Feature Selection Methods for Text Classification

[Extended Abstract]

Anirban Dasgupta
Yahoo! Research
Sunnyvale, CA 94089
anirban@yahoo-inc.com

Petros Drineas*
Rensselaer Polytechnic Institute
Troy, NY 12180
drinep@cs.rpi.edu

Boulos Harb*
University of Pennsylvania
Philadelphia, PA 19107
boulos@cis.upenn.edu

Vanja Josifovski
Yahoo! Research
Sunnyvale, CA 94089
vanjaj@yahoo-inc.com

Michael W. Mahoney
Yahoo! Research
Sunnyvale, CA 94089
mahoney@yahoo-inc.com

ABSTRACT

We consider feature selection for text classification both theoretically and empirically. Our main result is an unsupervised feature selection strategy for which we give worst-case theoretical guarantees on the generalization power of the resultant classification function $\tilde{f}$ with respect to the classification function f obtained when keeping all the features. To the best of our knowledge, this is the first feature selection method with such guarantees. In addition, the analysis leads to insights as to when and why this feature selection strategy will perform well in practice. We then use the TechTC-100, 20-Newsgroups, and Reuters-RCV2 data sets to evaluate empirically the performance of this and two simpler but related feature selection strategies against two commonly-used strategies. Our empirical evaluation shows that the strategy with provable performance guarantees performs well in comparison with other commonly-used feature selection strategies. In addition, it performs better on certain datasets under very aggressive feature selection.

Categories and Subject Descriptors: E.m [Data] : Miscellaneous; H.m [Information Systems] : Miscellaneous
General Terms: Algorithms, Experimentation
Keywords: Feature Selection, Text Classification, Random Sampling, Regularized Least Squares Classification

1. INTRODUCTION

Automated text classification is a particularly challenging task in modern data analysis, both from an empirical and from a theoretical perspective. This problem is of central interest in many internet applications, and consequently it has received attention from researchers in such diverse areas as information retrieval, machine learning, and the theory of algorithms. Challenges associated with automated text categorization come from many fronts: one must choose an appropriate data structure to represent the documents; one must choose an appropriate objective function to optimize in order to avoid overfitting and obtain good generalization; and one must deal with algorithmic issues arising as a result of the high formal dimensionality of the data.

Feature selection, i.e., selecting a subset of the features available for describing the data before applying a learning algorithm, is a common technique for addressing this last challenge [4, 13, 17, 20]. It has been widely observed that feature selection can be a powerful tool for simplifying or speeding up computations, and when employed appropriately it can lead to little loss in classification quality. Nevertheless, general theoretical performance guarantees are modest and it is often difficult to claim more than a vague intuitive understanding of why a particular feature selection algorithm performs well when it does. Indeed, selecting an optimal set of features is in general difficult, both theoretically and empirically; hardness results are known [5–7], and in practice greedy heuristics are often employed [4, 13, 17, 20].

We address these issues by developing feature selection strategies that are: *sufficiently simple* that we can obtain non-trivial provable worst-case performance bounds that accord with the practitioners' intuition; and at the same time *sufficiently rich* that they shed light on practical applications in that they perform well when evaluated against common feature selection algorithms. Motivated by recent work in applied data analysis—for example, work on Regularized Least Squares Classification (RLSC), Support Vector Machine (SVM) classification, and the Lasso shrinkage and selection method for linear regression and classification—that has a strongly geometric flavor, we view feature selection as a problem in dimensionality reduction. But rather than employing the Singular Value Decomposition (which, upon truncation, would result in a small number of dimensions, each of which is a linear combination of up to all of the original features), we will attempt to choose a small number of these features that preserve the relevant geometric structure in the data (or at least in the data insofar as the particular classification algorithm is concerned). We will see that this methodology is sufficiently simple and sufficiently rich so as to satisfy the dual criteria stated previously.

In somewhat more detail:

- We present a simple unsupervised algorithm for feature

*Work done in part while visiting Yahoo! Research.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
KDD'07, August 12–15, 2007, San Jose, California, USA.
Copyright 2007 ACM 978-1-59593-609-7/07/0008 ...$5.00.

selection and apply it to the RLSC problem. The algorithm assigns a univariate "score" or "importance" to every feature. It then randomly samples a small (independent of the total number of features, but dependent on the number of documents and an error parameter) number of features, and solves the classification problem induced on those features.

• We present a theorem which provides worst-case guarantees on the generalization power of the resultant classification function $\tilde{f}$ with respect to that of f obtained by using all the features. To the best of our knowledge, this is the first feature selection method with such guarantees.

• We provide additive-error approximation guarantees for any query document and relative-error approximation guarantees for query documents that satisfy a somewhat stronger but reasonable condition with respect to the training document corpus. Thus, the proof of our main quality-of-approximation theorem provides an analytical basis for commonly-held intuition about when such feature selection algorithms should and should not be expected to perform well.

• We provide an empirical evaluation of this algorithm on the TechTC-100, 20-Newsgroups, and Reuters-RCV2 datasets. In our evaluation, we use: the aforementioned univariate score function for which our main theorem holds—which, due to its construction, we will refer to as subspace sampling (SS); random sampling based on two other score functions that are simpler to compute—one based on weighted sampling (WS) and the other based on uniform sampling (US); as well as two common feature selection strategies—Information Gain (IG) and Document Frequency (DF)—that have been shown to perform well empirically (but for which the worst case analysis we perform would be quite difficult).

• We show that our main SS algorithm performs similarly to IG and DF and also to WS (which has a similar flavor to DF). In certain cases, e.g., under very aggressive feature selection on certain datasets, our main SS algorithm does better than the other methods. In other less aggressive cases, when it does similarly to IG, DF, and WS, we show that the univariate score that our provable SS algorithm computes is more closely approximated by the score provided by WS than it is at more aggressive levels of feature selection.

2. BACKGROUND AND RELATED WORK

2.1 Background

Learning a classification function can be regarded as approximating a multivariate function from sparse data. This problem is ill-posed and is solved in classical regularization theory by finding a function f that simultaneously has small empirical error and small norm in a Reproducing Kernel Hilbert Space (RKHS). That is, if the data consist of d examples $(z_1, y_1), \ldots, (z_d, y_d)$, where $z_i \in \mathbb{R}^d$ and $y_i \in \{-1, +1\}$, then one solves a Tikhonov regularization problem to find a function f that minimizes the functional:

$$\min_{f \in \mathcal{H}} \sum_{i=1}^{d} V(y_i, f(z_i)) + \lambda ||f||_K^2, \tag{1}$$

where $V(.,.)$ is a loss function, $||f||_K$ is a norm in a RKHS $\mathcal{H}$ defined by the positive definite function K, d is the number of data points, and λ is a regularization parameter [12, 34, 35]. Under general conditions [30], any $f \in \mathcal{H}$ minimizing (1) admits a representation of the form:

$$f(q) = \sum_{i=1}^{d} x_i K(q, z_i), \tag{2}$$

for some set of coefficients $x_i, i = \{1, \ldots, d\}$. Thus, the optimization problem (1) can be reduced to finding a set of coefficients $x_i, i = \{1, \ldots, d\}$. The theory of Vapnik then justifies the use of regularization functionals of the form appearing in (1) for learning from *finite* data sets [35].

If one chooses the square loss function,

$$V(y, f(z)) = (y - f(z))^2, \tag{3}$$

then, by combining (3) with (1) and (2), we obtain the following Regularized Least Squares Classification (RLSC) problem:

$$\min_{x \in \mathbb{R}^d} ||Kx - y||_2^2 + \lambda x^T K x, \tag{4}$$

where the $d \times d$ kernel matrix K is defined over the finite training data set and y is a d-dimensional $\{\pm 1\}$ class label vector [12, 26, 27].

As is standard, we will represent a document by an n-dimensional feature vector and thus a corpus of d training documents (where, generally, $n \gg d$) as an $n \times d$ matrix A. Similarly, we will consider an identity mapping to the feature space, in which case the kernel may be expressed as $K = A^T A$. If the Singular Value Decomposition (SVD) of A is $A = U \Sigma V^T$, then the solution and residual of (4) may be expressed as:

$$x_{\text{OPT}} \quad = \quad V(\Sigma^2 + \lambda I)^{-1} V^T y. \tag{5}$$

The vector x_{OPT} characterizes a classification function of the form (2) that generalizes well to new data. Thus, if $q \in \mathbb{R}^n$ is a new test or query document, then from (2) it follows that our binary classification function is:

$$f(q) = x_{\text{OPT}}^T A^T q. \tag{6}$$

That is, given a new document q that we wish to classify, if $f(q) > 0$ then q is classified as belonging to the class in question, and not otherwise.

2.2 Related Work

Feature selection is a large area. For excellent reviews, see [4, 13, 17, 20]. Papers more relevant to the techniques we employ include [14, 18, 24, 37, 39] and also [19, 22, 31, 36, 38, 40, 42]. Of particular interest for us will be the Information Gain (IG) and Document Frequency (DF) feature selection methods [39]. Hardness results have been described in [5–7].

RLSC has a long history and has been used for text classification: see, e.g., Zhang and Peng [41], Poggio and Smale [25], Rifkin, et. al. [27], Fung and Mangasarian (who call the procedure a Proximal Support Vector Machine) [15], Agarwal [3], Zhang and Oles [42], and Suykens and Vandewalle (who call the procedure a Least Squares Support Vector Machine) [33]. In particular, RLSC performs comparable to the popular Support Vector Machines (SVMs) for text categorization [15, 27, 33, 42]. Since it can be solved with vector space operations, RLSC is conceptually and theoretically simpler than SVMs, which require convex optimization techniques. In practice, however, RLSC is often slower, in particular for problems where the mapping to the feature space is not the identity (which are less common in text categorization applications). For a nice overview, see [26, 27].

We note in passing that if a hinge loss function is used instead of the square loss function of (3), i.e., if we set $V(f(x), y) = (1 - yf(x))_+$, where $(\xi)_+ = \max(\xi, 0)$, the classical SVM problem follows from (1). The proof of our main theorem will make use of matrix perturbation theory and the robustness of singular subspaces to the sampling implicit in our feature selection procedure; see [8, 11] and also [32]. We expect that our methodology will extend to SVM classification if one can prove analogous robustness results for the relevant convex sets in the SVM optimization.

3. OUR MAIN ALGORITHM

In this section, we describe our main sampling algorithm for feature selection and classification. Recall that we have a corpus of d training documents, each of which is described by $n \gg d$ features. Our main goal is to choose a small number r of features, where $d \lesssim r \ll n$, such that, by using only those r features, we can obtain good classification quality, both in theory and in practice, when compared to using the full set of n features. In particular, we would like to solve exactly or approximately a RLSC problem of the form (4) to get a vector to classify successfully a new document according to a classification function of the form (6).

In Figure 1, we present our main algorithm for performing feature selection, the SRLS ALGORITHM. The algorithm takes as input the $n \times d$ term-document (or feature-document) matrix A, a vector $y \in \mathbb{R}^d$ of document labels where $\text{sign}(y_j)$ labels the class of document $A^{(j)}$ (where $A^{(j)}$ denotes the j^{th} column of the matrix A and $A_{(i)}$ denotes the i^{th} row of A), and a query document $q \in \mathbb{R}^n$. It also takes as input a regularization parameter $\lambda \in \mathbb{R}^+$, a probability distribution $\{p_i\}_{i=1}^n$ over the features, and a positive integer r. The algorithm first randomly samples roughly r features according to the input probability distribution. Let $\tilde{A}$ be the matrix whose rows consist of the chosen feature vectors, rescaled appropriately, and let $\tilde{q}$ be the vector consisting of the corresponding elements of the input query document q, rescaled in the same manner. Then, if we define the $d \times d$ matrix $\tilde{K} = \tilde{A}^T\tilde{A}$ as our approximate kernel, the algorithm next solves the following RLSC problem:

$$\min_{x \in \mathbb{R}^d} \left\|\tilde{K}x - y\right\|_2^2 + \lambda x^T \tilde{K} x, \tag{7}$$

thereby obtaining an optimal vector $\tilde{x}_{\text{OPT}}$. Finally, the algorithm classifies the query q by computing,

$$\tilde{f} = f(\tilde{q}) = \tilde{q}^T \tilde{A} \tilde{x}_{\text{OPT}}. \tag{8}$$

If $\tilde{f} \geq 0$, then q is labeled 'positive'; and otherwise, q is labeled 'negative'. Our main theorem (in the next section) will give sufficient conditions on $\{p_i\}_{i=1}^n$ and r (as a function of A and λ) for relating the value of $\tilde{q}^T\tilde{A}\tilde{x}_{\text{OPT}}$ to the value of $q^T A x_{\text{OPT}}$ from (6) obtained by considering all n features.

An important aspect of our algorithm is the probability distribution $\{p_i\}_{i=1}^n$ input to the algorithm. One could perform random sampling with respect to any probability distribution. (Indeed, uniform sampling has often been presented as a "straw man" for other methods to beat.) On the other hand, as we show in Sections 4 and 6, more intelligent sampling can lead to improved classification performance, both theoretically and empirically. Also, note that rather than using the probability distribution $\{p_i\}_{i=1}^n$ over the features directly in r i.i.d. sampling trials (which might lead to the same feature being chosen multiple times), the SRLS

Input: $A \in \mathbb{R}^{n \times d}$; $y \in \mathbb{R}^d$; $q \in \mathbb{R}^n$; $\lambda \in \mathbb{R}^+$; $\{p_i \in [0,1] : i \in [n], p_i \geq 0, \sum_i p_i = 1\}$, and a positive integer $r \leq n$.

Output: A solution vector $\tilde{x}_{\text{OPT}} \in \mathbb{R}^d$; a residual $\tilde{\mathcal{Z}} \in \mathbb{R}$; and a classification $\tilde{f}$.

for $i = 1, \ldots, n$ **do**
  Pick i with probability $\tilde{p}_i = \min\{1, rp_i\}$;
  if i is picked **then**
    Include $A_{(i)}/\sqrt{\tilde{p}_i}$ as a row of $\tilde{A}$;
    Include $q_i/\sqrt{\tilde{p}_i}$ as the corresponding element of $\tilde{q}$;
end
Set $\tilde{K} = \tilde{A}^T\tilde{A}$;
Solve $\tilde{x}_{\text{OPT}} = \arg\min_{x \in \mathbb{R}^d} \left\|\tilde{K}x - y\right\|_2^2 + \lambda x^T \tilde{K} x$;
Set $\tilde{\mathcal{Z}} = \left\|\tilde{K}\tilde{x}_{\text{OPT}} - y\right\|_2^2 + \lambda \tilde{x}_{\text{OPT}}^T \tilde{K} \tilde{x}_{\text{OPT}}$;
Compute $\tilde{f} = f(\tilde{q}) = \tilde{q}^T \tilde{A} \tilde{x}_{\text{OPT}}$.

Figure 1: SRLS Algorithm: our main algorithm for Sampling for Regularized Least Squares classification.

ALGORITHM computes, for every $i \in \{1, \ldots, n\}$, a probability $\tilde{p}_i = \min\{1, rp_i\} \in [0, 1]$, and then the i^{th} row of A is chosen with probability $\tilde{p}_i$. Thus, r actually specifies an upper bound on the *expected* number of chosen rows of A: if X_i is a random variables that indicates whether the i^{th} row is chosen, then the expected number of chosen rows is $r' = \mathbf{E}\left[\sum_i X_i\right] = \sum_i \min\{1, rp_i\} \leq r \sum_i p_i = r$.

4. OUR MAIN THEOREMS

In this section, we provide our main quality-of-approximation theorems for the SRLS ALGORITHM. In these theorems, we will measure the quality of the classification by comparing the classification obtained from (6) using the output of an exact RLSC computation with the classification obtained from (8) using the output of the SRLS ALGORITHM, which operates on a much smaller set of features. The proof of these two theorems will be in Section 5.

Before stating these results, we review notation. Recall that: A is an $n \times d$ full-rank matrix, whose d columns correspond to d objects represented in an n-dimensional feature space; y is a d-dimensional class-indicator vector, i.e., the i^{th} entry of y denotes the class membership of the i^{th} object; $\lambda \geq 0$ is a regularization parameter. If we denote the SVD of A as $A = U\Sigma V^T$, then: U is the $n \times d$ matrix whose columns consist of the left singular vectors of A; $\sigma_{\max}$ and $\sigma_{\min}$ denote the largest and smallest, respectively, singular values of A; $\kappa_A = \sigma_{\max}/\sigma_{\min}$ is the condition number of A; and we will denote by $U^\perp$ any $n \times (n-d)$ orthogonal matrix whose columns span the subspace perpendicular to that spanned by the columns of U. In this case, a query document $q \in \mathbb{R}^n$ may be expressed as:

$$q = A\alpha + U^\perp \beta \ ,$$

for some vectors $\alpha \in \mathbb{R}^d$ and $\beta \in \mathbb{R}^{n-d}$. (Note that the "scale" of β is different from that of α, which for simplicity we have defined to account for the singular value information in A; this will manifest itself in the coefficients in the expressions of our main results.) Of course, $\tilde{A}$, $\tilde{q}$, $\tilde{x}_{\text{OPT}}$ are defined in the SRLS ALGORITHM of Figure 1, and x_{OPT} is an optimum of (4), as defined in (5).

Our first theorem establishes that if we randomly sam-

ple roughly $\tilde{O}(d/\epsilon^2)$ features according to a carefully chosen probability distribution of the form

$$p_i = \frac{\left\|U_{(i)}\right\|_2^2}{d} \quad , \quad \forall i \in 1 \ldots n, \tag{9}$$

i.e., proportional to the square of the Euclidean norms of the *rows* of the left singular vectors of the ($n \times d$ with $n \gg d$) matrix A, then we have an additive-error approximation bound for any query vector q. (We will use the common $\tilde{O}$ notation to hide factors that are polylogarithmic in d and ϵ for ease of exposition.)

THEOREM 1. *Let $\epsilon \in (0, 1/2]$ be an accuracy parameter. If the* SRLS ALGORITHM *(of Figure 1) is run with $r = \tilde{O}\left(d/\epsilon^2\right)$ and with sampling probabilities of the form (9), then, with probability at least* 0.98*:*

- *If $\lambda = 0$, then*

$$\left|\tilde{q}^T \tilde{A}\tilde{x}_{\text{OPT}} - q^T A x_{\text{OPT}}\right| \leq \frac{\epsilon\kappa_A}{\sigma_{\max}} \left\|\beta\right\|_2 \left\|y\right\|_2.$$

- *If $\lambda > 0$, then*

$$\left|\tilde{q}^T \tilde{A}\tilde{x}_{\text{OPT}} - q^T A x_{\text{OPT}}\right| \leq 2\epsilon\kappa_A \left\|\alpha\right\|_2 \left\|y\right\|_2 + \frac{2\epsilon\kappa_A}{\sigma_{\max}} \left\|\beta\right\|_2 \left\|y\right\|_2. \tag{10}$$

Note that (except for the trivial case where $\lambda = 0$), our theorem provides no guideline for the choice of λ. The second bound holds regardless of the choice of λ, which we will see is conveniently eliminated in the proof.

Note that the error bounds provided by Theorem 1 for the classification accuracy of our feature selection algorithm depend on: the condition number of A—this is a very common dependency in least squares problems; the amount of the query vector q that is "novel" with respect to the training set documents—$\left\|\beta\right\|_2$ measures how much of q lies outside the subspace spanned by the training set documents; as well as the alignment of the class membership vector y with the part of the query document q that lies in the subspace spanned by the columns of A. In particular, notice, for example, that if $\beta = 0$, namely if there is no "novel" component in the query vector q (equivalently, if q may be expressed as a linear combination of the documents that we have already seen without any information loss), then the error becomes exactly zero if $\lambda = 0$. If $\lambda > 0$ and $\beta = 0$, then the second term in (10) is zero.

One important question is whether one can achieve relative error guarantees. We are particularly interested in the case where $\beta = 0$ (the query vector has no new components), and $\lambda > 0$. The following theorem states that under additional assumptions we get such relative error guarantees. In particular, we need to make an assumption about how the query vector q interacts with the class discrimination vector y, so that we can replace the product of norms with the norm of products.

THEOREM 2. *Let $\epsilon \in (0, 1/2]$ be an accuracy parameter, and let $\lambda > 0$. Assume that the query document q lies entirely in the subspace spanned by the d training documents (the columns of A), and that the two vectors $V^T y$ and $\left(I + \lambda\Sigma^{-2}\right)^{-1} V^T \alpha$ are "close" (i.e., almost parallel or anti-parallel) to each other, in the sense that*

$$\left\|\left(I + \lambda\Sigma^{-2}\right)^{-1} V^T \alpha\right\|_2 \left\|V^T y\right\|_2 \leq \gamma \left\|\left(\left(I + \lambda\Sigma^{-2}\right)^{-1} V^T \alpha\right)^T V^T y\right\|_2 = \gamma \left|q^T A x_{\text{OPT}}\right|,$$

for some small constant γ. If the SRLS ALGORITHM *(of Figure 1) is run with $r = \tilde{O}\left(d/\epsilon^2\right)$ and with sampling probabilities of the form (9), then, with probability at least* 0.98*,*

$$\left|\tilde{q}^T \tilde{A}\tilde{x}_{\text{OPT}} - q^T A x_{\text{OPT}}\right| \leq 2\epsilon\gamma\kappa_A \left|q^T A x_{\text{OPT}}\right|.$$

Recall that the vector α contains the coefficients in the expression of q as a linear combination of the columns of A; hence, the assumption of Theorem 2 is related to the assumption that α is "close" to the classification vector y. (Notice that V is a full rotation, and $\left(I + \lambda\Sigma^{-2}\right)^{-1}$ essentially discounts the smaller singular values of A.) Thus, Theorem 2 quantifies the intuition that query vectors that are clearly correlated with the class discrimination axis will have smaller classification error. On the other hand, Theorem 1 indicates that ambiguous query vectors (i.e., vectors that are nearly perpendicular to the class indicator vector) will have higher classification errors after sampling since such vectors depend on almost all their features for accurate classification.

5. PROOF OF OUR MAIN THEOREMS

5.1 The sampling matrix formalism

For notational convenience in the proofs in this section, we define an $n \times n$ diagonal sampling matrix S. The diagonal entries of this matrix are determined by "coin flips" of the SRLS ALGORITHM. In particular, for all $i = 1, \ldots, n$, $S_{ii} = 1/\tilde{p}_i$ with probability $\tilde{p}_i = \min\{1, rp_i\}$; otherwise, $S_{ii} = 0$. Here, the p_i are defined in (9). Intuitively, the non-zero entries on the diagonal of S correspond to the rows of A that the algorithm selects.

5.2 Matrix perturbation results

Our proof will rely on the following three lemmas from matrix perturbation theory.

LEMMA 3. *For any matrix E such that $I + E$ is invertible, $(I + E)^{-1} = I + \sum_{i=1}^{\infty} (-E)^i$.*

LEMMA 4. *Let X and $\tilde{X} = X + E$ be invertible matrices. Then $\tilde{X}^{-1} - X^{-1} = -X^{-1} E \tilde{X}^{-1}$.*

For a proof, see Stewart and Sun [32], pp. 118.

LEMMA 5. *Let D and X be matrices such that the product DXD is a symmetric positive definite matrix with $X_{ii} = 1$. Let the product DED be a perturbation such that*

$$\left\|E\right\|_2 = \eta < \lambda_{\min}(X).$$

Here $\lambda_{\min}(X)$ corresponds to the smallest eigenvalue of X. Let λ_i be the i-th eigenvalue of DXD and let $\tilde{\lambda}_i$ be the i-th eigenvalue of $D(X + E)D$. Then,

$$\left|\frac{\lambda_i - \tilde{\lambda}_i}{\lambda_i}\right| \leq \frac{\eta}{\lambda_{\min}(X)}. \tag{11}$$

For a proof, see Demmel and Veselić [10].

5.3 Invertibility of matrices

Let $A = U\Sigma V^T$ be the SVD of A. Define,

$$\Delta = \Sigma U^T S U \Sigma = \Sigma(I + E)\Sigma \ . \tag{12}$$

Here E denotes how far away $U^T SU$ is from the identity. We will apply Lemma 5 on the matrix product $\Sigma U^T U\Sigma$, which is symmetric positive definite. (Notice that the matrix D of the lemma is Σ and the matrix X of the lemma is $U^T U = I$, thus $X_{ii} = 1$ for all i.) Towards that end, we need to bound the spectral norm of E, which has been provided by Rudelson and Vershynin in [28].

LEMMA 6. *Let $\epsilon \in (0, 1/2]$. Let $\tilde{p}_i = min\{1, rp_i\}$, let p_i be as in* (9)*, and let $r = \tilde{O}(d/\epsilon^2)$. Then, with probability at least 0.99,*

$$\|E\|_2 = \left\|I - U^T SU\right\|_2 = \left\|U^T U - U^T SU\right\| \leq \epsilon < 1.$$

We can now immediately apply Lemma 5, since the spectral norm of the perturbation is strictly less than one, which is the smallest eigenvalue of $U^T U = I$. Since Δ is symmetric positive definite, the i-th eigenvalue of Δ is equal to the i-th singular value of Δ; also, the i-th eigenvalue of $\Sigma U^T U\Sigma$ is equal to σ_i^2, where $\sigma_i = \Sigma_{ii}$. Thus Lemma 5 implies

LEMMA 7. *Let δ_i be the singular values of Δ. Then, with probability at least 0.99,*

$$\mid \delta_i - \sigma_i^2 \mid \ \leq \ \epsilon\sigma_i^2 \tag{13}$$

for all $i = 1 \ldots d$.

The following lemma is the main result of this section and states that all the matrices of interest are invertible.

LEMMA 8. *Using the above notation: Σ^2 is invertible; $\Sigma^2 + \lambda I$ is invertible for any $\lambda \geq 0$; Δ is invertible with probability at least* 0.99*; $\Delta + \lambda I$ is invertible for any $\lambda \geq 0$ with probability at least* 0.99*; and $I + E$ is invertible with probability at least* 0.99*.*

Proof. The first two statements follow trivially from the fact that A is full-rank. The third statement follows from Lemma 7 and the fourth statement follows by an analogous argument (omitted). The last statement follows from the fact that the spectral norm of E is at most ϵ, hence the singular values of $I + E$ are between $1 - \epsilon$ and $1 + \epsilon$.

5.4 The classification error of $\tilde{q}^T \tilde{A}\tilde{x}_{\text{OPT}}$

In this subsection, we will bound the difference between $q^T Ax_{\text{OPT}}$ and $\tilde{q}^T \tilde{A}\tilde{x}_{\text{OPT}}$. This bound provides a margin of error for the generalization power of $\tilde{A}\tilde{x}_{\text{OPT}}$ in classifying an *arbitrary* new document q with respect to the generalization power of $q^T Ax_{\text{OPT}}$. The following lemma provides a nice expression for $\tilde{x}_{\text{OPT}}$.

LEMMA 9. *With probability at least* 0.99*,*

$$\tilde{x}_{\text{OPT}} = V(\Delta + \lambda I)^{-1} V^T y \ .$$

Sketch of the proof. Writing down the normal equations for the sampled problem, and using the orthogonality of V and the invertibility of $\Delta + \lambda I$ and Δ provides the formula for $\tilde{x}_{\text{OPT}}$.

We now expand q into two parts: the part that lies in the subspace spanned by the columns of A and the part that lies in the perpendicular subspace, i.e., in the span of $U^\perp$:

$$q = A\alpha + U^\perp \beta \ . \tag{14}$$

Using $A = U\Sigma V^T$ and substituting x_{OPT} from (5), we get

$$\begin{aligned} q^T Ax_{\text{OPT}} &= \alpha^T A^T Ax_{\text{OPT}} + \beta^T U^{\perp T}(U\Sigma V^T)x_{\text{OPT}} \\ &= \alpha^T V\Sigma^2(\Sigma^2 + \lambda I)^{-1} V^T y \qquad (15) \\ &= \alpha^T V(I + \lambda\Sigma^{-2})^{-1} V^T y \ . \qquad (16) \end{aligned}$$

In the above we used the fact that $U^{\perp T} U = 0$ and the invertibility of Σ^2 and $\Sigma^2 + \lambda I$. We now focus on $\tilde{q}^T \tilde{A}\tilde{x}_{\text{OPT}}$, which may be rewritten (using our sampling matrix formalism from Section 5.1) as $q^T SA\tilde{x}_{\text{OPT}}$.

$$\begin{aligned} \left|q^T Ax_{\text{OPT}} - \tilde{q}^T \tilde{A}\tilde{x}_{\text{OPT}}\right| &= \left|q^T Ax_{\text{OPT}} - q^T SA\tilde{x}_{\text{OPT}}\right| \\ &\leq \left|q^T Ax_{\text{OPT}} - \alpha^T A^T SA\tilde{x}_{\text{OPT}}\right| \qquad (17) \\ &\quad + \left|\beta^T U^{\perp T} SA\tilde{x}_{\text{OPT}}\right| \ . \qquad (18) \end{aligned}$$

We will bound (17) and (18) separately. Using the formula for $\tilde{x}_{\text{OPT}}$ from Lemma 9 and $\Delta = \Sigma U^T SU\Sigma = \Sigma\,(I + E)\,\Sigma$ we get

$$\begin{aligned} \alpha^T A^T SA\tilde{x}_{\text{OPT}} &= \alpha^T V\Delta V^T \tilde{x}_{\text{OPT}} \\ &= \alpha^T V\Delta(\Delta + \lambda I)^{-1} V^T y \\ &= \alpha^T V(I + \lambda\Delta^{-1})^{-1} V^T y \\ &= \alpha^T V(I + \lambda\Sigma^{-1}\,(I + E)^{-1}\,\Sigma^{-1})^{-1} V^T y \\ &= \alpha^T V\left(I + \lambda\Sigma^{-2} + \lambda\Sigma^{-1}\Phi\Sigma^{-1}\right)^{-1} V^T y \ . \end{aligned}$$

To understand the last derivation notice that, from Lemma 3, $(I + E)^{-1} = I + \Phi$, where $\Phi = \sum_{i=1}^{\infty}(-E)^i$. We now bound the spectral norm of Φ.

$$\|\Phi\|_2 = \left\|\sum_{i=1}^{\infty}(-E)^i\right\|_2 \leq \sum_{i=1}^{\infty}\|E\|_2^i \leq \sum_{i=1}^{\infty}\epsilon^i = \frac{\epsilon}{1-\epsilon} \ , \tag{19}$$

using Lemma 6 and the fact that $\epsilon \leq 1/2$. We are now ready to bound (17).

$$\left|q^T Ax_{\text{OPT}} - \alpha^T A^T SA\tilde{x}_{\text{OPT}}\right| =$$
$$\left|\alpha^T V\left[\left(I + \lambda\Sigma^{-2} + \lambda\Sigma^{-1}\Phi\Sigma^{-1}\right)^{-1} - \left(I + \lambda\Sigma^{-2}\right)^{-1}\right] V^T y\right|.$$

Using Lemma 4 and noticing that all matrices involved are invertible, we bound the above quantity by

$$\left\|\alpha^T V\left(I + \lambda\Sigma^{-2}\right)^{-1}\right\|_2 \left\|V^T y\right\|_2 \|\Psi\|_2 \ ,$$

where $\Psi = \lambda\Sigma^{-1}\Phi\Sigma^{-1}\left(I + \lambda\Sigma^{-2} + \lambda\Sigma^{-1}\Phi\Sigma^{-1}\right)^{-1}$. In order to complete the bound for the term in (17) we bound the spectral norm of Ψ.

$$\begin{aligned} \Psi &= \lambda\Sigma^{-1}\Phi\Sigma^{-1}\left(\Sigma^{-1}\left(\Sigma^2 + \lambda I + \lambda\Phi\right)\Sigma^{-1}\right)^{-1} \\ &= \lambda\Sigma^{-1}\Phi\left(\Sigma^2 + \lambda I + \lambda\Phi\right)^{-1}\Sigma \ . \end{aligned}$$

Since we already have bounds for the spectral norms of Σ, Σ^{-1}, and Φ, we only need to bound the spectral norm of $\left(\Sigma^2 + \lambda I + \lambda\Phi\right)^{-1}$. Notice that the spectral norm of this matrix is equal to the inverse of the smallest singular value of $\Sigma^2 + \lambda I + \lambda\Phi$. Standard perturbation theory of matrices [32] and (19) imply that,

$$\left|\sigma_i\left(\Sigma^2 + \lambda I + \lambda\Phi\right) - \sigma_i\left(\Sigma^2 + \lambda I\right)\right| \leq \|\lambda\Phi\|_2 \leq \epsilon\lambda.$$

Here $\sigma_i(X)$ denotes the i^{th} singular value of the matrix X. Since $\sigma_i\left(\Sigma^2+\lambda I\right)=\sigma_i^2+\lambda$, where σ_i are the singular values of A,

$$\sigma_i^2+(1-\epsilon)\lambda \le \sigma_i\left(\Sigma^2+\lambda I+\lambda\Phi\right) \le \sigma_i^2+(1+\epsilon)\lambda.$$

Thus,

$$\left\|\left(\Sigma^2+\lambda I+\lambda\Phi\right)^{-1}\right\| = 1/\sigma_{\min}\left(\Sigma^2+\lambda I+\lambda\Phi\right) \le 1/\left(\sigma_{min}^2+(1-\epsilon)\lambda\right).$$

Here we let $\sigma_{\min}$ be the smallest singular value of A, and $\sigma_{\max}$ be the largest singular value of A. Combining all the above and using the fact that $\|\Sigma\|_2\left\|\Sigma\right\|_2^{-1}=\sigma_{\max}/\sigma_{\min}\le\kappa_A$ (the condition number of A), we bound (17):

$$\left|q^TAx_{\text{OPT}}-\alpha^TA^TSA\tilde{x}_{\text{OPT}}\right| \le \frac{\epsilon\lambda\kappa_A}{\sigma_{min}^2+(1-\epsilon)\lambda}\left\|\alpha^TV\left(I+\lambda\Sigma^{-2}\right)^{-1}\right\|_2\left\|V^Ty\right\|_2. \quad (20)$$

We now proceed to bound the term in (18).

$$\begin{aligned}&\left|\beta^TU^{\perp^T}SU\Sigma(\Delta+\lambda I)^{-1}V^Ty\right| \\ &\le \left\|q^TU^\perp U^{\perp^T}SU\right\|_2\left\|\Sigma(\Delta+\lambda I)^{-1}\right\|_2\left\|V^Ty\right\|_2 \\ &\le \epsilon\left\|U^\perp U^{\perp^T}q\right\|_2\left\|V^Ty\right\|_2\left\|\Sigma(\Delta+\lambda I)^{-1}\right\|_2 \\ &\le \epsilon\left\|\beta\right\|_2\left\|y\right\|_2\left\|\Sigma(\Delta+\lambda I)^{-1}\right\|_2 ,\end{aligned}$$

where the first inequality follows from $\beta=U^{\perp^T}q$; and the second inequality follows from the lemma below (whose proof is omitted—it is similar to Lemma 4.3 from [11]).

LEMMA 10. *Let $\epsilon\in(0,1/2]$. Given our notation, and our choices for $\tilde{p}_i$, p_i, and r, with probability at least 0.99,*

$$\left\|q^TU^\perp U^{\perp^T}SU\right\|_2 \le \epsilon\left\|U^\perp U^{\perp^T}q\right\|_2 .$$

To conclude the proof, we will bound the spectral norm of

$$\begin{aligned}\Sigma\left(\Delta+\lambda I\right)^{-1} &= \left(\Sigma^{-1}\Delta\Sigma+\lambda\Sigma^{-2}\right)^{-1}\Sigma^{-1} \\ &= \left(I+\lambda\Sigma^{-2}+E\right)^{-1}\Sigma^{-1}.\end{aligned}$$

It is now enough to get a lower bound for the smallest singular value of $I+\lambda\Sigma^{-2}+E$. We will compare the singular values of this matrix to the singular values of $I+\lambda\Sigma^{-2}$. From standard perturbation theory,

$$(1-\epsilon)+\frac{\lambda}{\sigma_i^2} \le \sigma_i\left(I+\lambda\Sigma^{-2}+E\right) \le (1+\epsilon)+\frac{\lambda}{\sigma_i^2} ,$$

and hence using $\sigma_{\max}/\sigma_{\min}=\kappa_A$,

$$\begin{aligned}\left\|\left(I+\lambda\Sigma^{-2}+E\right)^{-1}\Sigma^{-1}\right\|_2 &\le \frac{\sigma_{\max}^2}{\left((1-\epsilon)\sigma_{\max}^2+\lambda\right)\sigma_{\min}} \\ &= \frac{\kappa_A\sigma_{\max}}{(1-\epsilon)\sigma_{\max}^2+\lambda} \\ &\le \frac{2\kappa_A}{\sigma_{\max}} .\end{aligned}$$

In the above we used the fact that $\epsilon\le 1/2$, which implies that $(1-\epsilon)+\lambda/\sigma_{\max}^2\ge 1/2$. Combining the above, we get a bound for (18).

$$\left|\beta^TU^{\perp^T}SU\Sigma(\Delta+\lambda I)^{-1}V^Ty\right| \le \frac{2\epsilon\kappa_A}{\sigma_{\max}}\left\|\beta\right\|_2\left\|y\right\|_2. \quad (21)$$

In order to prove Theorem 1 for the case $\lambda=0$, notice that equation (20) becomes zero. For the case $\lambda>0$, notice that the denominator $\sigma_{\min}^2+(1-\epsilon)\lambda$ in (20) is always larger than $(1-\epsilon)\lambda$, and thus we can upper bound the prefactor in (20) by $2\epsilon\kappa_A$ (since $\epsilon\le 1/2$). Additionally, using $\left\|\alpha^TV\right\|_2=\left\|\alpha\right\|_2$ (since V is a full-rank orthonormal matrix),

$$\left\|\alpha^TV\left(I+\lambda\Sigma^{-2}\right)^{-1}\right\|_2 \le \left\|\alpha\right\|_2\left\|\left(I+\lambda\Sigma^{-2}\right)^{-1}\right\|_2 \le \left\|\alpha\right\|_2 .$$

The last inequality follows from the fact that the singular values of $I+\lambda\Sigma^{-2}$ are equal to $1+\lambda/\sigma_i^2$; thus, the spectral norm of its inverse is at most one. Combining the above with (20) and using $\left\|V^Ty\right\|_2=\left\|y\right\|_2$ concludes the proof of Theorem 1.

The proof of Theorem 2 follows by noticing that if β is all-zeros the right-hand side of (21) is zero. Since $\lambda>0$, we can use the aforementioned argument to bound the prefactor in (20) by $2\epsilon\kappa_A$, which concludes the proof.

6. EMPIRICAL RESULTS

In this section, we describe our empirical evaluations on three datasets: TechTC-100 [9]; 20-Newsgroups [1,2,21]; and Reuters RCV2 [23]. We compare several sampling-based feature selection strategies to feature selection methods commonly used in Information Retrieval (IR). Our aim is to compare classification results after performing feature selection with classification results from the original problem.

6.1 The Datasets

Table 1 summarizes the structure of the three datasets. The TechTC-100 family [9,16] consists of 100 datasets, each

Table 1: Datasets

Name	Classes	Terms	Train	Test
TechTC-100	2	20K	100×120	100×30
20-Newsgroups	20	62k	15k	4k
Reuters-rcv2	103	47k	23k	10k

having roughly 150 documents evenly spread across two classes. The categorization difficulty of these datasets, as measured in [9] (using the baseline SVM accuracy), is uniformly distributed between 0.6 and 1.0. Each dataset is stemmed, normalized according to SMART-*ltc* [29], and then split into four different test-train splits. The ratio of test to train documents we used is 1 : 4.

The 20-Newsgroups dataset [1, 2, 21], which consists of postings from 20 Usenet newsgroups, is well used in the IR literature. The dataset consists of 20 classes, each corresponding to a newsgroup, containing almost an equal number of documents. We used the document vectors provided by Rennie *et al.* [1], who applied the usual stemming and *ltc* normalization to this dataset, and split it into ten test-train splits. We employ only the first five splits for our empirical evaluations.

The last dataset is a subset of Reuters-RCV2 [23], that contains news-feeds from Reuters. We considered only the 103 topic codes as the classes. The class structure in Reuters is hierarchical, and as a result the sizes of the classes are highly skewed, with the 21 non-leaf classes accounting for 79% of the total number of documents. We considered all 103 topics as separate classes. We use both the *ltc*-normalized term-document matrix and the one test-train

split provided by Lewis *et al.* [23] for this dataset. For efficiency purposes, instead of using all 800K test documents for classification, we randomly select 10K test documents and report results on these.

Finally, for each of these datasets, we used our SRLS classifier with feature selection in a simple *one-vs-all* format.

6.2 Feature Selection Strategies

We investigate the following three sampling-based feature selection strategies. Since these strategies are randomized, we need only specify the probability distribution $\{p_i\}_{i=1}^n$ that is passed to the SRLS ALGORITHM.

• **Subspace Sampling (SS).** The probability of choosing each feature is proportional to the length squared of the corresponding row of the matrix U_k consisting of the top k left singular vectors of A, i.e.,

$$p_i = \left\|U_{k(i)}\right\|_2^2 / k. \tag{22}$$

(Thus, for our empirical evaluation we generalize Equation (9) to permit k to be a parameter.)

• **Weight-based Sampling (WS).** The probability of choosing each feature is proportional to the length squared of the corresponding row of the matrix A, i.e.,

$$p_i = \left\|A_{(i)}\right\|_2^2 / \left\|A\right\|_F^2. \tag{23}$$

• **Uniform Sampling (US)**. The probability of choosing each feature is equal, i.e., $p_i = 1/n$, for all $i = 1, \ldots, n$.

We compare the performance of these three strategies with that of the following two deterministic feature selection methods that are well-known in the IR literature.

• **Document Frequency (DF)**. The document frequency of a term is the number of training documents in which it appears.

• **Information Gain (IG)**. The IG feature selection method is based on a notion of the amount of information the presence or absence of a particular term contains about the class of the document [39]. It is measured as follows:

$$IG(t) = \sum_{c \in \{c_k, \bar{c_k}\}} \sum_{t \in \{t_i, \bar{t_i}\}} P(t,c) \cdot \log \frac{P(t,c)}{P(t)P(c)} \tag{24}$$

From its definition, it is clear that the IG strategy is a *supervised* strategy. That is, it uses the document labels in its choice of features to retain. In contrast, our sampling-based strategies, as well as the DF strategy, are *unsupervised*.

6.3 Results

6.3.1 Aggregation of Results

We investigate precision, recall, and the micro- and macro-averaged F1 measures aggregated over all classes. We are interested in comparing the performance of the classifier on the subsampled instance to the performance on the instance with the full feature set. Thus, we mainly report relative performances of the subsampled instances to the original instance. We first describe how the aggregation of the relative performances were done, taking the 20-Newsgroups dataset as an example.

We considered five test-train splits for the 20-Newsgroups dataset. For each split, $i = 1, \ldots, 5$, we obtained the optimal (micro-averaged F1) performance $\text{MIF}_{\max}(i)$ of the classifier on the full feature set by varying the regularization parameter λ. This procedure essentially determined

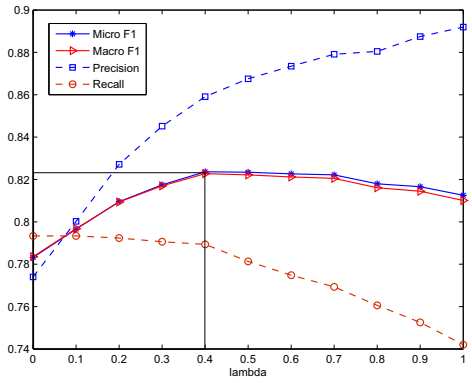
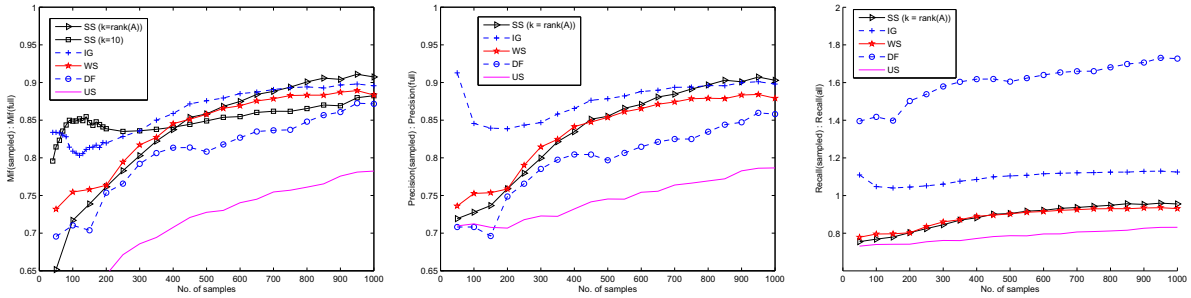


Figure 2: Performance of various values of λ for the first split of the 20-Newsgroup dataset. The optimal Micro-averaged and Macro-averaged F1 values occur at $\lambda = 0.4$ for this split.

both the baseline performance $\text{MIF}_{\max}(i)$ and the optimal value of the regularization parameter $\lambda_{\max}(i)$ for split i. Figure 2 plots the micro- and macro-averaged F1, average precision and average recall as a function of λ for one of the splits and shows the choice of the optimal value $\lambda_{\max}$ for this split. Next, for each of the randomized sampling strategies; subspace, weighted, and uniform, and for each expected sample size r, we collected the aggregated performances over five different samples in $\text{MIF}_s(i,r)$, $\text{MIF}_w(i,r)$ and $\text{MIF}_u(i,r)$, respectively. For the deterministic feature selection strategies of IG and DF, the performance values $\text{MIF}_{ig}(i,r)$ and $\text{MIF}_{df}(i,r)$ were obtained using one run for each sample size, as the sample size r is the actual number of features chosen. We then computed the relative performances for each feature selection strategy for this split; e.g., $\text{RelMIF}_s(i,r) = \text{MIF}_s(i,r)/\text{MIF}_{\max}(i)$. The relative performance $\text{RelMIF}(r)$ curves for each strategy were then obtained by averaging over all the splits. We followed the same strategy of averaging the relative performance RelMIF for the TechTC family, using four test-train splits for each dataset. For ease of exposition, we also average the RelMIF curves across the 100 different datasets. For the Reuters dataset, we used only one test-train split (that of Lewis *et al.* [23]).

6.3.2 Results for TechTC-100

For the TechTC-100 family, Figures 3(a), 3(b) and 3(c) demonstrate the performances of the various feature selection strategies. As mentioned earlier, we aggregated the RelMIF and the relative precision and recall values over all 100 of the datasets in the family. Figure 3(a) presents the RelMIF performance of all the feature selection strategies. All the selection strategies except document frequency (DF) and uniform sampling (US) achieve 85% of the original (involving no sampling) micro-averaged F1 performance with only 500 out of the (roughly) 20K original features. In general, the subspace sampling (SS) and information gain (IG) strategies perform best, followed closely by weighted sampling (WS). For very aggressive feature selection (choosing less than 0.1% of the original features), SS based on $k = 10$



(a) Micro-averaged F1 ratios. (b) Precision ratios. (c) Recall ratios.

Figure 3: Performance of various sampling methods on the TechTC-100 dataset.

singular vectors actually dominates both IG and the full-rank SS ($k = \text{rank}(A)$). Figures 3(b) and 3(c) present the precision and recall ratios averaged over all 100 datasets. The precision plot closely follows the micro-averaged performance. The recall performance is very different, with DF scoring much higher than all the other strategies.

In order to understand better the behavior of SS, e.g., why it does relatively well at very aggressive levels of feature selection and why other methods perform similar to it at less aggressive levels, we considered its behavior in more detail. Figure 4(a) demonstrates the variability of the IG and SS strategies across the 100 datasets of the TechTC family. We set the expected sample size parameter r to 1000, and sorted the 100 datasets according to the RelMIF performance of the IG strategy. The two curves show the performance of the IG and the SS strategies according to this ordering. The performance of the SS strategy seems uncorrelated with the performance of the IG-based feature selection. Also, the relative performance of the IG strategy varies from 0.6 to 1.1 whereas that of SS varies only from 0.8 to 1. The two horizontal lines represent the aggregated performances for the two methods over all the 100 datasets, and they correspond to the points plotted on Figure 3(a) at $r = 1000$. The aggregated performance of the SS method is marginally better than that of IG at this sample size. Since roughly half of the datasets have worse IG perfomance than the average, it follows from the figure that on roughly half of these 100 datasets, SS performs better than IG.

We also investigate the effect of the choice of the number of singular vectors k used by SS. Figure 4(b) plots the relative micro-averaged F1 of SS for various values of k. At aggressive levels of feature selection, smaller values of k give a better performance whereas for higher number of features, choosing k to be equal to the rank of the training matrix A seems to be the optimal strategy. As expected, using all the singular vectors (i.e. k equals the rank of the training matrix) and using the top singular vectors that capture 90% of the Frobenius norm behave similarly.

The performance plots in Figure 3(a) show that both of the weight-based strategies WS and DF perform similarly to SS when k is chosen to be close to the rank of the matrix A. Insight into why this may be the case can be obtained by examining the distance between the probability distributions as a function of k. Given two probability distributions $\bar{p} = \{p_1 \dots p_n\}$ and $\bar{q} = \{q_1, \dots, q_n\}$, a useful notion of distance between them is the Hellinger distance $H(\bar{p}, \bar{q})$:

$$H(\bar{p}, \bar{q}) = \sqrt{\sum_{i=1}^{n} (\sqrt{p_i} - \sqrt{q_i})^2}.$$

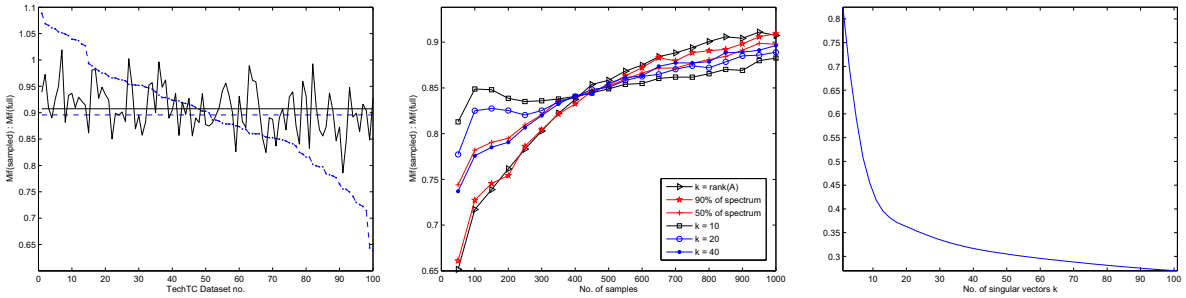
Figure 4(c) plots the Hellinger distance between the probability distribution of the WS (weighted sampling) strategy and the probability distribution of SS for various k, ranging from 1 to 100. In terms of Hellinger distance, WS is closer to SS for higher values of k. Thus, for less aggressive levels of feature selection, i.e., when the optimal strategy is to choose k to be as close to the rank of A as possible, the weight-based selection methods (WS, which has a similar flavor to DF) can serve as an efficient substitute for the SS strategy. This observation is in fact corroborated by the performance plots in Figure 4(c).

6.3.3 *Results for 20-Newsgroups*

Figure 5(a) plots the relative micro-averaged F1 performances RelMIF against the expected number of features chosen by each selection strategy for the 20-Newsgroups dataset.

For the SS strategy, we employed either $k = 1500$ singular vectors, which captures 35% of the total Frobenius norm, or $k = 100$, which captures 15%. Both the SS (for $k = 1500$) and IG strategies achieve about 90% of the micro-averaged F1 performance of the full feature-set with roughly 5K of the total 60K features. However, the classification performance of this dataset seems to degrade rapidly at aggressive levels of sampling. The IG-based strategy dominates the performance, being particulary better than the others at the aggressive levels. We see the same effect of SS with $k = 100$ being better among all unsupervised methods for aggressive selection. For this dataset, though, the effect is much less pronounced than it is for TechTC. In general, SS with $k = 1500$ strategy outperforms the other unsupervised feature selection strategies; however, it is only marginally better than the WS and DF methods. As expected, uniformly random feature selection falls far behind rest.

The relative precision and recall plots (not presented) show that the precision does not increase after selecting ca. 3000 features, while the recall steadily increases with increase in the number of selected features. This is presumably because although the terms chosen at this point are discriminative amongst the classes, there are documents in



(a) MIF performances of IG and SS on the TechTC dataset sorted by IG's performance. Horizontal lines are the average performances.

(b) MIF performance of SS on TechTC based on k singular vectors.

(c) Hellinger distance between the probability scores of WS and SS using k singular vectors.

Figure 4: Performance analysis for the TechTC-100 dataset.

the test set that do not contain these terms and thus affect the average recall performance.

6.3.4 *Results for Reuters*

Lastly, Figures 5(b) and 5(c) summarize the performance on the Reuters dataset. For SS strategy, we use either $k = 1500$, capturing 30% of the Frobenius norm, or $k = 100$ capturing only 12%. Under feature selection, the performance of this dataset actually improves marginally over the full set of features. Since the Reuters dataset has a wide skew in the sizes of different classes, we present the relative performance both for the micro-averaged and the macro-averaged F1 measures. As with 20-Newsgroups, the IG-based feature selection strategy performs marginally better than the others. In fact, for this dataset, the DF selection strategy also slightly outperforms the subspace-based methods.

7. CONCLUSIONS

Several directions present themselves for future work. First, we expect that our analysis is not tight in that we have permitted ourselves to sample enough such that Δ has an inverse. One might expect that we only need to sample enough to "capture" the part of the space that is not "cut off" by the regularization parameter λ, and a more refined analysis might demonstrate this. Second, we have based our analysis on recent work in the theory of algorithms for approximating the ℓ_2 regression problem [11], but similar methods also apply to the ℓ_p regression problem, for all $p \in [1, \infty)$ [8]. One might expect that by considering $p = 1$ we can apply our methods to the SVMs, which would be of interest due to the ubiquity of SVM-based classifiers in large-scale text analysis. For example, although we use matrix perturbation theory [32] to establish the robustness of certain subspaces to the feature selection process, if one can show that the relevant convex sets are similarly robust then our methodology should apply to SVM classification.

8. ACKNOWLEDGMENTS

We would like to thank Christos Faloutsos for helpful discussions during the initial stages of this project and Evgeniy Gabrilovich for pointing us to [9, 16].

9. REFERENCES

[1] 20 Newsgroups Dataset. J. Rennie. `http://people.csail.mit.edu/jrennie/20Newsgroups/`.

[2] 20 Newsgroups Dataset. UCI KDD Archive. `http://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups.html`.

[3] D.K. Agarwal. Shrinkage estimator generalizations of proximal support vector machines. In *Proceedings of the 8th Annual ACM SIGKDD Conference*, pages 173–182, 2002.

[4] A.L. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97:245–271, 1997.

[5] A.L. Blum and R.L. Rivest. Training a 3-node neural network is NP-complete. *Neural Networks*, 5:117–127, 1992.

[6] M. Charikar, V. Guruswami, R. Kumar, S. Rajagopalan, and A. Sahai. Combinatorial feature selection problems. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, pages 631–642, 2000.

[7] A. Das and D. Kempe. Algorithms for subset selection in linear regression. *Manuscript.*

[8] A. Dasgupta, P. Drineas, B. Harb, R. Kumar, and M. W. Mahoney. Sampling algorithms and coresets for ℓ_p regression. In *Manuscript submitted for publication.*

[9] D. Davidov, E. Gabrilovich, and S. Markovitch. Parameterized generation of labeled datasets for text categorization based on a hierarchical directory. In *Proceedings of the 27th Annual International ACM SIGIR Conference*, pages 250–257, 2004.

[10] J. Demmel and K. Veselić. Jacobi's method is more accurate than QR. *SIAM Journal on Matrix Analysis and Applications*, 13:1204–1245, 1992.

[11] P. Drineas, M.W. Mahoney, and S. Muthukrishnan. Sampling algorithms for ℓ_2 regression and applications. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1127–1136, 2006.

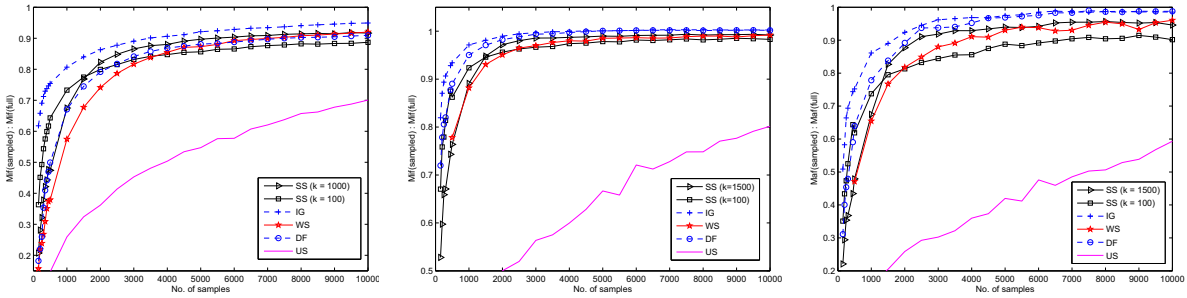
[12] T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. *Advances in Computational Mathematics*, 13(1):1–50, 1999.

[13] G. Forman. An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research*, 3:1289–1305, 2003.

[14] D. Fragoudis, D. Meretakis, and S. Likothanassis. Integrating feature and instance selection for text classification. In *Proceedings of the 8th Annual ACM SIGKDD Conference*, pages 501–506, 2002.

[15] G. Fung and O.L. Mangasarian. Proximal support vector machine classifiers. In *Proceedings of the 7th Annual ACM SIGKDD Conference*, pages 77–86, 2001.

[16] E. Gabrilovich and S. Markovitch. Text categorization with many redundant features: using aggressive feature selection



(a) 20-NG: Micro-averaged F1. (b) Reuters: Micro-averaged F1. (c) Reuters: Macro-averaged F1.

Figure 5: Performance of various sampling methods on the 20-Newsgroups and Reuters datasets.

to make SVMs competitive with C4.5. In *Proceedings of the 21th International Conference on Machine Learning*, pages 41–48, 2004.

[17] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[18] T. Joachims. Text categorization with suport vector machines: Learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning*, pages 137–142, 1998.

[19] H. Kim, P. Howland, and H. Park. Dimension reduction in text classification with support vector machines. *Journal of Machine Learning Research*, 6:37–53, 2005.

[20] R. Kohavi and G.H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97:273–324, 1997.

[21] K. Lang. Newsweeder: Learning to filter netnews. In *Proceedings of the 12th International Conference on Machine Learning*, pages 331–339, 1995.

[22] E. Leopold and J. Kindermann. Text categorization with support vector machines. How to represent texts in input space? *Machine Learning*, 46:423–444, 2002.

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

[24] D. Mladenic and M. Grobelnik. Feature selection for unbalanced class distribution and Naive Bayes. In *Proceedings of the 16th International Conference on Machine Learning*, pages 258–267, 1999.

[25] T. Poggio and S. Smale. The mathematics of learning: Dealing with data. *Notices of the AMS*, 50(5):537–544, May 2003.

[26] R. Rifkin. *Everything Old Is New Again: A Fresh Look at Historical Approaches in Machine Learning*. PhD thesis, Massachusetts Institute of Technology, 2002.

[27] R. Rifkin, G. Yeo, and T. Poggio. Regularized least-squares classification. In J.A.K. Suykens, G. Horvath, S. Basu, C. Micchelli, and J. Vandewalle, editors, *Advances in Learning Theory: Methods, Models and Applications*, NATO Science Series III: Computer and Systems Sciences, pages 131–154. VIOS Press, 2003.

[28] M. Rudelson and R. Vershynin. Sampling from large matrices: an approach through geometric functional analysis. *Manuscript*.

[29] G. Salton. The smart document retrieval project. In *Proceedings of the 14th Annual International ACM SIGIR Conference*, pages 356–358, 1991.

[30] Bernhard Schölkopf, Ralf Herbrich, Alex J. Smola, and Robert C. Williamson. A generalized representer theorem. In *Proceedings of the 14th Annual Conference on Computational Learning Theory (COLT 2001) and the 5th European Conference on Computational Learning Theory (EuroCOLT 2001)*, pages 416–426, 2001.

[31] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[32] G.W. Stewart and J.G. Sun. *Matrix Perturbation Theory*. Academic Press, New York, 1990.

[33] J.A.K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

[34] A.N. Tikhonov and V.Y. Arsenin. *Solutions of Ill-Posed Problems*. W.H. Winston, Washington, D.C., 1977.

[35] V.N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

[36] Y. Yang. Noise reduction in a statistical approach to text categorization. In *Proceedings of the 18th Annual International ACM SIGIR Conference*, pages 256–263, 1995.

[37] Y. Yang. Sampling strategies and learning efficiency in text categorization. In *AAAI Spring Symposium on Machine Learning in Information Access*, pages 88–95, 1996.

[38] Y. Yang and X. Liu. A re-examination of text categorization methods. In *Proceedings of the 22nd Annual International ACM SIGIR Conference*, pages 42–49, 1999.

[39] Y. Yang and J.O. Pedersen. A comparative study on feature selection in text categorization. In *Proceedings of the 14th International Conference on Machine Learning*, pages 412–420, 1997.

[40] J. Zhang and Y. Yang. Robustness of regularized linear classification methods in text categorization. In *Proceedings of the 26th Annual International ACM SIGIR Conference*, pages 190–197, 2003.

[41] P. Zhang and J. Peng. SVM vs regularized least squares classification. In *Proceedings of the 17th International Conference on Pattern Recognition*, pages 176–179, 2004.

[42] T. Zhang and F.J. Oles. Text categorization based on regularized linear classification methods. *Information Retrieval*, 4(1):5–31, 2001.