, which corresponds to the δ_c representation defined below with $c = 0$.

Definition 2 *The standard representation $\delta_c(y) \in \mathfrak{X}$, where $c \geq 0$, of a word sequence $y = \langle y_1,\dots,y_N\rangle$ is*

$$\delta_c(y)(i,j) = \begin{cases} \frac{c}{1+c|V|} & y_i \neq j \\ \frac{1+c}{1+c|V|} & y_i = j \end{cases}. \tag{4}$$

Equation (4) is consistent with Definition 1 since $\sum_{j\in V} \delta_c(y)(i,j) = \frac{1+c|V|}{1+c|V|} = 1$. The parameter c in the above definition injects categorical smoothing as in (3) to avoid zero counts in the δ_c representation.

The standard representation δ_c assumes that each word in the sequence $y = \langle y_1,\dots,y_N\rangle$ occupies a single temporal location $1,\dots,N$. In general, however, Definition 1 lets several words occupy the same location by smoothing the influence of words y_j across different document positions. Doing so is central in converting the discrete-time standard representation to a continuous representation that is much more convenient for modeling and analysis.

Definition 1 is problematic since according to it, two documents of different lengths are considered as fundamentally different objects. It is not clear, for example, how to compare two documents $x_1 : \{1,\dots,N_1\} \times V \to [0,1]$, $x_2 : \{1,\dots,N_2\} \times V \to [0,1]$ of varying lengths $N_1 \neq N_2$. To allow a unified treatment and comparison of documents of arbitrary lengths we map the set $\{1,\dots,N\}$ to a continuous canonical interval, which we arbitrarily choose to be $[0,1]$.

Definition 3 *A length-normalized document x is a function $x : [0,1] \times V \to [0,1]$ such that*

$$\sum_{j \in V} x(t,j) = 1, \qquad \forall t \in [0,1].$$

The set of length-normalized documents is denoted $\mathfrak{X}'$.

A simple way of converting a document $x \in \mathfrak{X}$ to a length-normalized document $x' \in \mathfrak{X}'$ is expressed by the length-normalization function defined below.

Definition 4 *The length-normalization of a document $x \in \mathfrak{X}$ of length N is the mapping*

$$\varphi : \mathfrak{X} \to \mathfrak{X}' \quad \varphi(x)(t,j) = x(\lceil tN \rceil, j)$$

where $\lceil r \rceil$ is the smallest integer greater than or equal to r.

The length-normalization process abstracts away from the actual document length and focuses on the sequential variations within the document relative to its length. In other words, we treat two documents with similar sequential contents but different lengths in a similar fashion. For example the two documents $\langle y_1, y_2, \dots, y_N \rangle$ and $\langle y_1, y_1, y_2, y_2, \dots, y_N, y_N \rangle$ or the more realistic example of a news story and its summary would be mapped to the same length-normalized representation. The assumption that the actual length does not matter and sequential trends should be considered relative to the total length may not hold in some cases. We comment on this assumption further and on how to relax it in Section 7.

We formally define bag of words as the integral of length-normalized documents with respect to time. As we show later, this definition is equivalent to the popular definition of bag of words expressed in Equation (3).

Definition 5 *The bag of words or bow representation of a document y is $\rho(\varphi(\delta_c(y)))$ defined by*

$$\rho : \mathfrak{X}' \to \mathbb{P}_{V-1} \quad \textit{where} \quad [\rho(x)]_j = \int_0^1 x(t,j)\,dt, \tag{5}$$

and $[\cdot]_j$ denotes the j-th component of a vector.

Above, $\mathbb{P}_{V-1}$ stands for the multinomial simplex

$$\mathbb{P}_{V-1} = \left\{ \theta \in \mathbb{R}^V : \forall i\ \theta_i \geq 0, \sum_{j=1}^{V} \theta_j = 1 \right\}$$

which is the subset of $\mathbb{R}^V$ representing the set of all distributions on V events. The subscript $V-1$ is used in $\mathbb{P}_{V-1}$ rather than V in order to reflect its intrinsic dimensionality. The simplex and its Fisher or information geometry are a central part of this paper. Appendix A contains a brief overview of

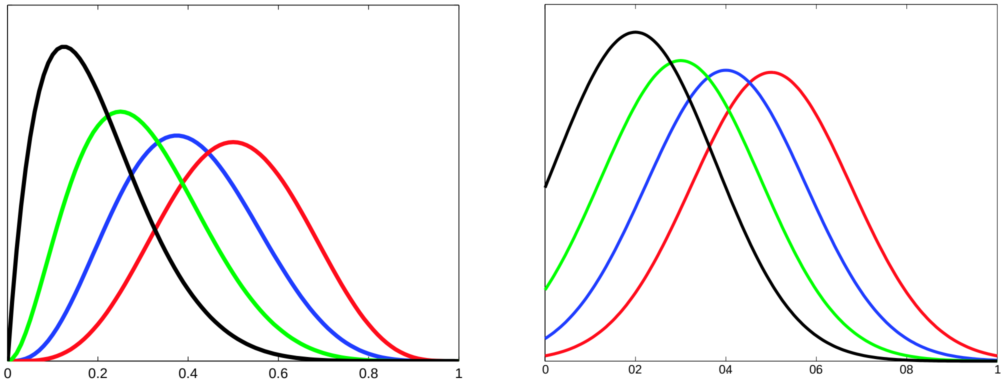


Figure 1: Beta (left) and bounded Gaussian (right) smoothing kernels for $\mu = 0.2, 0.3, 0.4, 0.5$.

the necessary background and further details may be found in Kass and Voss (1997), Amari and Nagaoka (2000) and Lebanon (2005). Note that the function ρ in Definition 5 is well defined since

$$\sum_{j\in V}[\rho(x)]_j = \sum_{j\in V}\int_0^1 x(t,j)\,dt = \int_0^1 \sum_{j\in V} x(t,j)\,dt = \int_0^1 1\,dt = 1 \quad \Longrightarrow \quad \rho(x)\in\mathbb{P}_{V-1}.$$

A local alternative to the bag of words is obtained by integrating a length-normalized document with respect to a non-uniform measure on $[0,1]$. In particular, integrating with respect to a measure that is concentrated around a particular location $\mu\in[0,1]$ provides a smoothed characterization of the local word histogram. In accordance with the statistical literature of non-parametric smoothing we refer to such a measure as a smoothing kernel. Formally, we define it as a function $K_{\mu,\sigma}:[0,1]\to\mathbb{R}$ parameterized by a location parameter $\mu\in[0,1]$ and a scale parameter $\sigma\in(0,\infty)$. The parameter μ represents the (length-normalized) document location at which the measure is concentrated and σ represents its spread or amount of smoothing. We further assume that $K_{\mu,\sigma}$ is smooth in t,μ and is normalized, that is, $\int_0^1 K_{\mu,\sigma}(t)\,dt = 1$.

One example of a smoothing kernel on $[0,1]$ is the Gaussian pdf restricted to $[0,1]$ and renormalized

$$K_{\mu,\sigma}(x) = \begin{cases} \frac{N(x;\mu,\sigma)}{\Phi((1-\mu)/\sigma)-\Phi(-\mu/\sigma)} & x\in[0,1] \\ 0 & x\notin[0,1] \end{cases} \tag{6}$$

where $N(x;\mu,\sigma)$ is the Gaussian pdf with mean μ and variance σ^2 and Φ is the cdf of $N(x;0,1)$. Another example is the beta distribution pdf

$$K_{\mu,\sigma}(x) = \text{Beta}\left(x\,;\,\beta\frac{\mu}{\sigma}\,,\,\beta\frac{1-\mu}{\sigma}\right) \tag{7}$$

where β is selected so that the two parameters of the beta distribution will be greater than 1. The above beta pdf has expectation μ and variance that is increasing in the scale parameter σ. The bounded Gaussian and beta kernels are illustrated in Figure 1.

Definition 6 *The locally weighted bag of words or lowbow representation of the word sequence y is* $\gamma(y) = \{\gamma_\mu(y) : \mu \in [0,1]\}$ *where* $\gamma_\mu(y) \in \mathbb{P}_{V-1}$ *is the local word histogram at μ defined by*

$$[\gamma_\mu(y)]_j = \int_0^1 \varphi(\delta_c(y))(t,j)\, K_{\mu,\sigma}(t) dt. \tag{8}$$

Equation (8) indeed associates a document location with a local histogram or a point in the simplex $\mathbb{P}_{V-1}$ since

$$\begin{aligned}\sum_{j\in V}[\gamma_\mu(y)]_j &= \sum_{j\in V}\int_0^1 \varphi(\delta_c(y))(t,j)K_{\mu,\sigma}(t)\,dt = \int_0^1 K_{\mu,\sigma}(t)\sum_{j\in V}\varphi(\delta_c(y))(t,j)\,dt\\ &= \int_0^1 K_{\mu,\sigma}(t)\cdot 1\,dt = 1.\end{aligned}$$

Geometrically, the lowbow representation of documents is equivalent to parameterized curves in the simplex. The following theorem establishes the continuity and smoothness of these curves which enables the use of differential geometry in the analysis of the lowbow representation and its properties.

Theorem 1 *The lowbow representation is a continuous and differentiable parameterized curve in the simplex, in both the Euclidean and the Fisher geometry.*

Proof We prove below only the continuity of the lowbow representation. The proof of differentiability proceeds along similar lines. Fixing y, the mapping $\mu \mapsto \gamma_\mu(y)$ maps $[0,1]$ into the simplex $\mathbb{P}_{V-1}$. Since $K_{\mu,\sigma}(t)$ is continuous on a compact region $(\mu,t) \in [0,1]^2$, it is also uniformly continuous and we have

$$\begin{aligned}\lim_{\varepsilon\to 0}|[\gamma_\mu(y)]_j - [\gamma_{\mu+\varepsilon}(y)]_j| &= \lim_{\varepsilon\to 0}\left|\int_0^1 \varphi(\delta_c(y))(t,j)K_{\mu,\sigma}(t) - \varphi(\delta_c(y))(t,j)K_{\mu+\varepsilon,\sigma}(t)dt\right|\\ &\le \lim_{\varepsilon\to 0}\int_0^1 \varphi(\delta_c(y))(t,j)|K_{\mu,\sigma}(t) - K_{\mu+\varepsilon,\sigma}(t)|\,dt\\ &\le \lim_{\varepsilon\to 0}\sup_{t\in[0,1]}|K_{\mu,\sigma}(t) - K_{\mu+\varepsilon,\sigma}(t)|\int_0^1 \varphi(\delta_c(y))(t,j)\,dt\\ &= \lim_{\varepsilon\to 0}\sup_{t\in[0,1]}|K_{\mu,\sigma}(t) - K_{\mu+\varepsilon,\sigma}(t)| = 0.\end{aligned}$$

As a result,

$$\lim_{\varepsilon\to 0}\|\gamma_\mu(y) - \gamma_{\mu+\varepsilon}(y)\|_2 = \sqrt{\sum_{j\in V}|[\gamma_\mu(y)]_j - [\gamma_{\mu+\varepsilon}(y)]_j|^2} \to 0$$

proving the continuity of $\gamma_\mu(y)$ in the Euclidean geometry. Continuity in the Fisher geometry follows since it shares the same topology as the Euclidean geometry. ■

It is important to note that the parameterized curve that corresponds to the lowbow representation consists of two parts: the geometric figure $\{\gamma_\mu(y) : \mu \in [0,1]\} \subset \mathbb{P}_{V-1}$ and the parameterization function $\mu \mapsto \gamma_\mu(y)$ that ties the local histogram to a location μ in the normalized document. While it is easy to ignore the parameterization function when dealing with parameterized curves, one must be aware that different lowbow representations may share similar geometric figures but possess different parameterization speeds. Thus it is important to keep track of the parameterization speed as well as the geometric figure.

The geometric properties of the curve depend on the word sequence, the kernel shape and the kernel scale parameter. The kernel scale parameter is especially important as it determines the amount of temporal smoothing employed. As the following theorem shows, if $\sigma \to \infty$ the lowbow curve degenerates into a single point corresponding to the bow representation. As a consequence we view the popular bag of words representation (3) as a special case of the lowbow representation.

Theorem 2 *Let $K_{\mu,\sigma}$ be a smoothing kernel such that when $\sigma \to \infty$, $K_{\mu,\sigma}(x)$ is constant in μ, x. Then for $\sigma \to \infty$, the lowbow curve $\gamma(y)$ degenerates into a single point corresponding to the bow representation of* (3).

Proof Since the kernel is both constant and normalized over $[0,1]$, we have $K_{\mu,\sigma}(t) = 1$ for all $\mu, t \in [0,1]$. For all $\mu \in [0,1]$,

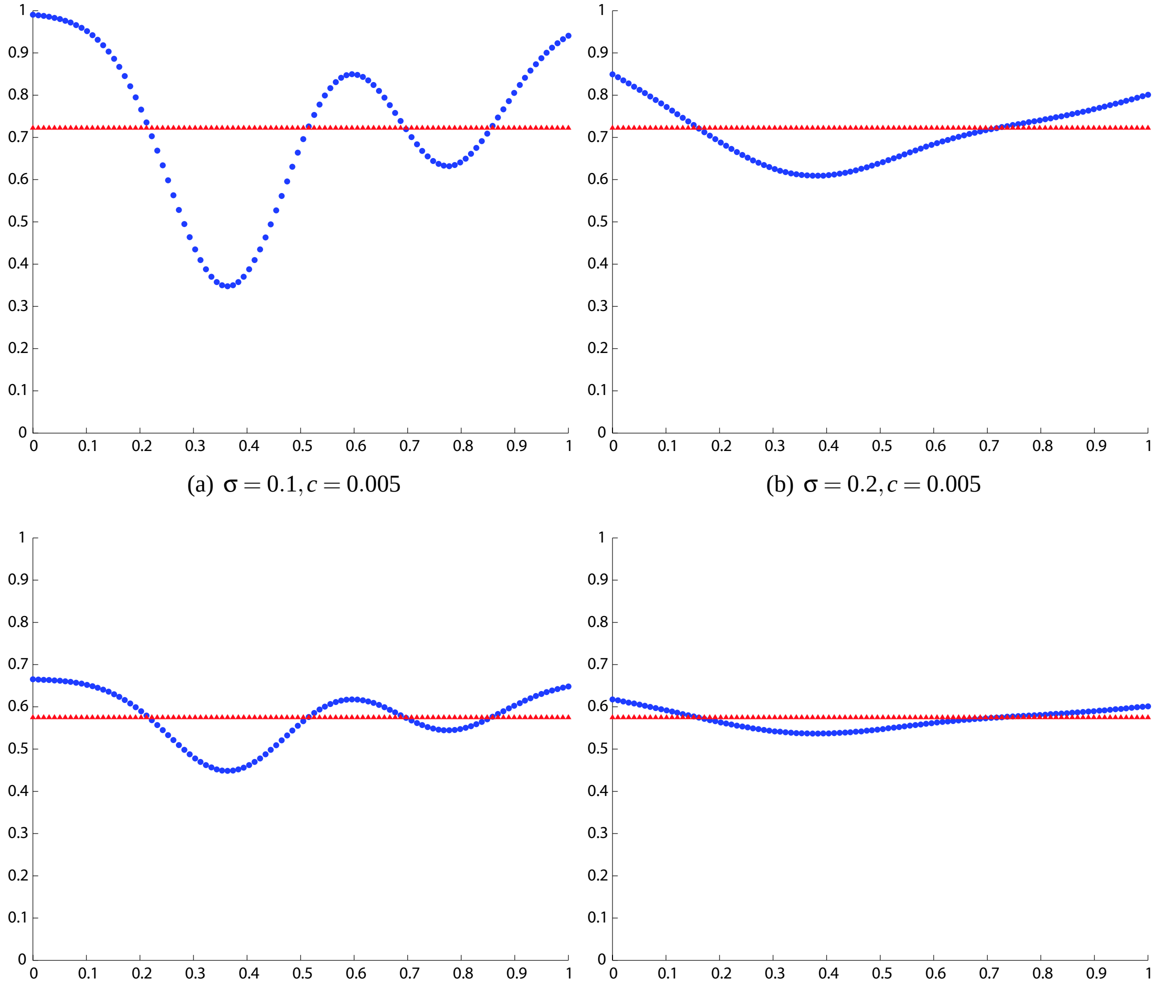
$$\begin{aligned}
[\gamma_\mu(y)]_j &= \int_0^1 \varphi(\delta_c(y))(t,j) K_{\mu,\sigma}(t)\, dt = \int_0^1 \varphi(\delta_c(y))(t,j)\, dt \\
&= \sum_{i=1}^N \frac{1}{N} \left(\delta_{y_i,j} \frac{1+c}{1+c|V|} + (1-\delta_{y_i,j}) \frac{c}{1+c|V|} \right) \\
&\propto \sum_{i=1}^N \delta_{y_i,j}(1+c) + (1-\delta_{y_i,j})c \propto \sum_{i=1}^N (\delta_{y_i,j} + c).
\end{aligned}$$

■

Intuitively, small σ will result in a simplicial curve that quickly moves between the different corners of the simplex as the words $y_1, y_2, \ldots, y_N$ are encountered. The extreme case of $\sigma \to 0$ represents a discontinuous curve equivalent to the original word sequence representation (1). It is unlikely that either of the extreme cases $\sigma \to \infty$ or $\sigma \to 0$ will be an optimal choice from a modeling perspective. By varying σ between 0 and ∞, the lowbow representation interpolates between these two extreme cases and captures sequential detail at different resolutions. Selecting an appropriate scale $0 < \sigma < \infty$ we obtain a sequential resolution that captures sequential trends at a certain resolution while smoothing away finer temporal details.

Figures 2-3 illustrate the curve resulting from the lowbow representation and its dependency on the kernel scale parameter and the smoothing coefficient. Notice how the curve shrinks as σ increases until it reaches the single point that is the bow model. Increasing c, on the other hand, pushes the geometric figure towards the center of the simplex.

It is useful to have a quantitative characterization of the complexity of the lowbow representation as a function of the chosen kernel and σ. To this end, the kernel's complexity, defined below, serves as a bound for variations in the lowbow curve.



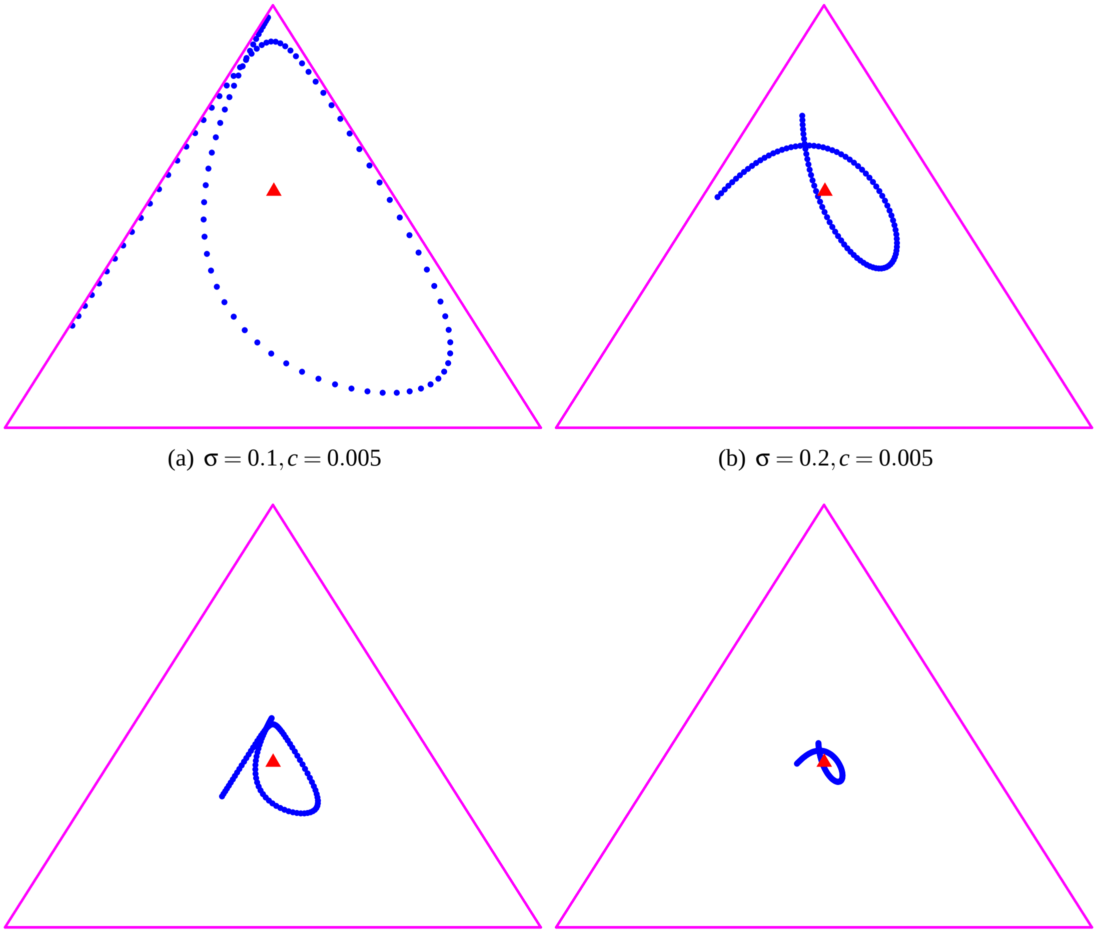
(a) $\sigma = 0.1, c = 0.005$

(b) $\sigma = 0.2, c = 0.005$

(c) $\sigma = 0.1, c = 1$

(d) $\sigma = 0.2, c = 1$

Figure 2: The curve in $\mathbb{P}_1$ resulting from the lowbow representation of the word sequence $\langle$`1 1 1 2 2 1 1 1 2 1 1`$\rangle$. Since $[\gamma_\mu(y)]_2 = 1 - [\gamma_\mu(y)]_1$ we visualize the curve by graphing $[\gamma_\mu(y)]_1$ as a function of μ. The figures illustrate the differences as the scale parameter of the Gaussian kernel σ increases from 0.1 to 0.2 (left vs. right column) and the smoothing coefficient c varies from 0.005 to 1 (first vs. second row). Increasing the kernel scale causes some local features to vanish, for example the second local minimum. In addition, increasing σ shrinks the figure towards the single bow point (represented by the horizontal line). Increasing the smoothing coefficient c causes the figure to stay away from the boundary of the simplex and concentrate in the center. Since the curves are composed of 100 dots, the distances between the dots indicate the parameterization speed of the curves.



(a) $\sigma = 0.1, c = 0.005$

(b) $\sigma = 0.2, c = 0.005$

(c) $\sigma = 0.1, c = 1$

(d) $\sigma = 0.2, c = 1$

Figure 3: The curve in $\mathbb{P}_2$ resulting from the lowbow representation of the word sequence $\langle$`1 3 3 3 2 2 1 3 3`$\rangle$. In this case $\mathbb{P}_2$ is visualized as a triangle in $\mathbb{R}^2$ (see Figure 15 for visualizing $\mathbb{P}_2$). The figures illustrate the differences as the scale parameter of the Gaussian kernel σ increases from 0.1 to 0.2 (left vs. right column) and the smoothing coefficient c varies from 0.005 to 1 (first vs. second row). Increasing the kernel scale causes some local features to vanish, for example the tail in the bottom left corner of $\mathbb{P}_2$. In addition, increasing σ shrinks the figure towards the single bow point (represented by the triangle). Increasing the smoothing coefficient c causes the figure to stay away from the boundary of the simplex and concentrate in the center. Since the curves are composed of 100 dots, the distances between the dots indicate the parameterization speed of the curves.

Definition 7 *Let $K_{\mu,\sigma}(t)$ be a kernel that is Lipschitz continuous*[1] *in μ with a Lipschitz constant $C_K(t)$. The kernel's complexity is defined as*

$$O(K) = \sqrt{V}\int_0^1 C_K(t)\,dt.$$

The theorem below proves that the lowbow curve is Lipschitz continuous with a Lipschitz constant $O(K)$, thus connecting the curve complexity with the shape and the scale of the kernel.

Theorem 3 *The lowbow curve $\gamma(y)$ satisfies*

$$\|\gamma_\mu(y)-\gamma_\tau(y)\|_2 \;\leq\; |\mu-\tau|\;O(K), \quad \forall \mu,\tau\in[0,1].$$

Proof

$$\begin{aligned}
|[\gamma_\mu(y)]_j-[\gamma_\tau(y)]_j| &\leq \int_0^1 \varphi(\delta_c(y))(t,j)|K_{\mu,\sigma}(t)-K_{\tau,\sigma}(t)|\,dt\\
&\leq \int_0^1 |K_{\mu,\sigma}(t)-K_{\tau,\sigma}(t)|\,dt\\
&\leq |\mu-\tau|\int_0^1 C_K(t)\,dt
\end{aligned}$$

and so $\|\gamma_\mu(y)-\gamma_\tau(y)\|_2 = \sqrt{\sum_{j\in V}|[\gamma_\mu(y)]_j-[\gamma_\tau(y)]_j|^2} \leq |\mu-\tau|O(K)$. ■

3. Modeling of Simplicial Curves

Modeling functional data such as lowbow curves is known in the statistics literature as functional data analysis (e.g., Ramsay and Dalzell, 1991; Ramsay and Silverman, 2005). Previous work in this area focused on low dimensional functional data such as one dimensional or two dimensional curves. In this section we discuss some issues concerning generative and conditional modeling of lowbow curves. Additional information regarding the practical use of lowbow curves in a number of text processing tasks may be found in Section 5.

Geometrically, a lowbow curve is a point in an infinite product of simplices $\mathbb{P}_{V-1}^{[0,1]}$ that is naturally equipped with the product topology and geometry of the individual simplices. In practice, maintaining a continuous representation is often difficult and unnecessary. Sampling the path at representative points $\mu_1,\ldots,\mu_l\in[0,1]$ provides a finite dimensional lowbow representation equivalent to a point in the product space $\mathbb{P}_{V-1}^{l}$. Thus, even though we proceed below to consider continuous curves and infinite dimensional spaces $\mathbb{P}_{V-1}^{[0,1]}$, in practice we will typically discretize the curves and replace integrals with appropriate summations.

Given a Riemannian metric g on the simplex, its product form

$$g'_\theta(u,v) = \int_0^1 g_{\theta(t)}(u(t),v(t))\,dt$$

defines a corresponding metric on lowbow curves. As a result, geometric structures compatible with the base metric g, such as distance or curvature, give rise to analogous product versions. For

[1] A Lipschitz continuous function f satisfies $|f(x)-f(y)|\leq C|x-y|$ for some constant C called the Lipschitz constant.

example, the distance between lowbow representations of two word sequences $\gamma(y), \gamma(z) \in \mathbb{P}_m^{[0,1]}$ is the average distance between the corresponding time coordinates

$$d(\gamma(y), \gamma(z)) = \int_0^1 d(\gamma_\mu(y), \gamma_\mu(z))\, d\mu \tag{9}$$

where $d(\gamma_\mu(y), \gamma_\mu(z))$ depends on the simplex geometry under consideration, e.g. Equation (21) in the case of the Fisher geometry or $d(\gamma_\mu(y), \gamma_\mu(z)) = \|\gamma_\mu(y) - \gamma_\mu(z)\|_2$ in the case of Euclidean geometry.

Using the integrated distance formula (9) we can easily adapt distance-based algorithms to the lowbow representation. For example, k-nearest neighbor classifiers are adapted by replacing standard distances such as the Euclidean distance or cosine similarity with the integrated distance (9) or its discretized version.

In contrast to the base distance on $\mathbb{P}_{V-1}$ which is used in the bow representation, the integrated distance (9) captures local differences in text sequences. For example, it compares the beginning of document y with the beginning of document z, the middle with the middle, and the end with the end. While it may be argued that the above is not expected to always accurately model differences between documents, it does hold in some cases. For example, news articles have a natural semantic progression starting with a brief summary at the beginning and delving into more detail later on, often in a chronological manner. Similarly, other documents such as web pages and emails share a similar sequential structure. Section 5.3 provides some experimental support for this line of thought and also describes some alternatives.

In a similar way, we can also apply kernel-based algorithms such as SVM to documents using the lowbow representation by considering a kernel over $\mathbb{P}_{V-1}^{[0,1]}$. For example, the product geometry may be used to define a product diffusion process whose kernel can conveniently capture local relationships between documents. Assuming a base Fisher geometry we obtain the approximated diffusion kernel

$$K_t(\gamma(y), \gamma(z)) \propto \exp\left(-\frac{1}{t}\left(\int_0^1 \arccos\left(\sum_{j\in V}\sqrt{[\gamma_\mu(y)]_j[\gamma_\mu(z)]_j}\right) d\mu\right)^2\right) \tag{10}$$

using the parametrix expansion described in Berger et al. (1971). We omit the details as they are closely related to the derivations of Lafferty and Lebanon (2005). Alternative kernels can be obtained using the mechanism of Hilbertian metrics developed by Christensen et al. (1984) and Hein and Bousquet (2005).

The Fisher diffusion kernel of Lafferty and Lebanon (2005) achieves excellent performance in standard text classification experiments. We show in Section 5 that its lowbow version (10) further improves upon those results. In addition to classification, the lowbow diffusion kernel may prove useful for other tasks such as dimensionality reduction using kernel PCA, support vector regression, and semi-supervised learning.

The lowbow representation may also be used to construct generative models for text that generalize the naive Bayes or multinomial model. By estimating the probability $p(y)$ associated with a given text sequence y, such models serve an important role in machine translation, speech recognition and information retrieval. In contrast to the multinomial model which ignores the sequential progression in a document, lowbow curves γ may be considered as a semiparametric generative model assigning the probability vector γ_μ to the generation of words around the document location μN. Formally this amounts to the following process:

Step 1 Draw a document length N from some distribution on positive integers.

Step 2 Generate the words $y_1,\dots,y_N$ according to

$$y_i \sim \text{Mult}(\theta_{i1},\dots,\theta_{iV}) \quad \text{where} \quad \theta_{ij} \propto \int_{(i-1)/N}^{i/N} [\gamma_\mu]_j \, d\mu.$$

The above model can also be used to describe situations in which the underlying document distribution changes with time (e.g., Forman, 2006). Lebanon and Zhao (2007) describe a local likelihood model that is essentially equivalent to the generative lowbow model described above. In contrast to the model of Blei and Lafferty (2006) the lowbow generative model is not based on latent topics and is inherently smooth.

The differential characteristics of the lowbow curve convey significant information and deserve closer inspection. As pointed out by Ramsay and Silverman (2005), applying linear differential operators L_α to functional data $L_\alpha f = \sum_i \alpha_i D^i f$ (where D^i is the i-th derivative operator) often reveals interesting features and properties that are normally difficult to detect. The simplest such operator is the first derivative or velocity $D\gamma_\mu = \dot{\gamma}_\mu$ (defined by $[\dot{\gamma}_\mu]_j = \mathrm{d}[\gamma_\mu]_j/\mathrm{d}\mu$) which reveals the instantaneous direction of the curve at a certain time point as well as the current speed through its norm $\|\dot{\gamma}_\mu\|$. More specifically, we can obtain a tangent vector field $\dot{\gamma}$ along the curve that describes sequential topic trends and their change. Higher order differential operators such as the curvature reveal the amount of curve variation or deviation from a straight line. Integrating the norm of the curvature tensor over $\mu \in [0,1]$ provides a measure of the sequential topic complexity or variability throughout the document. We demonstrate such differential operators and their use in visualization and segmentation of documents in Section 5. Further details concerning differential operators and their role in visualizing lowbow curves may be found in Mao et al. (2007).

In general, it is fair to say that modeling curves is more complicated than modeling points. However, if done correctly it has the potential to capture information that otherwise would remain undetected. Keeping in mind that we can control the amount of variability by changing σ, thereby interpolating between $\langle y_1,\dots,y_N\rangle$ and (3), we are able to effectively model sequential trends in documents. The choice of σ controls the amount of smoothing and as in non-parametric density estimation, an appropriate choice is crucial to the success of the model. This notion is explored in greater detail in the next section.

4. Kernel Smoothing, Bias-Variance Tradeoff and Generalization Error Bounds

The choice of the kernel scale parameter σ or the amount of smoothing is essential for the success of the lowbow framework. Choosing a σ that is too large would result in a function class that is relatively weak and will not be able to express the desired sequential content. Choosing a σ that is too small would result in a rich function class that is destined to overfit the available training set. This central tradeoff has been analyzed in statistics through the bias and variance of the estimated parameters and in computational learning theory through generalization error bounds. In this section, we discuss this tradeoff from both viewpoints. Further details concerning the statistical properties of the lowbow estimator as a local likelihood model for streaming data may be found in Lebanon and Zhao (2007). Practical aspects concerning the selection of σ appear in Section 5.

4.1 Bias and Variance Tradeoff

We discuss the bias and variance of the lowbow model $\gamma(y)$ as an estimator for an underlying semi-parametric model $\{\theta_t : t \in [0,1]\} \subset \mathbb{P}_{V-1}$ which we assume generated the observed document y. The model assigns a local multinomial θ_t to different locations t and proceeds to generate the words $y_i, i = 1,\ldots,N$ according to $y_i \sim_{iid} \theta_{i/N}$. Note that the iid sampling assumption simply implies that the sampling of the words from their respective multinomials are independent. It does not prevent the assumption of a higher order structure, Markovian or otherwise, on the relationship between the multinomials generating adjacent words $\theta_{i/N}, \theta_{(i+1)/N}$.

The bias and variance of the the lowbow estimator $\gamma(y) = \hat{\theta}(y)$, reveal the expected tradeoff by considering their dependence on the kernel scale σ. We start by writing the components of $\gamma(y) = \hat{\theta}$ as a weighted combination of the sampled words

$$\hat{\theta}_{\mu j} = \int_0^1 y(t,j) K_{\mu,\sigma}(t)\, dt = \sum_{i=1}^{N} y(i,j) \int_{(i-1)/N}^{i/N} K_{\mu,\sigma}(t)\, dt = \sum_{\tau \in J} w_{\mu-\tau} y(\mu-\tau, j)$$

where $y \in \mathfrak{X}'$, $w_i = \int_{(i-1)/N}^{i/N} K_{\mu,\sigma}(t)\, dt$ and $J = \{\mu - N, \ldots, \mu - 1\}$. It is relatively simple to show that $\hat{\theta}_{\mu j}$ is a consistent estimator of $\theta_{\mu j}$ under conditions that ensure the weight function w approaches a delta function at μ as the number of samples goes to infinity (e.g., Wand and Jones, 1995). In our case, the number of samples is fixed and is dictated by the number of words in the document. However, despite the lack of an asymptotic trend $N \to \infty$ we can still gain insight from analyzing the dependency of the bias and variance of the lowbow estimator as a function of the kernel scale parameter σ.

Using standard results concerning the expectation and variance of Bernoulli random variables we have

$$\begin{aligned}
\mathsf{bias}\,(\hat{\theta}_{\mu j}) &= \mathsf{E}\,(\hat{\theta}_{\mu j} - \theta_{\mu j}) = \sum_{\tau \in J} w_{\mu-\tau} \mathsf{E}\,(y(\mu-\tau, j)) - \theta_{\mu j} \\
&= \sum_{\tau \in J} w_{\mu-\tau} (\theta_{\mu-\tau,j} - \theta_{\mu j}). \qquad (11) \\
\mathsf{Var}\,(\hat{\theta}_{\mu j}) &= \mathsf{E}\,(\hat{\theta}_{\mu j} - \mathsf{E}\,\hat{\theta}_{\mu j})^2 = \mathsf{E}\,\left(\sum_{\tau \in J} w_{\mu-\tau} (y(\mu-\tau, j) - \theta_{\mu-\tau,j}) \right)^2 \\
&= \sum_{\tau \in J} \sum_{\tau' \in J} w_{\mu-\tau} w_{\mu-\tau'} \mathsf{E}\,(y(\mu-\tau, j) - \theta_{\mu-\tau})(y(\mu-\tau', j) - \theta_{\mu-\tau',j}) \\
&= \sum_{\tau \in J} w_{\mu-\tau}^2 \mathsf{Var}\,(y(\mu-\tau, j)) \\
&= \sum_{\tau \in J} w_{\mu-\tau}^2 \theta_{\mu-\tau,j} (1 - \theta_{\mu-\tau,j}). \qquad (12)
\end{aligned}$$

The bias term clearly depends on the weight vector w and on the rate of local changes in the true parameter $\theta_{\mu j}$. For a certain fixed model $\hat{\theta}_{\mu j}$, the bias clearly decreases as the weight distribution approaches a delta function at μ, that is, $w_i = 1$ if $i = \mu$ and 0 otherwise. In fact, in the limiting case of $w_i = \delta_{i\mu}$, $\mathsf{bias}\,(\hat{\theta}_{\mu j}) = 0$ and $\mathsf{Var}\,(\hat{\theta}_{\mu j}) = \theta_{\mu j}(1 - \theta_{\mu j})$. As the weight distribution becomes less localized, the bias will increase (in the absolute value) and the variance will typically decrease due to the shape of the function $f(w_i) = w_i^2$ for $w_i \in [0,1]$. The precise characterization of the variance

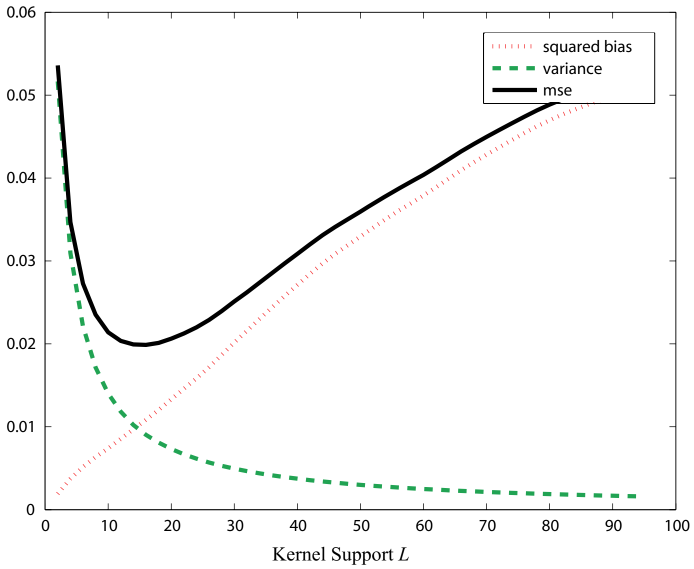


Figure 4: Squared bias, variance and mean squared error of the lowbow estimator $\hat{\theta}_{ij}$ as a function of a triangular kernel support, that is, L in (13). The curve was generated by averaging over synthetic data θ_{ij} drawn from a bounded Wiener process on $[0,1]$.

reduction depends on the model $\theta_{\mu j}$ and the functional form of the kernel. Figure 4 contains an illustration of the squared bias, variance and mean squared error for the discretized triangular kernel

$$w_i = \frac{1}{Z}\left(1 - \frac{2}{L}|i|\right) \qquad i = -L/2, \ldots, L/2 \tag{13}$$

where L defines the kernel support and Z ensures normalization. In the figure, we used synthetic data $\theta_{ij}, i = 1, \ldots, 100$ generated from a bounded Wiener process on $[0,1]$ (i.e., a bounded random walk with Gaussian increments). To avoid phenomena that correspond to a particular sample path we averaged the bias and variance over 200 samples from the process.

The problem of selecting a particular weight vector w or kernel K for the lowbow estimator that minimizes the mean squared error is related to the problem of bandwidth kernel selection in local regression and density estimation. The simple estimate obtained from the plug-in rule for the bias and variance (i.e., $\theta_{\mu j} \mapsto \hat{\theta}_{\mu j}$ in Equations (11)-(12)) is usually not recommended due to the poor estimation performance of plug-in rules (e.g., Cleveland and Loader, 1996). More sophisticated estimates exist, including adaptive estimators that may select different bandwidths or kernels at different points. An alternative approach, which we adopted in our experiments, is to use cross validation or bootstrapping in the selection process.

4.2 Large Deviation Bounds and Covering Numbers

An alternative approach to bias-variance analysis is to characterize the kernel scale tradeoff through the study of generalization error bounds. Such bounds use large deviation techniques to characterize the difference between the empirical risk or training error and the expected risk uniformly over a class of functions $L = \{f_\alpha : \alpha \in I\}$. These bounds are expressed probabilistically and usually take the following form (Anthony and Bartlett, 1999)

$$P\left(\sup_{\alpha \in I} |\mathsf{E}_p(\mathcal{L}(f_\alpha(Z))) - \mathsf{E}_{\tilde{p}}(\mathcal{L}(f_\alpha(Z)))| \geq \varepsilon\right) \leq C(\mathcal{L}, L, n, \varepsilon). \tag{14}$$

Above, Z represents any sequence of n examples - either X in the unsupervised scenario or (X,Y) in the supervised scenario and $\mathsf{E}_p, \mathsf{E}_{\tilde{p}}$ represent the expectation over the sampling distribution and the empirical distribution $\tilde{p}(z) = \frac{1}{n}\sum_{i=1}^{n} \delta_{z,z_i}$. $\mathcal{L}$ represents some loss function, for example classification error rate and the function C measures the rate of uniform convergence of the empirical risk $\mathsf{E}_{\tilde{p}}(\mathcal{L}(f_\alpha(Z)))$ to the true risk $\mathsf{E}_p(\mathcal{L}(f_\alpha(Z)))$ over the function class $L = \{f_\alpha : \alpha \in I\}$.

To obtain a model with a small expected risk we need to balance the following two goals. On the one hand, we need to minimize the empirical risk $\mathsf{E}_{\tilde{p}}(\mathcal{L}(\alpha, Z))$ since the expected risk is typically close to the empirical risk (by the uniform law of large numbers). On the other hand, we need to tighten the bound (14) by selecting a function class L that results in a small value of the function C. This tradeoff, presented by Vapnik (1998) under the name structural risk minimization, is the computational learning theory analog of the statistical bias-variance concept.

A lowbow representation with a small σ would lead to a low empirical risk since it results in a richer and more accurate expression of the data. Increasing σ forms a lossy transformation and hence leads to essential loss of data features and higher training error but would reduce C and therefore also the bound on the expected error.

The most frequent way to bound C is through the use of the covering number which measures the size of a function class (Dudley, 1984; Anthony and Bartlett, 1999). The covering number enables several ways of determining the rate of uniform convergence C in (14), for example see Theorem 1 and 2 in Zhang (2002).

Definition 8 *Let $x = x_1, \ldots, x_n \in \mathcal{X}$ be a set of observations and $f_\alpha : \mathcal{X} \to \mathbb{R}$ be a parameterized function. The covering number in p-norm $\mathcal{N}_p(f, \varepsilon, (x_1, \ldots, x_n))$ is the minimum number m of vectors $v_1, \ldots, v_m \in \mathbb{R}^n$ for which*

$$\forall \alpha\, \exists v_j \quad \textit{such that} \quad \left(\frac{1}{n}\sum_{i=1}^{n} |f_\alpha(x_i) - v_{ji}|^p\right)^{1/p} \leq \varepsilon.$$

In other words, the set $\{f_\alpha(x) : \alpha \in I\} \subset \mathbb{R}^n$ is covered by m ε-balls centered at $v_1, \ldots, v_m$.

Definition 9 *The uniform covering number $\mathcal{N}_p(f, \varepsilon, n)$ is defined as*

$$\mathcal{N}_p(f, \varepsilon, n) = \sup_{x_1, \ldots, x_n} \mathcal{N}_p(f, \varepsilon, x_1, \ldots, x_n).$$

The covering numbers themselves are difficult to compute precisely and are usually bounded themselves. Recent research results that bound the covering numbers for important function classes such as neural networks, support vector machines, boosting and logistic regression may be found in Williamson et al. (2001), Guo et al. (2002) and Zhang (2002). We focus on the covering number bounds in Zhang (2002) for linear classifiers as they are relatively easy to express in terms of the kernel scale parameter. The theorem and bounds below are expressed for continuous lowbow representation and continuous linear classifiers. The same results hold with analogous proofs in the more practical case of finite dimensional linear classifiers and discretized lowbow representations.

Theorem 4 *For the class of continuous linear classifiers $L = \{f_\alpha(\gamma(y)) : \|\alpha\|_2 \le a\}$ operating on continuous lowbow representation*

$$f_\alpha(\gamma(y)) = \sum_{j\in V}\int_0^1 \alpha_j(\mu)[\gamma_\mu(y)]_j\,d\mu$$

we have the following bounds on the L_2 and L_∞ covering numbers

$$\mathcal{N}_2(L,\varepsilon,n) \le 2^{\lceil a^2b^2/\varepsilon^2\rceil \log_2(2n+1)}$$
$$\mathcal{N}_\infty(L,\varepsilon,n) \le 2^{36(a^2b^2/\varepsilon^2)\log_2(2\lceil 4ab/\varepsilon+2\rceil n+1)}$$

where $b = \min(1, |||\mathbf{K}_\sigma|||_2)$.

Above, α represents a vector of weight functions $\alpha = (\alpha_1,\dots,\alpha_V), \alpha_i : [0,1] \to \mathbb{R}$ that parameterize linear operators on $\gamma(y)$. The norm $\|\alpha\|_2$ is defined as $\sqrt{\sum_j \int \alpha_j^2(t)\,dt}$. $\mathbf{K}_\sigma$ is an operator on $f : [0,1] \mapsto [0,1]$ such that $\big(\mathbf{K}_\sigma f\big)(\mu) = \int K_{\mu,\sigma}(t) f(t)\,dt$. The induced 2-norm of the operator is (e.g., Horn and Johnson, 1990)

$$|||\mathbf{K}_\sigma|||_2 = \sup_{\|f\|_2=1} \|\mathbf{K}_\sigma f\|_2 = \sup_{\|f\|_2=1} \sqrt{\int \left(\int K_{\mu,\sigma}(t) f(t)\,dt\right)^2 d\mu} \tag{15}$$

where $\|f\|_2 = \sqrt{\int f^2(t)\,dt}$.

Proof First note that the L_2 norm of the lowbow representation can be bounded by the constant 1

$$\begin{aligned}
\|\gamma(y)\|_2^2 &= \sum_{j\in V}\int_0^1 ([\gamma_\mu(y)]_j)^2\,d\mu = \sum_{j\in V}\int_0^1 \left(\int_0^1 x(t,j) K_{\mu,\sigma}(t)\,dt\right)^2 d\mu \\
&= \sum_{j\in V}\int_0^1 \left(\iint_{[0,1]^2} x(t,j)x(t',j)K_{\mu,\sigma}(t)K_{\mu,\sigma}(t')\,dtdt'\right) d\mu \\
&\le \iiint_{[0,1]^3} \left(\sum_{j\in V} x^2(t,j)\right)^{1/2} \left(\sum_{j\in V} x^2(t',j)\right)^{1/2} K_{\mu,\sigma}(t)K_{\mu,\sigma}(t')\,dtdt'd\mu \\
&\le \iiint_{[0,1]^3} \left(\sum_{j\in V} x(t,j)\right)^{1/2} \left(\sum_{j\in V} x(t',j)\right)^{1/2} K_{\mu,\sigma}(t)K_{\mu,\sigma}(t')\,dtdt'd\mu \\
&= \int_0^1 \left(\int_0^1 K_{\mu,\sigma}(t)\,dt\right)^2 d\mu = 1.
\end{aligned}$$

Alternatively, an occasionally tighter bound that depends on the operator norm and therefore on the kernel's scale parameter is

$$\begin{aligned}
\|\gamma(y)\|_2^2 &= \sum_{j\in V}\int_0^1 ([\gamma_\mu(y)]_j)^2 d\mu = \sum_{j\in V}\int_0^1 \left(\int_0^1 K_{\mu,\sigma}(t)x(t,j)dt\right)^2 d\mu = \sum_{j\in V} \|\mathbf{K}_\sigma x(\cdot,j)\|_2^2 \\
&\le \sum_{j\in V} |||\mathbf{K}_\sigma|||_2^2 \, \|x(\cdot,j)\|_2^2 = |||\mathbf{K}_\sigma|||_2^2 \left(\sum_{j\in V} \|x(\cdot,j)\|_2^2\right) \le |||\mathbf{K}_\sigma|||_2^2 \left(\sum_{j\in V} \|x(\cdot,j)\|_1\right) \\
&= |||\mathbf{K}_\sigma|||_2^2 \left(\int_0^1 \sum_{j\in V} x(t,j)dt\right) = |||\mathbf{K}_\sigma|||_2^2
\end{aligned}$$

where the last two inequalities follow from the definition of the induced operator norm (Horn and Johnson, 1990) and the fact that $\|x(\cdot,j)\|_2^2 = \int_0^1 x(t,j)^2 dt \leq \int_0^1 x(t,j)dt = \|x(\cdot,j)\|_1$.

The proof is concluded by plugging in the above bound into Corollary 3 and Theorem 4 of Zhang (2002):

$$\|x\|_2 \leq b, \|w\|_2 \leq a \quad \Rightarrow \quad \log_2 \mathcal{N}_2(L,\varepsilon,n) \leq \lceil a^2b^2/\varepsilon^2 \rceil \log_2(2n+1), \tag{16}$$

$$\|x\|_2 \leq b, \|w\|_2 \leq a \quad \Rightarrow \quad \log_2 \mathcal{N}_\infty(L,\varepsilon,n) \leq 36\frac{a^2b^2}{\varepsilon^2} \log_2(2\lceil 4ab/\varepsilon+2\rceil n+1). \tag{17}$$

Note that since the bounds in (16)-(17) do not depend on the dimensionality of the data x they hold for any dimensionality, as well as in the limit of continuous data and continuous linear operators as above. ■

The theorem above remains true (and actually it is closer to the original statements in Zhang, 2002) for discretized lowbow representation $\{\gamma_\mu(y) : \mu \in T\}$ where T is a finite set which reduces the lowbow representation and α to a matrix form and discretize the linear operator $\langle \alpha, \gamma(y)\rangle = \sum_{j\in V}\sum_{\mu\in T} \alpha_{j\mu}[\gamma_\mu(y)]_j$. The covering number bounds in Theorem 4 may be directly applied, using either the continuous or the discretized versions, to bound the classification expected error rate for linear classifiers such as support vector machines, Fisher's linear discriminant, boosting, and logistic regression. We do not reproduce these results here since active research in this area frequently improves the precise form of the bound and the constants involved.

As the kernel scale parameter σ decreases, the kernel becomes less uniform thus increasing the possible variability in the data representation $\|\gamma(y)\|_2^2$ and the covering number bound. In the case of the bounded Gaussian kernel (6) we compute $|||\mathbf{K}_\sigma|||_2$ as a function of the kernel scale parameter σ which is illustrated in Figure 5.

5. Experiments

In this section, we demonstrate lowbow's applicability to several text processing tasks, including text classification using nearest neighbor and support vector machines, text segmentation, and document visualization. All experiments use real world data.

5.1 Text Classification using Nearest Neighbor

We start by examining lowbow and its properties in the context of text classification using a nearest neighbor classifier. We report experimental results for the WebKB faculty vs. course task and the Reuters-21578 top ten categories (1 vs. all) using the standard mod-apte training-testing split. In the WebKB task we repeatedly sampled subsets for training and testing with equal positive and negative examples. In the Reuters task we randomly sampled subsets of the mod-apte split for training and testing data which resulted in unbalanced train and test sets containing more negative than positive examples. Sampling training sets of different sizes from the mod-apte split enabled us to examine the behavior of the classifiers as a function of the size of the training set.

The continuous quantities in the lowbow calculation were approximated by a discrete sample of 5 equally spaced points in the interval $[0,1]$ turning the integrals into simple sums. As a result, the computational complexity associated with the lowbow representation is simply the number of sampling points times the complexity of the corresponding bow classifier. Choosing 5 sampling

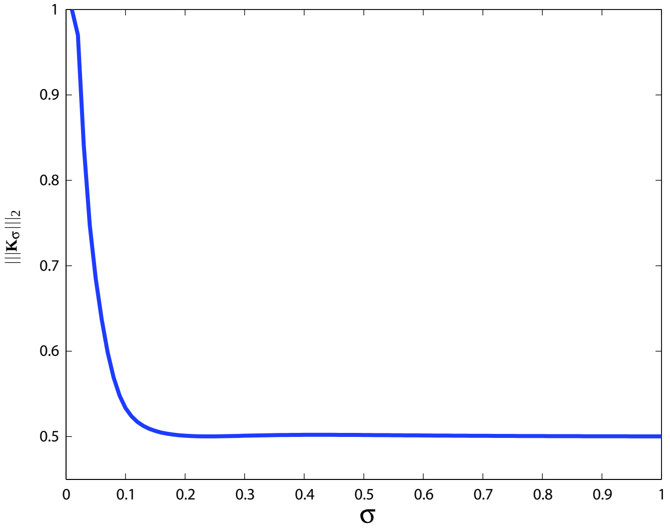


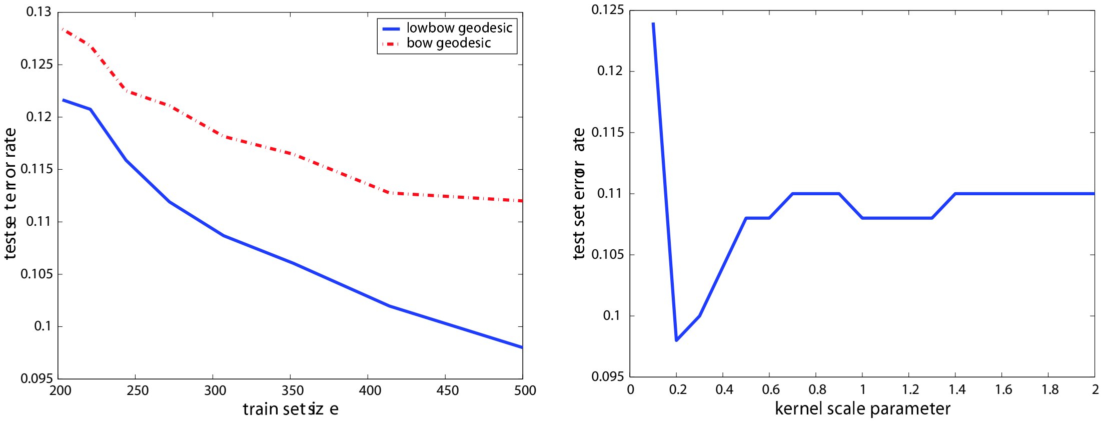
Figure 5: $|||\mathbf{K}_\sigma|||_2$ for the bounded Gaussian kernel (6) as a function of the kernel scale parameter σ. The continuous 2-norm definition in (15) is approximated by 5 equally spaced samples for μ and 20 equally spaced samples for t.

points is rather arbitrary in our case and we did not find it critical to the nature of the experimental results. Throughout the experiments we used the bounded Gaussian kernel (6) and computed several alternatives for the kernel scale parameter σ and chose the best one. While not entirely realistic, this setting enables us to examine lowbow's behavior in the optimistic scenario of being able to find the best scale parameter. The next section includes similar text classification experiments using SVM that explore further the issue of automatically selecting the scale parameter σ.

Figure 6 (top) displays results for nearest neighbor classification using the Fisher geodesic distance on the WebKB data. The left graph is a standard train-set size vs. test set error rate comparing the bow geodesic (lowbow with $\sigma \to \infty$) (dashed) and the lowbow geodesic distance. The right graph displays the dependency of the test set error rate on the scale parameter indicating an optimal scale at around $\sigma = 0.2$ (for repeated samplings of 500 training examples). In both cases, the performances of standard bow techniques such as tf cosine similarity or Euclidean distance were significantly inferior (20-40% higher error rate) than the Fisher geodesic distances and as a result are not displayed.

Figure 6 (bottom) displays test set error rates for the Reuters-21578 task. The 10 rows in the table indicate the classification task of identifying each of the 10 most popular classes in the Reuters collection. The columns represent varying training set sizes sampled from the mod-apte split. The lowbow geodesic distance for an intermediate scale is denoted by err_1 and for $\sigma \to \infty$ is denoted by err_2. Tf-cosine similarity and Euclidean distance for bow are denoted by err_3 and err_4.

The experiments indicate that lowbow geodesic clearly outperforms, for most values of σ, the standard tf-cosine similarity and Euclidean distance for bow (represented by $\text{err}_3, \text{err}_4$). In addition they also indicate that in general, the best scale parameter for lowbow is an intermediate one, rather than the standard bow model $\sigma \to \infty$ thus validating the hypothesis that we can leverage sequential



	Train Size = 100				Train Size = 200				Train Size = 400			
class	err_1	err_2	err_3	err_4	err_1	err_2	err_3	err_4	err_1	err_2	err_3	err_4
1	**9.9**	10.6	11.2	11.0	**8.1**	9.7	8.2	10.7	**6.7**	7.3	11.2	9.0
2	**11.6**	12.8	17.6	22.4	**9.4**	9.7	17.6	19.9	7.9	**7.8**	17.2	17.9
3	**6.8**	7.6	6.9	12.9	**5.8**	7.2	7.8	16.9	5.4	**5.3**	10.2	12.6
4	5.6	6.5	6.5	**5.5**	**4.8**	**4.8**	7.1	7.0	**4.5**	4.7	8.5	7.5
5	6.6	**6.2**	9.0	11.4	**5.7**	6.8	6.7	10.3	**5.0**	5.6	5.8	7.4
6	**5.7**	5.8	5.8	10.8	**5.2**	5.3	5.3	10.0	**4.8**	5.4	5.6	11.3
7	**4.2**	5.1	7.0	12.9	**4.2**	4.3	7.9	9.0	**3.9**	4.3	5.8	7.5
8	**3.0**	3.2	4.7	7.6	**3.0**	3.3	3.4	3.4	**2.6**	2.9	3.2	3.9
9	**2.8**	4.0	4.9	7.9	3.1	**3.0**	6.4	2.8	**2.9**	3.2	4.7	5.1
10	2.7	2.9	3.6	**2.6**	**2.6**	3.0	5.8	3.1	2.3	2.6	3.7	**2.2**

Figure 6: Experimental test set error rates for WebKB course vs. faculty task (top) and Reuters top 10 classes using samples from mod-apte split (bottom). err_1 is obtained using the lowbow geodesic distance with the optimal kernel scale. err_2–err_4 denote using geodesic distance, tf-Cosine similarity and Euclidean distance for bow.

information using the lowbow framework to improve on global bow models. The next section describes similar experiments using SVM on the RCV1 data set which include automatic selection of the scale parameter σ.

5.2 Text Classification using Support Vector Machine

We extended our WebKB and Reuters-21578 text classification experiments to the more recently released and larger RCV1 data set (Lewis et al., 2004). In particular, we focused on the 1 vs. all classification tasks for topics that correspond to leaf nodes in the topic hierarchy and contain no less than 5000 documents. This results in a total of 43 topic codes displayed in Table 1. For further description of the topic hierarchy of the RCV1 data set refer to Lewis et al. (2004).

In our experiments we examined the classification performance of SVM with the Fisher diffusion kernel for bow (Lafferty and Lebanon, 2005) and its corresponding product version for lowbow (10) (which reverts to the kernel of Lafferty and Lebanon (2005) for $\sigma \to \infty$). Our experiments validate the findings in Lafferty and Lebanon (2005) which indicate a significantly poorer performance for linear or RBF kernels. We therefore omit these results and concentrate on comparing the SVM performance for the kernel (10) using various values of σ.

We report the classification performance of SVM using the kernel (10) for three different values of σ: (i) $\sigma \to \infty$ represents the standard bow diffusion kernel of Lafferty and Lebanon (2005) (ii) σ_{opt} represents the best performing scale parameter in terms of test set error rate, and (iii) $\hat{\sigma}_{\text{opt}}$ represents an automatically selected scale parameter based on minimizing the leave-one-out cross validation (loocv) train-set error estimate computed by the SVM-light toolkit. In case of ties, we pick the σ with the smallest value, thus favoring less local smoothing. The loocv estimate is computed at no extra cost and is a convenient way to adaptively estimate σ_{opt}. In all of our experiments below we ignore the role of the diffusion time t in (10) and simply try several different values and choose the best one.

Table 1 reports the test set error rates and standard errors corresponding to the three scales $\hat{\sigma}_{\text{opt}}, \sigma \to \infty, \sigma_{\text{opt}}$ for the selected RCV1 1 vs. all classification tasks. Notice that in general, the lowbow $\hat{\sigma}_{\text{opt}}$ significantly outperforms the standard bow approach. The performance of σ_{opt} further improves on that indicating that a more intelligent scale selection method could result in even lower error rates. Table 1 is also displayed graphically in Figure 8 for $\hat{\sigma}_{\text{opt}}$ and $\sigma \to \infty$. Figure 7 shows the corresponding train set loocv error rates and standard errors.

In our experiments, the sampling of the train and test sets were balanced, with equal number of positive and negative examples. Selections of the optimal σ_{opt} and the estimated $\hat{\sigma}_{\text{opt}}$ were done based on the following set of possible values $\{0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 2, 4, 10, 100\}$. In all the classification tasks, lowbow performs substantially better than bow. The error bars indicate one standard deviation from the mean, and support experimentally the assertion that lowbow has lower variance.

Figure 9 compares the performance of lowbow for $\hat{\sigma}_{\text{opt}}$, $\sigma \to \infty$, and σ_{opt} as a function of the train set size (with the testing size being fixed as 200). As pointed out earlier, the performance of $\hat{\sigma}_{\text{opt}}$ is consistently better than bow with some room for improvement represented by the σ_{opt}.

5.3 Dynamic Time Warping of Lowbow Curves

As presented in the previous sections, the lowbow framework normalizes the time interval $[1,N]$ to $[0,1]$ thus achieving an embedding of documents of varying lengths in $\mathbb{P}_{V-1}^{[0,1]}$. Proceeding with the

	$\hat{\sigma}_{\text{opt}}$	$\sigma \to \infty$ (bow)	σ_{opt}
C11	0.1021 ± 0.0122	0.1234 ± 0.0198	0.0755 ± 0.0071
C12	0.0519 ± 0.0099	0.0664 ± 0.0161	0.0324 ± 0.0072
C13	0.1316 ± 0.0156	0.1527 ± 0.0232	0.1008 ± 0.0088
C14	0.0359 ± 0.0070	0.0537 ± 0.0151	0.0190 ± 0.0050
C1511	0.0494 ± 0.0105	0.0636 ± 0.0163	0.0296 ± 0.0067
C152	0.0860 ± 0.0117	0.1129 ± 0.0239	0.0660 ± 0.0075
C171	0.0522 ± 0.0113	0.0662 ± 0.0185	0.0310 ± 0.0067
C172	0.0313 ± 0.0077	0.0491 ± 0.0134	0.0175 ± 0.0053
C174	0.0066 ± 0.0044	0.0138 ± 0.0080	0.0003 ± 0.0011
C181	0.0634 ± 0.0105	0.0879 ± 0.0174	0.0444 ± 0.0066
C183	0.0283 ± 0.0083	0.0400 ± 0.0135	0.0126 ± 0.0036
C21	0.1269 ± 0.0151	0.1541 ± 0.0298	0.0985 ± 0.0105
C22	0.0614 ± 0.0121	0.0839 ± 0.0235	0.0400 ± 0.0063
C24	0.1009 ± 0.0147	0.1192 ± 0.0267	0.0725 ± 0.0078
C312	0.0494 ± 0.0097	0.0684 ± 0.0176	0.0299 ± 0.0059
C411	0.0321 ± 0.0084	0.0456 ± 0.0109	0.0156 ± 0.0050
C42	0.0550 ± 0.0132	0.0745 ± 0.0211	0.0347 ± 0.0071
E11	0.0356 ± 0.0088	0.0489 ± 0.0174	0.0196 ± 0.0049
E131	0.0213 ± 0.0066	0.0320 ± 0.0115	0.0083 ± 0.0038
E211	0.0372 ± 0.0079	0.0526 ± 0.0151	0.0233 ± 0.0045
E212	0.0293 ± 0.0077	0.0441 ± 0.0142	0.0150 ± 0.0038
E512	0.0568 ± 0.0091	0.0694 ± 0.0186	0.0339 ± 0.0054
E71	0.0051 ± 0.0042	0.0106 ± 0.0084	0.0000 ± 0.0000
G154	0.0155 ± 0.0067	0.0238 ± 0.0127	0.0043 ± 0.0033
GCRIM	0.0533 ± 0.0111	0.0730 ± 0.0164	0.0315 ± 0.0059
GDEF	0.0373 ± 0.0106	0.0526 ± 0.0164	0.0221 ± 0.0047
GDIP	0.0485 ± 0.0112	0.0694 ± 0.0180	0.0309 ± 0.0058
GDIS	0.0306 ± 0.0065	0.0451 ± 0.0190	0.0145 ± 0.0052
GENV	0.0466 ± 0.0106	0.0626 ± 0.0155	0.0301 ± 0.0045
GHEA	0.0298 ± 0.0088	0.0406 ± 0.0154	0.0148 ± 0.0045
GJOB	0.0512 ± 0.0117	0.0628 ± 0.0169	0.0308 ± 0.0057
GPOL	0.0675 ± 0.0099	0.0800 ± 0.0178	0.0434 ± 0.0073
GPRO	0.0624 ± 0.0094	0.0800 ± 0.0204	0.0414 ± 0.0068
GSPO	0.0035 ± 0.0032	0.0095 ± 0.0068	0.0000 ± 0.0000
GVIO	0.0359 ± 0.0080	0.0483 ± 0.0136	0.0185 ± 0.0044
GVOTE	0.0274 ± 0.0076	0.0415 ± 0.0130	0.0126 ± 0.0045
M11	0.0395 ± 0.0099	0.0602 ± 0.0165	0.0213 ± 0.0055
M12	0.0366 ± 0.0096	0.0495 ± 0.0120	0.0200 ± 0.0051
M131	0.0343 ± 0.0087	0.0485 ± 0.0131	0.0184 ± 0.0054
M132	0.0300 ± 0.0085	0.0401 ± 0.0134	0.0141 ± 0.0042
M141	0.0236 ± 0.0070	0.0379 ± 0.0122	0.0106 ± 0.0044
M142	0.0181 ± 0.0061	0.0311 ± 0.0102	0.0065 ± 0.0036
M143	0.0200 ± 0.0067	0.0320 ± 0.0101	0.0076 ± 0.0038

Table 1: Mean and standard error of the test set error rate over 40 realizations of 200 testing and 500 training documents for RCV1 C, E, G, M categories that also appear in Figure 8. Best achievable error rates for lowbow are also reported in the third column.

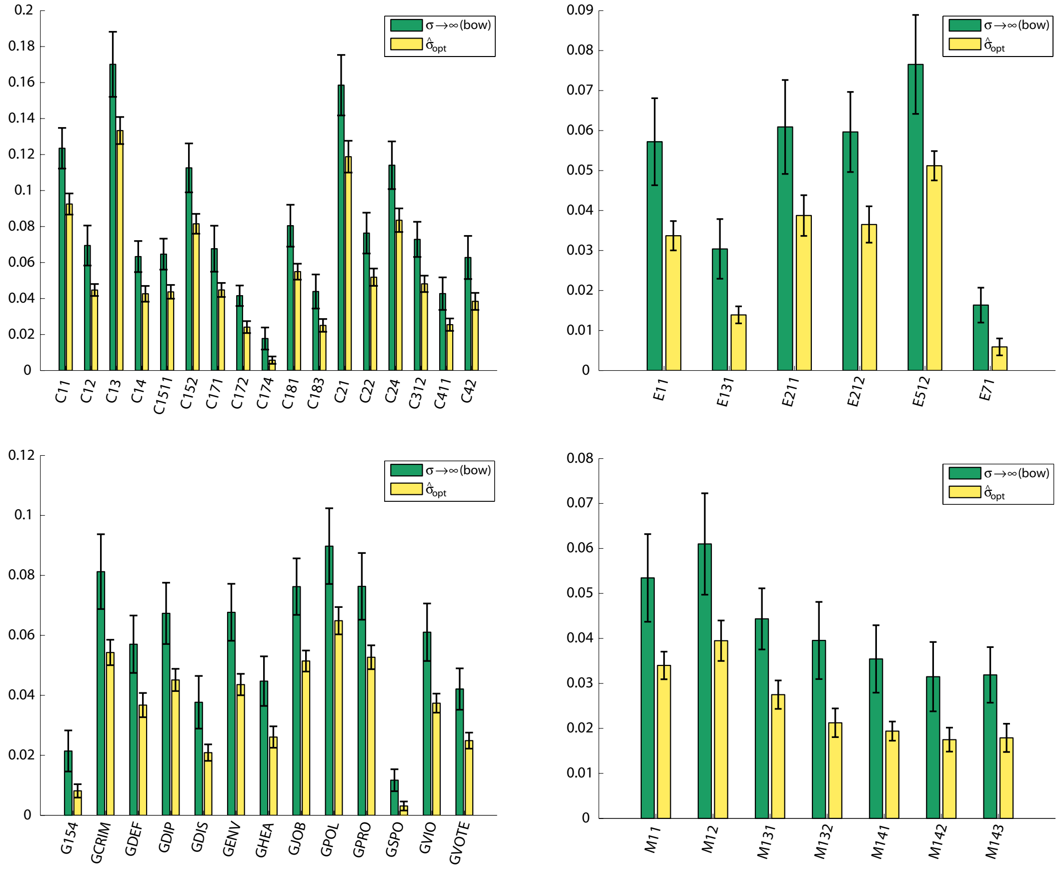


Figure 7: Mean and standard error of train set leave-one-out cross-validation (loocv) error rates. Results are averaged over 40 realizations of 500 training documents with a balanced positive and negative sampling. Lowbow results correspond to $\hat{\sigma}_{\text{opt}}$.

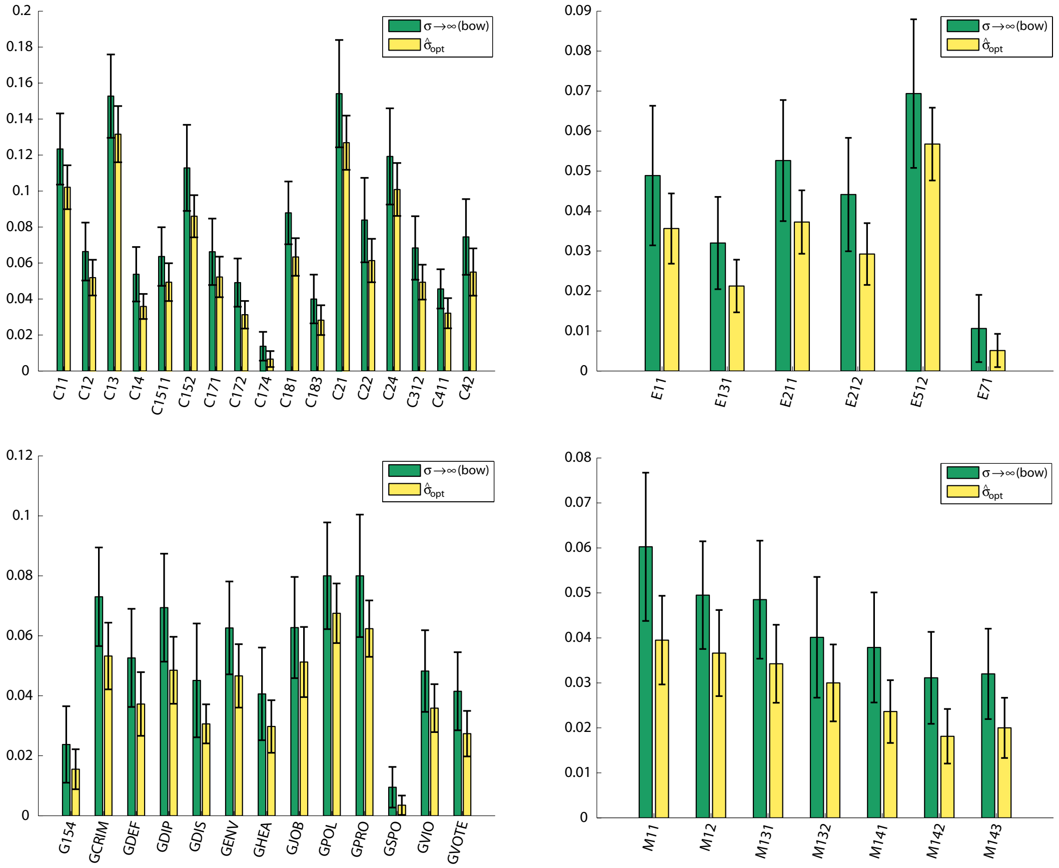


Figure 8: Mean and standard error of test set error rates. Results are averaged over 40 realizations of 500 training and 200 testing documents with a balanced positive and negative sampling. Lowbow results correspond to $\hat{\sigma}_{\text{opt}}$. See Table 1 for associated values.

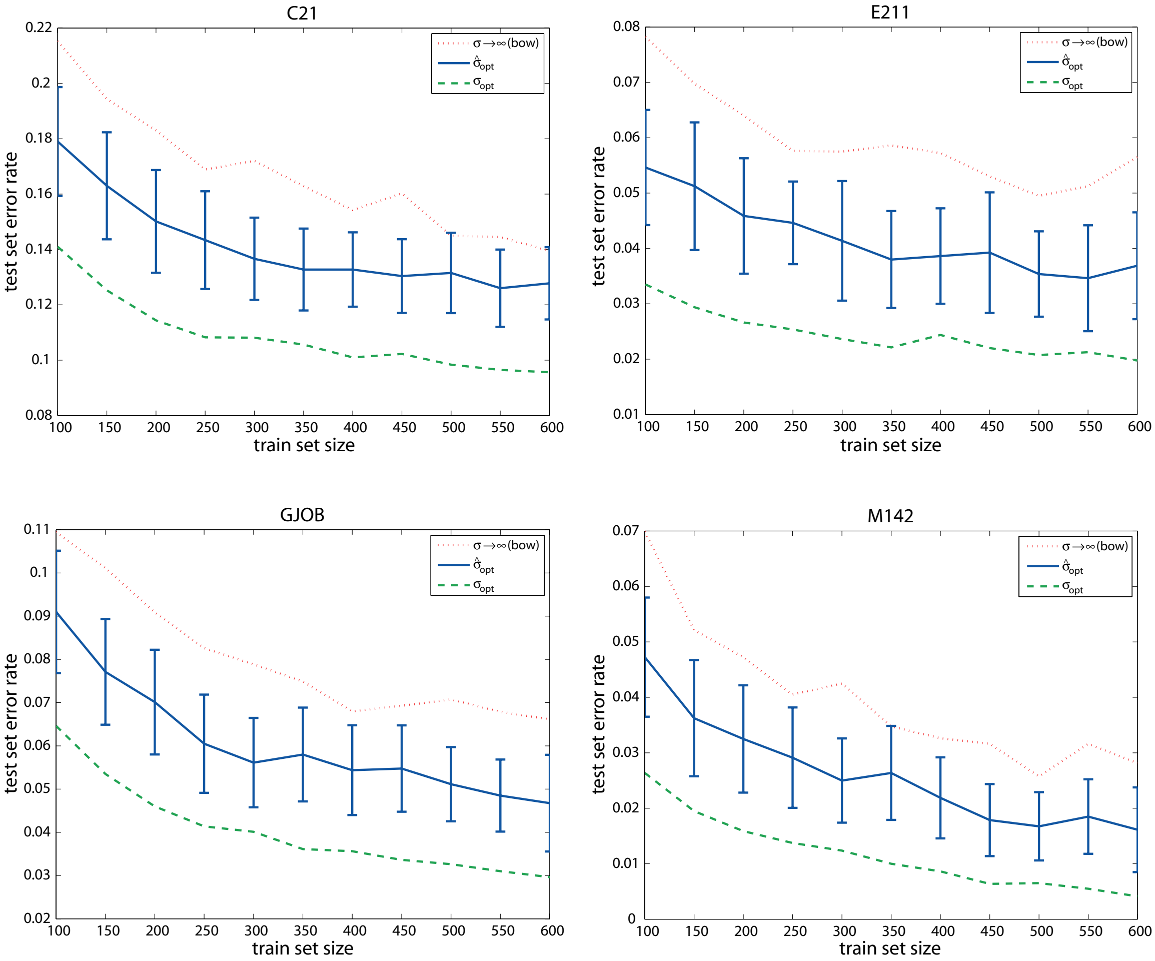


Figure 9: Test set error rate as a function of training size averaged over 40 realizations for RCV1 tasks C21, E211, GJOB and M142 (1 vs. all).

assumption of a product geometry, lowbow representations corresponding to different documents y,z relate to each other by comparing $\gamma_\mu(y)$ to $\gamma_\mu(z)$ for all $\mu \in [0,1]$, for example as is the case in the integrated distance

$$d(\gamma(y),\gamma(z)) = \int_0^1 d(\gamma_\mu(y),\gamma_\mu(z))\,d\mu. \tag{18}$$

This seems reasonable if the two documents y,z share a sequential progression of a similar rate, after normalizing for document length. However, such an assumption seems too restrictive in general as documents of different nature such as news stories and personal webpages are unlikely to posses such similar sequential progression. This assumption also seems untrue to a lesser extent for two documents written by different authors who posses their own individual styles. Such cases can be modeled by introducing time-warping or re-parameterization functions that match the individual temporal domains of lowbow curves to a unique canonical parameterization. Before proceeding to discuss such re-parameterization in the context of lowbow curves we briefly review their use in speech recondition and functional data analysis.

In speech recognition such re-parameterization functions are used to align the time axes corresponding to two speech signals uttered by different individuals or by the same individual under different circumstances. These techniques, commonly referred to as dynamic time warping (DTW) (Sakoe and Chiba, 1978), define the distance between two signals s,r as

$$d(s,r) = \min_{\iota_1,\iota_2 \in I} \int d(s(\iota_1(t)), r(\iota_2(t)))\,dt \tag{19}$$

where I represent the class of smooth monotonic increasing bijections $\iota : [0,1] \to [0,1]$. Using dynamic programming the discretized minimization problem corresponding to (19) can be efficiently computed, resulting in the wide spread use of DTW in the speech recognition community.

Similarly, such time parameterization techniques have been studied in functional data analysis under the name curve registration (Ramsay and Silverman, 2005). In contrast to dynamic time warping, curve registration is usually performed by an iterative procedure aimed at aligning salient features of the data and minimizing the post-alignment residual.

In contrast to the smoothness and monotonic nature of the re-parameterization class I in speech recognition and functional data analysis, it seems reasonable to allow some amount of discontinuity in lowbow re-parameterization. For example, while one document may posses a certain sequential progression, a second document may reverse the appearance of some of the sequential trends. Adjusting the original DTW definition of the re-parameterization family I we obtain the following modified characterization of the class of admissible re-parameterization.

Bijection Re-parameterization $\iota \in I$ are a bijection from $[0,1]$ onto itself.

Piecewise smoothness The re-parameterization functions $\iota \in I$ are piecewise smooth and monotonic, that is, given two partitions of $[0,1]$ to sequences of disjoint intervals $A_1,\ldots,A_r$ with $\cup A_j = [0,1]$ and $B_1,\ldots,B_r$ with $\cup B_j = [0,1]$ we have that for some permutation π over r items, $\iota : A_j \to B_{\pi(j)}$ is a smooth monotonic increasing bijection for all $j = 1,\ldots,r$.

The requirement above of piecewise continuity seems reasonable as it is natural to expect some re-ordering among sections or paragraphs of similar documents. Using a combination of dynamic programming similar to the one of Sakoe and Chiba (1978) and a variation of earth mover distance

(Rubner et al., 2000) known as the Hungarian algorithm (Munkres, 1957), the minimization problem (19) over the class I described above may be computed efficiently.

We conducted a series of experiments examining the benefit in introducing dynamic time warping or registration in text classification. Somewhat surprisingly, adding dynamic time warping or registration to lowbow classification resulted in only a marginal modification of the distances and consequently only a marginal improvement in classification performance. There are two reasons for this relatively minor effect introduced by the dynamic time warping. First, the RCV1 corpus for which these experiments were conducted consists of documents containing a fairly homogeneous semantic structure and presentation. As such, the curves can reasonably be compared by using integrated distances or kernels without a need for re-parameterization. Second, the local smoothing inherent in the lowbow representation makes it fairly robust to some amount of temporal misalignment. In particular, by selecting the kernel scale parameter appropriately we are able to prevent unfortunate effects due to different sequential parameterizations. Although surprising, this is indeed a positive result as it indicates that the lowbow representation is relatively robust to different time parameterization, at least when applied to documents sharing similar structure such as news stories in RCV1 corpus or webpages in the WebKB data set.

5.4 Text Segmentation

Text segmentation is the task of discovering topical boundaries inside documents, for example transcribed news-wire data. In general, this task is hard to accomplish using low order n-gram information. Most methods use a combination of longer range n-grams and other sequential features such as trigger pairs. Our approach in this section is not to carefully construct a state-of-the-art text segmentation system but rather to demonstrate the usefulness of the continuous lowbow representation in this context. More information on text segmentation and a recent exponential model-based approach may be found in Beeferman et al. (1999).

The boundaries between different text segments, by definition, separate document parts containing different word distributions. In the context of lowbow curves, this would correspond to sudden dramatic shifts in the curve location. Due to the continuity of the lowbow curves, such sudden movements may be discovered by considering the gradient vector field $\dot{\gamma}_\mu$ along the lowbow curve. In addition to containing predictive information that can be used in segmentation models, the gradient enables effective visualization of the instantaneous change that is central to human-assisted segmentation techniques.

To illustrate the role of the gradient $\dot{\gamma}_\mu(y)$ in segmentation we examine its behavior for a document y containing clear and pre-determined segments. Following Beeferman et al. (1999), we consider documents y created by concatenating news stories which resemble the continuous transcription of news stories. We examine the behavior of the lowbow curve and its gradient for two documents created in this fashion. The curves were sampled at 100 equally spaced points and the gradient is approximated by the finite difference in word histograms localized at adjacent time points. Generally speaking, for purpose of visualization the number of sampling points should be proportional to the length of the document in order to accurately capture the change in the local word histogram. This is different from the classification task which is not sensitive to the choice of the number of samples and thus favors a small value in the interest of lowering the computational complexity associated with classification.

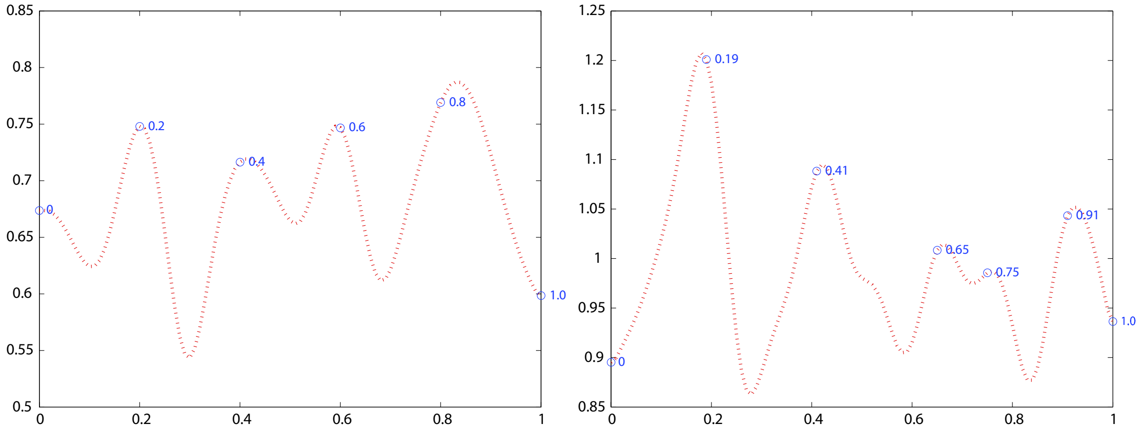


Figure 10: Velocity of the lowbow curve as a function of t. Left: five randomly sampled new stories of equal size ($\sigma = 0.08$). Right: three successive RCV1 news articles of varying lengths ($\sigma = 0.065$).

The first document was created by concatenating five randomly sampled news stories from the Wall Street Journal data set. To ensure that the different segments will be of equal length, we removed the final portions of the longer stories thus creating predetermined segment borders at $\mu = 0.2, 0.4, 0.6, 0.8$. The gradient norm $\|\dot{\gamma}_\mu(y)\|_2$ of this document is displayed in the the left panel of Figure 10. Notice how the 4 equally spaced internal segment borders (displayed by the numbered circles in the figure), correspond almost precisely, to the local maxima of the gradient norm.

The second document, represents a more realistic scenario where the segments correspond to successive news stories of varying lengths. We created it by randomly picking three successive news articles from the RCV1 collection (document id: 18101, 18102 and 18103) and concatenating them into a single document. The two internal segment borders occur at $\mu = 0.19$ and $\mu = 0.41$ (the last story is obviously longer than the first two stories). The right panel of Figure 10 displays the gradient norm $\|\dot{\gamma}_\mu\|_2$ for the corresponding lowbow curve. The curve has five local maxima, with the largest two local maxima corresponding almost precisely to the segment borders at $\mu = 0.19$ and $\mu = 0.41$. The three remaining local maxima correspond to internal segment boundaries within the third story. Indeed, the third news story begins with discussion of London shares and German stocks; it then switches to discuss French stocks at point $\mu = 0.65$ before switching again at $\mu = 0.75$ to talk about how the Bank of Japan's quarterly corporate survey affects the foreign exchange. The story moves on at $\mu = 0.95$ to discuss statistics of today's currencies and stock market. As with the different news story boundaries, the internal segment boundaries of the third story closely match the local maxima of the gradient norm.

The lowbow curve itself carries additional information beyond the gradient norm for segmentation purposes. Portions of the curve corresponding to different segments will typically contain different local word histograms and will therefore occupy different parts of the simplex. Sampling the curve at an equally spaced time grid $\{\mu_1, \dots, \mu_k\} \subset [0,1]$ and clustering the points $\{\gamma_{\mu_1}(y), \dots, \gamma_{\mu_k}(y)\}$ reveals distinct segments as distinct clusters. A similar approach uses partial human feedback to present to a user a low dimensional embedding of $\{\gamma_{\mu_1}(y), \dots, \gamma_{\mu_k}(y)\}$ given his or her choice of the scale parameter. The benefit in doing so is that it is typically much easier to visualize graphics than

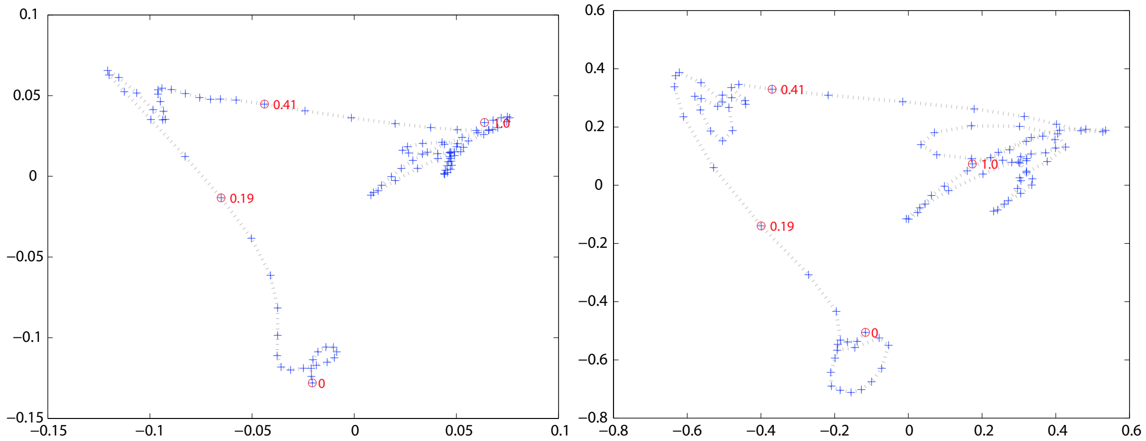


Figure 11: 2D embeddings of the lowbow curve representing the three successive RCV1 stories (see text for more details) using PCA (left, $\sigma = 0.02$) and MDS (right, $\sigma = 0.01$).

text content. Such techniques for rapid document visualization and browsing are also illustrated in the next section.

Figure 11 shows the 2D projection of the lowbow curve for the three concatenated RCV1 stories mentioned above. To embed the high dimensional curve in two dimensions we used principal component analysis (PCA) (left panel) and multidimensional scaling using the Fisher geodesic distance (right). The blue crosses indicate the positions of the sampled points in the low dimensional embedding while the red circles correspond to the segment boundaries of the three RCV1 documents. In both figures, $\{\gamma_{\mu_1}(y), \ldots, \gamma_{\mu_k}(y)\}$ are naturally grouped into three clusters, indicating the presence of three different segments. The distance between successive points near the segment boundaries is relatively large which demonstrates the high speed of the lowbow curve at these points (compare it with the gradient norm in right panel of Figure 10).

5.5 Text Visualization

We conclude the experiments with a text visualization demonstration based on the current journal article. Visualizing this document has the added benefit that the reader is already familiar with the text, hopefully having read it carefully thus far. Additional visualization applications of the lowbow framework may be found in Mao et al. (2007).

The gradient norm $\|\dot{\gamma}(t)\|_2$ of the lowbow curve of this paper is displayed in Figure 12. The marks indicate the beginning of each section that are identified by numbers in parentheses, for example, Section 2 begins at $\mu = 0.085$. Almost all of the local maxima of the curve correspond precisely to the section boundaries, with some interesting exceptions. For example, the global maximum occurs near $\mu = 0.17$ where we finish the lowbow definition (Definition 6) and start proving its properties (Theorem 1). Interestingly, the gradient speed does not distinguish between the two subsections concerning nearest neighbor and SVM classification experiments ($\mu = 0.61$). In performing the above experiment, the abstract, references and this subsection (Section 5.5) are excluded from the generation of the lowbow curve and equations were replaced by special markers.

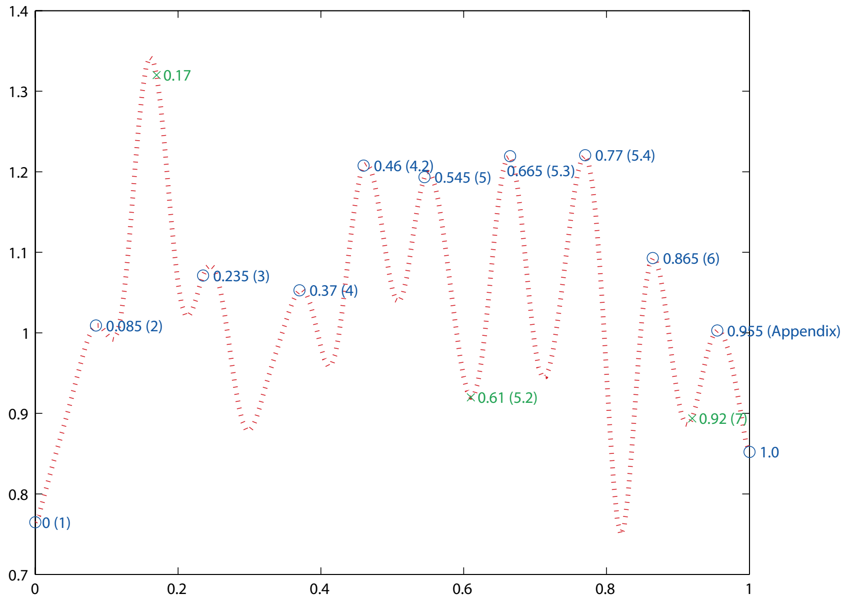


Figure 12: Velocity of the lowbow curve computed for this paper as a function of μ ($\sigma = 0.04$). Abstract, references and Section 5.5 are excluded from curve generation. The marks indicate the beginning of each section that are identified by numbers in parentheses, for example, Section 2 begins at $\mu = 0.085$.

Figure 13 depicts 2D projections of the lowbow curve corresponding to Sections 5.1–5.4 (section boundaries occurring at $\mu = \{0.2, 0.38, 0.72\}$) using PCA (left) and MDS (right) based on Fisher geodesic distance. As previously demonstrated, the lowbow curve nicely reveals three clusters corresponding to the different subsections with the exception of not distinguishing between the nearest neighbor and SVM experiments. Using interactive graphics it is possible to extract more information from the lowbow curves by examining the 3D PCA projection, displayed in Figure 14. The dense clustering of the points at the beginning of the 2D curve is separated in the 3D figure, however, there is no way to separate the crossing at the end of the curve in both 3D and 2D.

6. Related Work

The use of n-gram and bow has a long history in speech recognition, language modeling, information retrieval and text classification. Recent monographs surveying these areas are Jelinek (1998), Manning and Schutze (1999), and Baeza-Yates and Ribeiro-Neto (1999). In speech recognition and language modeling n-grams are used typically with $n = 1, 2, 3$. In classification, on the other hand, 1-grams or bow are the preferred option. While some attempts have been made to use bi-grams and tri-grams in classification as well as to incorporate richer representations, the bow representation is still by far the most popular.

Comparisons of several statistical methods for text classification using the bow representation may be found in Yang (1999). Joachims (2000) applies support vector machines to text classification using various 1-gram representations. Zhang and Oles (2001) consider several regularized linear

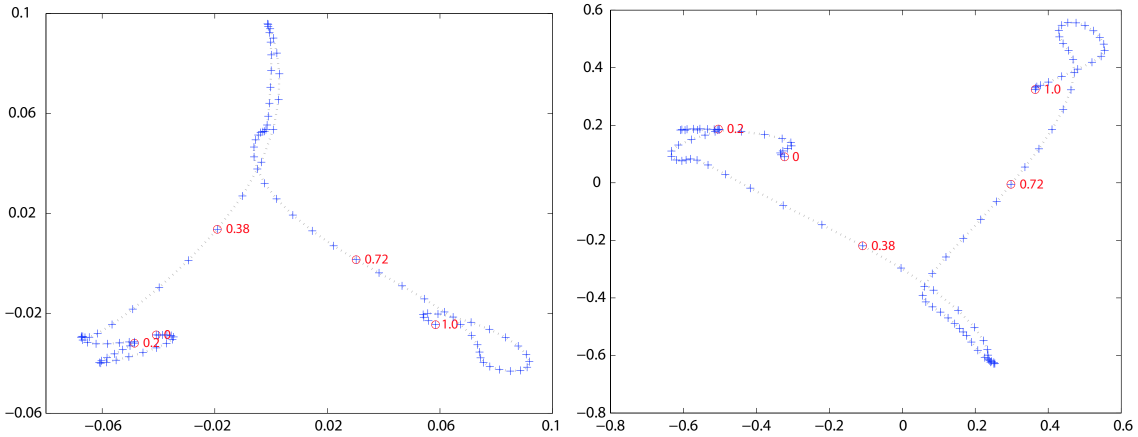


Figure 13: 2D embeddings of the lowbow curve computed for Section 5.1–5.4 using PCA (left, $\sigma = 0.03$) and MDS (right, $\sigma = 0.02$).

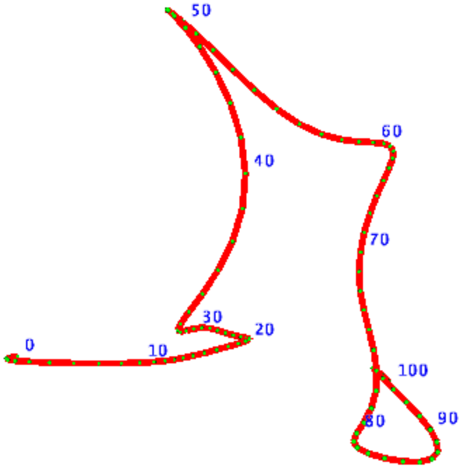


Figure 14: 3D embeddings of the lowbow curve computed for Section 5.1–5.4 using PCA ($\sigma = 0.03$). The numbers are $\mu \times 100$.

classifiers and Schapire and Singer (2000) experiment with AdaBoost. In general, most papers use a representation that is based on the word histogram with L_2 or L_1 normalization, binary word appearance events, or tfidf. The differences between the above representations tend to be minor and there is no clear conclusion which precise representation works best. Once the representation has been fixed, it is generally accepted that support vector machines with linear or rbf kernels result in the state-of-the-art performance, with logistic regression slightly trailing behind.

A geometric point of view considering the bow representation as a point in the multinomial simplex is expressed in Lebanon (2005) and Lafferty and Lebanon (2005). A recent overview of the geometrical properties of probability spaces is provided in Kass (1989) and Amari and Nagaoka (2000). The use of simplicial curves in text modeling is a relatively new approach but has been previously considered by Gous (1998) and Hall and Hofmann (2000). However, in contrast to these papers we represent a single document, rather than a corpus, as a curve in the simplex. The use of the heat or diffusion kernel in machine learning appeared first in Kondor and Lafferty (2002) in the context of graphs and later in Lafferty and Lebanon (2003) in the context of Riemannian manifolds. Cuturi (2005) describes some related ideas that lead to a non-smooth multi-scale view of images. These ideas were later expanded (Cuturi et al., 2007) to consider dynamic time warping which is highly relevant to the problem of matching two lowbow curves.

Modeling functional data such as lowbow curves in statistics has been studied in the context of functional data analysis. The recent monograph by Ramsay and Silverman (2005) provides an interesting survey and advocates the use of continuous representations even for data that is normally obtained in a discrete form. Wand and Jones (1995) and Loader (1999) provide a recent overview of local smoothing in statistics which is closely related to the lowbow framework.

Document visualization solutions were generally considered for visualizing a corpus of documents rather than visualizing a single document. Typical approaches include dimensionality reduction of the bow representation using methods such as multi-dimensional scaling and PCA. For examples see Fortuna et al. (2005) and Havre et al. (2002). IN-SPIRE is a document visualization tool[2] developed at Pacific Northwest National Lab that uses related ideas for corpus visualization. Blei and Lafferty (2006) use a dynamic extension of latent Dirichlet allocation (Blei et al., 2003) to explore and visualize temporal changes in a corpus of time-stamped documents. In contrast to most of the studies mentioned above, we are concerned with the sequentially modeling of a single document, rather than a corpus, at one or more sequential resolutions.

7. Discussion

The lowbow representation is a promising new direction in text modeling. By varying σ it interpolates between the standard word sequence representation $\langle y_1, \ldots, y_N \rangle$ and bow. In contrast to *n*-gram, it captures topical trends and incorporates long range information. On the other hand, the lowbow novelty is orthogonal to *n*-gram as it is possible to construct lowbow curves over *n*-gram counts.

Under our current model, two different lowbow curves are compared in a point-wise manner. When an attempt was made to register the curves by constructing a time warping function to synchronize the two curves, little or no improvement was found. This is due partly to the nature of the data and partly to the robustness of the lowbow representation being insensible to time-misalignment by adapting the scale parameter.

[2] IN-SPIRE can be found at `http://in-spire.pnl.gov/`.

In this paper we have focused on analyzing lowbow curves at one particular scale σ. An alternative and potentially more powerful approach is to consider a family of lowbow curves corresponding to a collection of scale parameters σ. The resulting family of curves constitute a multiresolution representation of documents similar to the use of Gabor filters and wavelets in signal processing. Exploring such sequential multiresolution representation and their relationship to wavelets should be more closely examined.

The lowbow framework is aesthetically pleasing, and achieves good results both in terms of numeric classification accuracy and in terms of presenting a convenient visual text representation to a user. Using a smoothing kernel, it naturally interpolates between bow and complete sequential information. In contrast to categorical smoothing methods employed by *n*-grams (such as back-off and interpolation) lowbow employs temporal smoothing which is potentially more powerful due to its ordinal nature. The correspondence with smooth curves in the simplex enables the use of a wide array of tools from differential geometry and analysis that are otherwise unavailable to standard discrete text representations.

Acknowledgments

The authors wish to thank Jeff Bilmes for his interesting comments regarding the lowbow representation, and the anonymous reviewers for their helpful suggestions. This work was supported in part by NSF grant DMS-0604486.

Appendix A. The Multinomial Simplex and its Geometry

In this section, we present a brief description of the multinomial simplex and its information geometry. Since the simplex is the space of bow representations, its geometry is crucial to the lowbow representation. The brief description below uses some concepts from Riemannian geometry. For additional information concerning the geometry of the simplex refer to Kass and Voss (1997), Amari and Nagaoka (2000), or Lebanon (2005). Standard textbooks on differential and Riemannian geometry are Spivak (1975), Lee (2002), and Boothby (2003).

The multinomial simplex $\mathbb{P}_m$ for $m > 0$ is the m-dimensional subset of $\mathbb{R}^{m+1}$ of all probability vectors or histograms over $m+1$ objects

$$\mathbb{P}_m = \left\{ \theta \in \mathbb{R}^{m+1} : \forall i \ \theta_i \geq 0, \sum_{j=1}^{m+1} \theta_j = 1 \right\}.$$

Its connection to the multinomial distribution is that every $\theta \in \mathbb{P}_m$ corresponds to a multinomial distribution over $m+1$ items.

The simplex definition above includes the boundary of vectors with zero probabilities. In order to formally consider the simplex as a differentiable manifold we need to remove that boundary and consider only strictly positive probability vectors. A discussion concerning the positivity restriction and its relative unimportance in practice may be found in Lafferty and Lebanon (2005).

The topological structure of $\mathbb{P}_m$, which determines the notions of convergence and continuity, is naturally inherited from the standard topological structure of the embedding space $\mathbb{R}^{m+1}$. The geometrical structure of $\mathbb{P}_m$ that determines the notions of distance, angle and curvature is determined by a local inner product $g_\theta(\cdot,\cdot), \theta \in \mathbb{P}_m$, called the Riemannian metric. The most obvious

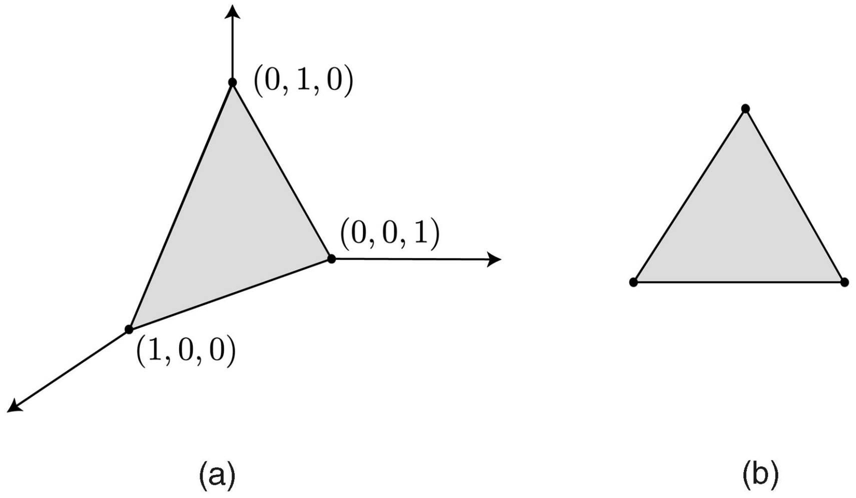


Figure 15: The 2-simplex $\mathbb{P}_2$ may be visualized as a surface in $\mathbb{R}^3$ (left) or as a triangle in $\mathbb{R}^2$ (right).

choice, perhaps, is the standard Euclidean inner product $g_\theta(u,v) = \sum u_i v_i$. However, such a choice is problematic from several aspects (Lebanon, 2005). A more motivated choice for a local inner product on the simplex is the Fisher information metric

$$\begin{aligned} g_\theta(u,v) &= \sum_{ij} u_i v_j \mathsf{E}_{p_\theta}\left(\frac{\partial \log p_\theta(x)}{\partial \theta_i}\frac{\partial \log p_\theta(x)}{\partial \theta_j}\right) \\ &= \sum_{i=1}^{m+1} \frac{u_i v_i}{\theta_i} \end{aligned} \tag{20}$$

where $p_\theta(x)$ above is the multinomial probability associated with the parameter θ. It can be shown that the Fisher information metric is the only invariant metric under sufficient statistics transformations (Čencov, 1982; Campbell, 1986). In addition, various recent results motivate the Fisher geometry from a practical perspective (Lafferty and Lebanon, 2005).

The inner product (20) defines the geometric properties of distance, angle and curvature on $\mathbb{P}_m$ in a way that is quite different from the Euclidean inner product. The distance function $d : \mathbb{P}_m \times \mathbb{P}_m \to [0, \pi/2]$ corresponding to (20) is

$$d(\theta,\eta) = \arccos\left(\sum_{i=1}^{m+1} \sqrt{\theta_i \eta_i}\right) \quad \theta,\eta \in \mathbb{P}_m. \tag{21}$$

The distance function (21) and the Euclidean distance function $d(\theta,\eta) = \sqrt{\sum(\theta_i - \eta_i)^2}$ resulting from the inner product $g_\theta(u,v) = \sum u_i v_i$ on $\mathbb{P}_2$ are illustrated in Figures 15-16.

By determining geometric properties such as distance, the choice of metric for $\mathbb{P}_m$ is of direct importance to the bow representation of documents and its modeling. For example, while the Euclidean metric is homogeneous across the simplex, the Fisher metric (20) emphasizes the area close to the boundary. In addressing the question of modeling the lowbow curves, the geometry of the simplex plays a central role. It dictates notions such as the distance between two curves,

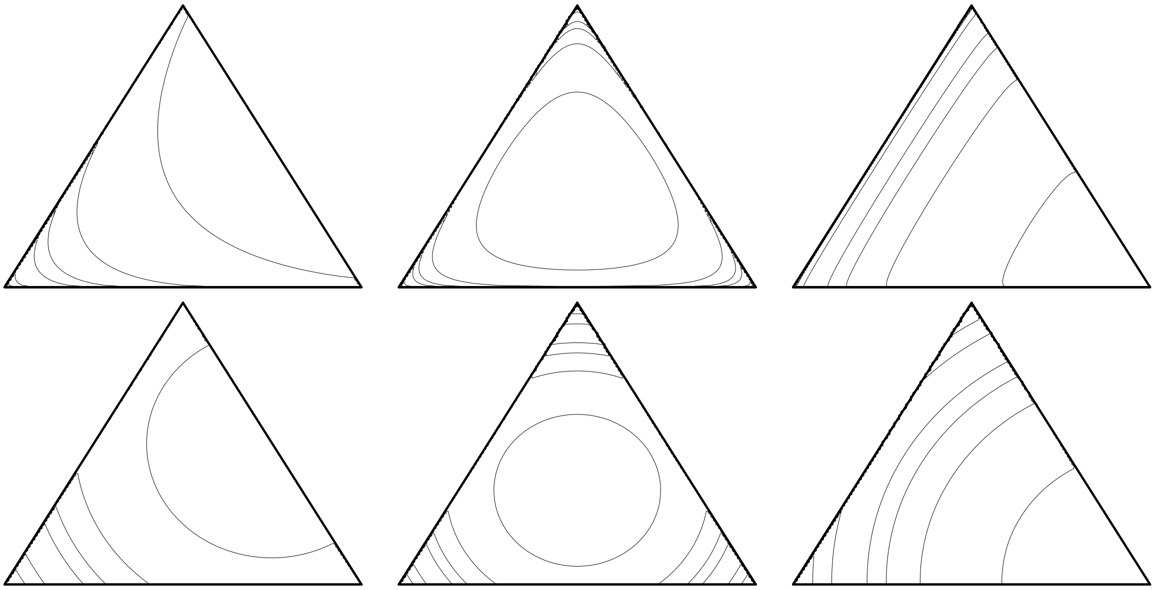


Figure 16: Equal distance contours on $\mathbb{P}_2$ from the upper right edge (left column), the center (center column), and lower right corner (right column). The distances are computed using the Fisher information metric (top row) and the Euclidean metric (bottom row).

the instantaneous direction of a curve, and the curvature or complexity of a curve. Understanding the relationship between these geometric notions and g_θ is a necessary prerequisite for modeling documents using the lowbow representation.

References

S.-I. Amari and H. Nagaoka. *Methods of Information Geometry*. American Mathematical Society, 2000.

M. Anthony and P. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999.

R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, 1999.

D. Beeferman, A. Berger, and J. D. Lafferty. Statistical models for text segmentation. *Machine Learning,* 34(1-3):177–210, 1999.

M. Berger, P. Gauduchon, and E. Mazet. Le spectre d'une varieté Riemannienne. *Lecture Notes in Mathematics*, Vol. 194, Springer-Verlag, 1971.

D. Blei and J. Lafferty. Dynamic topic models. In *Proceedings of the Twenty-Third International Conference on Machine Learning*, pages 113–120, 2006.

D. Blei, A. Ng, , and M. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

W. M. Boothby. *An Introduction to Differentiable Manifolds and Riemannian Geometry*. Academic Press, 2003.

L. L. Campbell. An extended Čencov characterization of the information metric. *Proceedings of the American Mathematical Society*, 98(1):135–141, 1986.

N. N. Čencov. *Statistical Decision Rules and Optimal Inference*. American Mathematical Society, 1982.

S. Chen and R. Rosenfeld. A survey of smoothing techniques for ME models. *IEEE Transactions on Speech and Audio Processing*, 8(1), 2000.

J. P. R. Christensen, C. Berg, and P. Ressel. *Harmonic Analysis on Semi-Groups*. Springer, 1984.

W. S. Cleveland and C. L. Loader. *Statistical Theory and Computational Aspects of Smoothing*, chapter Smoothing by Local Regression: Principles and Methods, pages 10–49. Springer, 1996.

M. Cuturi. *Learning from Structured Objects with Semigroup Kernels*. PhD thesis, Ecole des Mines de Paris, 2005.

M. Cuturi, J.-P, Vert, O. Birkenes, and T. Matsui. A kernel for time series based on global alignments. In *Proc. of the 32nd IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 413–416, 2007.

R. M. Dudley. A course on empirical processes. *Lecture Notes in Mathematics*, 1097:2–142, 1984.

G. Forman. Tackling concept drift by temporal inductive transfer. In *Proc. of the ACM SIGIR Conference*, 2006.

B. Fortuna, M. Grobelnik, and D. Mladenic. Visualization of text document corpus. *Informatica*, 29(4):497–504, 2005.

A. Gous. *Exponential and Spherical Subfamily Models*. PhD thesis, Stanford University, 1998.

Y. Guo, P. L. Bartlett, J. Shawe-Taylor, and R. C. Williamson. Covering numbers for support vector machines. *IEEE Transaction on Information Theory*, 48(1):239–250, 2002.

K. Hall and T. Hofmann. Learning curved multinomial subfamilies for natural language processing and information retrieval. In *Proc. of the 17th International Conference on Machine Learning*, pages 351–358, 2000.

S. Havre, E. Hetzler, P. Whitney, and L. Nowell. Themeriver: Visualizing thematic changes in large document collections. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):9–20, 2002.

M. Hein and O. Bousquet. Hilbertian metrics and positive definite kernels on probability measures. In *Proceedings of AI and Statistics*, pages 136–143, 2005.

R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1990.

F. Jelinek. *Statistical Methods for Speech Recognition*. MIT Press, 1998.

T. Joachims. *The Maximum Margin Approach to Learning Text Classifiers Methods, Theory and Algorithms*. PhD thesis, Dortmund University, 2000.

R. E. Kass. The geometry of asymptotic inference. *Statistical Science*, 4(3):188–234, 1989.

R. E. Kass and P. W. Voss. *Geometrical Foundation of Asymptotic Inference*. John Wiley & Sons, 1997.

R. Kondor and J. Lafferty. Diffusion kernels on graphs and other discrete structures. In *Proceedings of the 19th International Conference on Machine Learning*, 2002.

J. Lafferty and G. Lebanon. Information diffusion kernels. In *Advances in Neural Information Processing, 15*. MIT Press, 2003.

J. Lafferty and G. Lebanon. Diffusion kernels on statistical manifolds. *Journal of Machine Learning Research*, 6:129–163, January 2005.

G. Lebanon. *Riemannian Geometry and Statistical Machine Learning*. PhD thesis, Carnegie Mellon University, 2005.

G. Lebanon and Y. Zhao. Local likelihood modeling of the concept drift phenomenon. Technical Report 07-10, Statistics Department, Purdue University, 2007.

J. M. Lee. *Introduction to Smooth Manifolds*. Springer, 2002.

D. Lewis, Y. Yang, T. Rose, and F. Li. RCV1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5:361–397, 2004.

C. Loader. *Local Regression and Likelihood*. Springer, 1999.

C. D. Manning and H. Schutze. *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.

Y. Mao, J. Dillon, and G. Lebanon. Sequential document visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6), 2007.

J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, 1957.

J. Ramsay and B. W. Silverman. *Functional Data Analysis*. Springer, second edition, 2005.

J. O. Ramsay and C. J. Dalzell. Some tools for functional data analysis. *Journal of the Royal Statistical Society B*, 53(3):539–572, 1991.

Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2), 2000.

H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Trans. Acoust. Speech Signal Process*, 26(1):43–49, 1978.

R. E. Schapire and Y. Singer. Boostexter: A boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168, 2000.

M. Spivak. *A Comprehensive Introduction to Differential Geometry*, volume 1-5. Publish or Perish, 1975.

V. N. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, 1998.

M. P. Wand and M. C. Jones. *Kernel Smoothing*. Chapman and Hall/CRC, 1995.

R. C. Williamson, A. J. Smola, and B. Schölkopf. Generalization performance of regularization networks and support vector machines via entropy numbers of compact operators. *IEEE Transactions on Information Theory*, 47(6):2516–2532, 2001.

Y. Yang. An evaluation of statistical approaches to text categorization. *Journal of Information Retrieval*, 1(1/2):67–88, 1999.

Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proc. of ACM-SIGIR conference*, 2001.

T. Zhang. Covering number bounds for certain regularized linear function classes. *Journal of Machine Learning Research*, 2:527–550, 2002.

T. Zhang and F. J. Oles. Text categorization based on regularized linear classification methods. *Information Retrieval*, 4:5–31, April 2001.