# Scaling Semi-supervised Naive Bayes with Feature Marginals

**Michael R. Lucas** and **Doug Downey**
Northwestern University
2133 Sheridan Road
Evanston, IL 60208
mlucas@u.northwestern.edu
ddowney@eecs.northwestern.edu

## Abstract

Semi-supervised learning (SSL) methods augment standard machine learning (ML) techniques to leverage unlabeled data. SSL techniques are often effective in text classification, where labeled data is scarce but large unlabeled corpora are readily available. However, existing SSL techniques typically require multiple passes over the entirety of the unlabeled data, meaning the techniques are not applicable to large corpora being produced today.

In this paper, we show that improving marginal word frequency estimates using unlabeled data can enable semi-supervised text classification that scales to massive unlabeled data sets. We present a novel learning algorithm, which optimizes a Naive Bayes model to accord with statistics calculated from the unlabeled corpus. In experiments with text topic classification and sentiment analysis, we show that our method is both more scalable and more accurate than SSL techniques from previous work.

## 1 Introduction

Semi-supervised Learning (SSL) is a Machine Learning (ML) approach that utilizes large amounts of unlabeled data, combined with a smaller amount of labeled data, to learn a target function (Zhu, 2006; Chapelle et al., 2006). SSL is motivated by a simple reality: the amount of available machine-readable data is exploding, while human capacity for hand-labeling data for any given ML task remains relatively constant. Experiments in text classification and other domains have demonstrated that by leveraging unlabeled data, SSL techniques improve machine learning performance when human input is limited (e.g., (Nigam et al., 2000; Mann and McCallum, 2010)).

However, current SSL techniques have scalability limitations. Typically, for each target concept to be learned, a semi-supervised classifier is trained using iterative techniques that execute multiple passes over the unlabeled data (e.g., Expectation-Maximization (Nigam et al., 2000) or Label Propagation (Zhu and Ghahramani, 2002)). This is problematic for text classification over large unlabeled corpora like the Web: new target concepts (new tasks and new topics of interest) arise frequently, and performing even a single pass over a large corpus for each new target concept is intractable.

In this paper, we present a new SSL text classification approach that scales to large corpora. Instead of utilizing unlabeled examples directly for each given target concept, our approach is to precompute a small set of *statistics* over the unlabeled data in advance. Then, for a given target class and labeled data set, we utilize the statistics to improve a classifier.

Specifically, we introduce a method that extends Multinomial Naive Bayes (MNB) to leverage marginal probability statistics $P(w)$ of each word $w$, computed over the unlabeled data. The marginal statistics are used as a constraint to improve the class-conditional probability estimates $P(w|+)$ and $P(w|-)$ for the positive and negative classes, which are often noisy when estimated over sparse labeled data sets. We refer to the technique as MNB with Frequency Marginals (MNB-FM).

In experiments with large unlabeled data sets and sparse labeled data, we find that MNB-FM is both faster and more accurate on average than standard SSL methods from previous work, including Label Propagation, MNB with Expectation-Maximization,, and the recent Semi-supervised Frequency Estimate (SFE) algorithm (Su et al., 2011). We also analyze how MNB-



*Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 343–351,
Sofia, Bulgaria, August 4-9 2013. 

FM improves accuracy, and find that surprisingly MNB-FM is especially useful for improving class-conditional probability estimates for words that never occur in the training set.

The paper proceeds as follows. We formally define the task in Section 2. Our algorithm is defined in Section 3. We present experimental results in Section 4, and analysis in Section 5. We discuss related work in Section 6 and conclude in Section 7 with a discussion of future work.

## 2 Problem Definition

We consider a semi-supervised classification task, in which the goal is to produce a mapping from an instance space $\mathcal{X}$ consisting of $T$-tuples of non-negative integer-valued features $\mathbf{w} = (w_1, \ldots, w_T)$, to a binary output space $\mathcal{Y} = \{-, +\}$. In particular, our experiments will focus on the case in which the $w_i$'s represent word counts in a given document, in a corpus of vocabulary size $T$.

We assume the following inputs:

- A set of zero or more *labeled* documents $D_L = \{(\mathbf{w}^d, y^d) | d = 1, \ldots, n\}$, drawn i.i.d. from a distribution $P(\mathbf{w}, y)$ for $\mathbf{w} \in \mathcal{X}$ and $y \in \mathcal{Y}$.
- A large set of *unlabeled* documents $D_U = \{(\mathbf{w}^d) | d = n+1, \ldots, n+u\}$ drawn from the marginal distribution $P(\mathbf{w}) = \sum_y P(\mathbf{w}, y)$.

The goal of the task is to output a classifer $f : \mathcal{X} \to \mathcal{Y}$ that performs well in predicting the classes of given unlabeled documents. The metrics of evaluation we focus on in our experiments are detailed in Section 4.

Our semi-supervised technique utilizes statistics computed over the labeled corpus, denoted as follows. We use $N_w^+$ to denote the sum of the occurrences of word $w$ over all documents in the positive class in the labeled data $D_L$. Also, let $N^+ = \sum_{w \in D_L}^{n} N_w^+$ be the sum value of all word counts in the labeled positive documents. The count of the remaining words in the positive documents is represented as $N_{\neg w}^+ = N^+ - N_w^+$. The quantities $N^-$, $N_w^-$, and $N_{\neg w}^-$ are defined similarly for the negative class.

## 3 MNB with Feature Marginals

We now introduce our algorithm, which scalably utilizes large unlabeled data stores for classification tasks. The technique builds upon the multinomial Naive Bayes model, and is denoted as MNB with Feature Marginals (MNB-FM).

### 3.1 MNB-FM Method

In the text classification setting , each feature value $w^d$ represents count of observations of word $w$ in document $d$. MNB makes the simplifying assumption that word occurrences are conditionally independent of each other given the class ($+$ or $-$) of the example. Formally, let the probability $P(w|+)$ of the $w$ in the positive class be denoted as $\theta_w^+$. Let $P(+)$ denote the prior probability that a document is of the positive class, and $P(-) = 1 - P(+)$ the prior for the negative class. Then MNB represents the class probability of an example as:

$$P(+|d) = \frac{\prod_{w \in d} (\theta_w^+)^{w^d} P(+)}{\prod_{w \in d} (\theta_w^-)^{w^d} P(-) + \prod_{w \in d} (\theta_w^+)^{w^d} P(+)} \tag{1}$$

MNB estimates the parameters $\theta_w^+$ from the corresponding counts in the training set. The maximum-likelihood estimate of $\theta_w^+$ is $N_w^+ / N^+$, and to prevent zero-probability estimates we employ "add-1" smoothing (typical in MNB) to obtain the estimate:

$$\theta_w^+ = \frac{N_w^+ + 1}{N^+ + |T|}.$$

After MNB calculates $\theta_w^+$ and $\theta_w^-$ from the training set for each feature in the feature space, it can then classify test examples using Equation 1.

MNB-FM attempts to improve MNB's estimates of $\theta_w^+$ and $\theta_w^-$, using statistics computed over the unlabeled data. Formally, MNB-FM leverages the equality:

$$P(w) = \theta_w^+ P_t(+) + \theta_w^- P_t(-) \tag{2}$$

The left-hand-side of Equation 2, $P(w)$, represents the probability that a given randomly drawn token from the unlabeled data happens to be the word $w$. We write $P_t(+)$ to denote the probability that a randomly drawn $token$ (i.e. a word occurrence) from the corpus comes from the positive class. Note that $P_t(+)$ can differ from $P(+)$, the prior probability that a document is positive, due to variations in document length. $P_t(-)$ is defined similarly for the negative class. MNB-FM is motivated by the insight that the left-hand-side of

Equation 2 can be estimated in advance, without knowledge of the target class, simply by counting the number of tokens of each word in the unlabeled data.

MNB-FM then uses this improved estimate of $P(w)$ as a constraint to improve the MNB parameters on the right-hand-side of Equation 2. We note that $P_t(+)$ and $P_t(-)$, even for a small training set, can typically be estimated reliably— *every* token in the training data serves as an observation of these quantities. However, for large and sparse feature spaces common in settings like text classification, many features occur in only a small fraction of examples—meaning $\theta_w^+$ and $\theta_w^-$ must be estimated from only a handful of observations. MNB-FM attempts to improve the noisy estimates $\theta_w^+$ and $\theta_w^-$ utilizing the robust estimate for $P(w)$ computed over unlabeled data.

Specifically, MNB-FM proceeds by assuming the MLEs for $P(w)$ (computed over unlabeled data), $P_t(+)$, and $P_t(-)$ are correct, and re-estimates $\theta_w^+$ and $\theta_w^-$ under the constraint in Equation 2.

First, the maximum likelihood estimates of $\theta_w^+$ and $\theta_w^-$ given the training data $D_L$ are:

$$\begin{aligned}
&\arg\max_{\theta_w^+,\theta_w^-} P(D_L|\theta_w^+,\theta_w^-) \\
&\quad = \arg\max_{\theta_w^+,\theta_w^-} \theta_w^{+(N_w^+)}(1-\theta_w^+)^{(N_{\neg w}^+)} \\
&\qquad\qquad \theta_w^{-(N_w^-)}(1-\theta_w^-)^{(N_{\neg w}^-)} \\
&\quad = \arg\max_{\theta_w^+,\theta_w^-} N_w^+ \ln(\theta_w^+) + N_{\neg w}^+ \ln(1-\theta_w^+) + \\
&\qquad\qquad N_w^- \ln(\theta_w^-) + N_{\neg w}^- \ln(1-\theta_w^-)
\end{aligned} \tag{3}$$

We can rewrite the constraint in Equation 2 as:

$$\theta_w^- = K - \theta_w^+ L$$

where for compactness we represent:

$$K = \frac{P(w)}{P_t(-)}; L = \frac{P_t(+)}{P_t(-)}.$$

Substituting the constraint into Equation 3 shows that we wish to choose $\theta_w^+$ as:

$$\begin{aligned}
&\arg\max_{\theta_w^+} N_w^+ \ln(\theta_w^+) + N_{\neg w}^+ \ln(1-\theta_w^+) + \\
&\qquad N_w^- \ln(K - L\theta_w^+) + N_{\neg w}^- \ln(1-K+L\theta_w^+)
\end{aligned}$$

The optimal values for $\theta_w^+$ are thus located at the solutions of:

$$0 = \frac{N_w^+}{\theta_w^+} + \frac{N_{\neg w}^+}{\theta_w^+ - 1} + \frac{LN_w^-}{L\theta_w^+ - K} + \frac{LN_{\neg w}^-}{L\theta_w^+ - K + 1}$$

Both $\theta_w^+$ and $\theta_w^-$ are constrained to valid probabilities in [0,1] when $\theta_w^+ \in [0, \frac{K}{L}]$. If $N_{\neg w}^+$ and $N_w^-$ have non-zero counts, vertical asymptotes exist at 0 and $\frac{K}{L}$ and guarantee a solution in this range. Otherwise, a valid solution may not exist. In that case, we default to the add-1 Smoothing estimates used by MNB. Finally, after optimizing the values $\theta_w^+$ and $\theta_w^-$ for each word $w$ as described above, we normalize the estimates to obtain valid conditional probability distributions, i.e. with $\sum_w \theta_w^+ = \sum_w \theta_w^- = 1$

### 3.2 MNB-FM Example

The following concrete example illustrates how MNB-FM can improve MNB parameters using the statistic $P(w)$ computed over unlabeled data. The example comes from the Reuters Aptemod text classification task addressed in Section 4, using bag-of-words features for the Earnings class. In one experiment with 10 labeled training examples, we observed 5 positive and 5 negative examples, with the word "resources" occurring three times in the set (once in the positive class, twice in the negative class).

MNB uses add-1 smoothing to estimate the conditional probability of the word "resources" in each class as $\theta_w^+ = \frac{1+1}{216+33504} = 5.93\text{e-}5$, and $\theta_w^- = \frac{2+1}{547+33504} = 8.81\text{e-}5$. Thus, $\frac{\theta_w^+}{\theta_w^-} = 0.673$ implying that "resources" is a negative indicator of the Earnings class. However, this estimate is inaccurate. In fact, over the *full* dataset, the parameter values we observe are $\theta_w^+ = \frac{93}{168549} = 5.70\text{e-}4$ and $\theta_w^- = \frac{263}{564717} = 4.65\text{e-}4$, with a ratio of $\frac{\theta_w^+}{\theta_w^-} = 1.223$. Thus, in actuality, the word "resources" is a mild *positive* indicator of the Earnings class. Yet because MNB estimates its parameters from only the sparse training data, it can be inaccurate.

The optimization in MNB-FM seeks to accord its parameter estimates with the feature frequency, computed from unlabeled data, of $P(w) = 4.89\text{e-}4$. We see that compared with $P(w)$, the $\theta_w^+$ and $\theta_w^-$ that MNB estimates from the training data are both too low by almost an order of magnitude. Further, the maximum likelihood estimate for $\theta_w^-$ (based on an occurrence count of 2 out of 547 observations) is somewhat more reliable than that for $\theta_w^+$ (1 of 216 observations). As a result, $\theta_w^+$ is adjusted upward relatively *more* than $\theta_w^-$ via MNB-FM's constrained ML estimation. MNB-FM returns $\theta_w^+ = 6.52\text{e-}5$ and $\theta_w^- = 6.04\text{e-}5$. The ratio

$\frac{\theta_w^+}{\theta_w^-}$ is 1.079, meaning MNB-FM correctly identifies the word "resources" as an indicator of the positive class.

The above example illustrates how MNB-FM can leverage frequency marginal statistics computed over unlabeled data to improve MNB's conditional probability estimates. We analyze how frequently MNB-FM succeeds in improving MNB's estimates in practice, and the resulting impact on classification accuracy, below.

## 4 Experiments

In this section, we describe our experiments quantifying the accuracy and scalability of our proposed technique. Across multiple domains, we find that MNB-FM outperforms a variety of approaches from previous work.

### 4.1 Data Sets

We evaluate on two text classification tasks: topic classification, and sentiment detection. In topic classification, the task is to determine whether a test document belongs to a specified topic. We train a classifier separately (i.e., in a binary classification setting) for each topic and measure classification performance for each class individually.

The sentiment detection task is to determine whether a document is written with a positive or negative sentiment. In our case, the goal is to determine if the given text belongs to a positive review of a product.

#### 4.1.1 RCV1

The Reuters RCV1 corpus is a standard large corpus used for topic classification evaluations (Lewis et al., 2004). It includes 804,414 documents with several nested target classes. We consider the 5 largest base classes after punctuation and stopwords were removed. The vocabulary consisted of 288,062 unique words, and the total number of tokens in the data set was 99,702,278. Details of the classes can be found in Table 1.

#### 4.1.2 Reuters Aptemod

While MNB-FM is designed to improve the scalability of SSL to large corpora, some of the comparison methods from previous work were not tractable on the large topic classification data set RCV1. To evaluate these methods, we also experimented with the Reuters ApteMod dataset (Yang and Liu, 1999), consisting of 10,788 documents belonging to 90 classes. We consider the 10 most frequent classes, with varying degrees of positive/negative skew. Punctuation and stopwords were removed during preprocessing. The Aptemod data set contained 33,504 unique words and a total of 733,266 word tokens. Details of the classes can be found in Table 2.

| Class | # Positive |
|---|---|
| CCAT | 381327 (47.40%) |
| GCAT | 239267 (29.74%) |
| MCAT | 204820 (25.46%) |
| ECAT | 119920 (14.91%) |
| GPOL | 56878 (7.07%) |

Table 1: RCV1 dataset details

| Class | # Positive |
|---|---|
| Earnings | 3964 (36.7%) |
| Acquisitions | 2369 (22.0%) |
| Foreign | 717 (6.6%) |
| Grain | 582 (5.4%) |
| Crude | 578 (5.4%) |
| Trade | 485 (4.5%) |
| Interest | 478 (4.4%) |
| Shipping | 286 (2.7%) |
| Wheat | 283 (2.6%) |
| Corn | 237 (2.2%) |

Table 2: Aptemod dataset details

#### 4.1.3 Sentiment Classification Data

In the domain of Sentiment Classification, we tested on the Amazon dataset from (Blitzer et al., 2007). Stopwords listed in an included file were ignored for our experiments and we only the considered unigram features. Unlike the two Reuters data sets, each category had a unique set of documents of varying size. For our experiments, we only used the 10 largest categories. Details of the categories can be found in Table 3.

In the Amazon Sentiment Classification data set, the task is to determine whether a review is positive or negative based solely on the reviewer's submitted text. As such, the positive and negative

| Class | # Instances | # Positive | Vocabulary |
|---|---|---|---|
| Music | 124362 | 113997 (91.67%) | 419936 |
| Books | 54337 | 47767 (87.91%) | 220275 |
| Dvd | 46088 | 39563 (85.84%) | 217744 |
| Electronics | 20393 | 15918 (78.06%) | 65535 |
| Kitchen | 18466 | 14595 (79.04%) | 47180 |
| Video | 17389 | 15017 (86.36%) | 106467 |
| Toys | 12636 | 10151 (80.33%) | 37939 |
| Apparel | 8940 | 7642 (85.48%) | 22326 |
| Health | 6507 | 5124 (78.75%) | 24380 |
| Sports | 5358 | 4352 (81.22%) | 24237 |

Table 3: Amazon dataset details

labels are equally relevant. For our metrics, we calculate the scores for both the positive and negative class and report the average of the two (in contrast to the Reuters data sets, in which we only report the scores for the positive class).

### 4.2 Comparison Methods

In addition to Multinomial Naive Bayes (discussed in Section 3), we evaluate against a variety of supervised and semi-supervised techniques from previous work, which provide a representation of the state of the art. Below, we detail the comparison methods that we re-implemented for our experiments.

#### 4.2.1 NB + EM

We implemented a semi-supervised version of Naive Bayes with Expectation Maximization, based on (Nigam et al., 2000). We found that 15 iterations of EM was sufficient to ensure approximate convergence of the parameters.

We also experimented with different weighting factors to assign to the unlabeled data. While performing per-data-split cross-validation was computationally prohibitive for NB+EM, we performed experiments on one class from each data set that revealed weighting unlabeled examples at $1/5$ the weight of a labeled example performed best. We found that our re-implementation of NB+EM slightly outperformed published results on a separate data set (Mann and McCallum, 2010), validating our design choices.

#### 4.2.2 Logistic Regression

We implemented Logistic Regression using L2-Normalization, finding this to outperform L1-Normalized and non-normalized versions. The strength of the normalization was selected for each training data set of each size utilized in our experiments.

The strength of the normalization in the logistic regression required cross-validation, which we limited to 20 values logarithmically spaced between $10^{-4}$ and $10^{4}$. The optimal value was selected based upon the best average F1 score over the 10 folds. We selected a normalization parameter separately for each subset of the training data during experimentation.

#### 4.2.3 Label Propagation

For our large unlabeled data set sizes, we found that a standard Label Propogation (LP) approach, which considers propagating information between all pairs of unlabeled examples, was not tractable. We instead implemented a constrained version of LP for comparison.

In our implementation, we limit the number of edges in the propagation graph. Each node propagates to only to its 10 nearest neighbors, where distance is calculated as the cosine distance between the tf-idf representation of two documents. We found the tf-idf weighting to improve performance over that of simple cosine distance. Propagation was run for 100 iterations or until the entropy dropped below a predetermined threshold, whichever occurred first. Even with these aggressive constraints, Label Propagation was intractable to execute on some of the larger data sets, so we do not report LP results for the RCV1 dataset or for the 5 largest Amazon categories.

#### 4.2.4 SFE

We also re-implemented a version of the recent Semi-supervised Frequency Estimate approach (Su et al., 2011). SFE was found to outperform MNB and NB+EM in previous work. Consistent with our MNB implementation, we use Add-1 Smoothing in our SFE calculations although its use is not specifically mentioned in (Su et al., 2011).

SFE also augments multinomial Naive Bayes with the frequency information $P(w)$, although in a manner distinct from MNB-FM. In particular, SFE uses the equality $P(+|w) = P(+, w)/P(w)$ and estimates the rhs using $P(w)$ computed over all the unlabeled data, rather than using only labeled data as in standard MNB. The primary distinction between MNB-FM and SFE is that SFE adjusts sparse estimates $P(+, w)$ in the same way as non-sparse estimates, whereas MNB-FM is designed to adjust sparse estimates more than non-sparse ones. Further, it can be shown that as $P(w)$ of a word $w$ in the unlabeled data becomes larger than that in the labeled data, SFE's estimate of the ratio $P(w|+)/P(w|-)$ approaches one. Depending on the labeled data, such an estimate can be arbitrarily inaccurate. MNB-FM does not have this limitation.

### 4.3 Results

For each data set, we evaluate on 50 randomly drawn training splits, each comprised of 1,000 randomly selected documents. Each set included at least one positive and one negative document. We

| Data Set | MNB-FM | SFE | MNB | NBEM | LProp | Logist. |
|---|---|---|---|---|---|---|
| Apte (10) | 0.306 | 0.271 | **0.336** | 0.306 | 0.245 | 0.208 |
| Apte (100) | **0.554** | 0.389 | 0.222 | 0.203 | 0.263 | 0.330 |
| Apte (1k) | **0.729** | 0.614 | 0.452 | 0.321 | 0.267 | 0.702 |
| Amzn (10) | **0.542** | 0.524 | 0.508 | 0.475 | 0.470* | 0.499 |
| Amzn (100) | **0.587** | 0.559 | 0.456 | 0.456 | 0.498* | 0.542 |
| Amzn (1k) | 0.687 | 0.611 | 0.465 | 0.455 | 0.539* | **0.713** |
| RCV1 (10) | **0.494** | 0.477 | 0.387 | 0.485 | - | 0.272 |
| RCV1 (100) | **0.677** | 0.613 | 0.337 | 0.470 | - | 0.518 |
| RCV1 (1k) | 0.772 | 0.735 | 0.408 | 0.491 | - | **0.774** |

* Limited to 5 of 10 Amazon categories

Table 4: F1, training size in parentheses

respected the order of the training splits such that each sample was a strict subset of any larger training sample of the same split.

We evaluate on the standard metric of F1 with respect to the target class. For Amazon, in which both the "positive" and "negative" classes are potential target classes, we evaluate using macro-averaged scores.

The primary results of our experiments are shown in Table 4. The results show that MNB-FM improves upon the MNB classifier substantially, and also tends to outperform the other SSL and supervised learning methods we evaluated. MNB-FM is the best performing method over all data sets when the labeled data is limited to 10 and 100 documents, except for training sets of size 10 in Aptemod, where MNB has a slight edge.

Tables 5 and 6 present detailed results of the experiments on the RCV1 data set. These experiments are limited to the 5 largest base classes and show the F1 performance of MNB-FM and the various comparison methods, excluding Label Propagation which was intractable on this data set.

| Class | MNB-FM | SFE | MNB | NBEM | Logist. |
|---|---|---|---|---|---|
| CCAT | 0.641 | **0.643** | 0.580 | 0.639 | 0.532 |
| GCAT | 0.639 | 0.686 | 0.531 | **0.732** | 0.466 |
| MCAT | **0.572** | 0.505 | 0.393 | 0.504 | 0.225 |
| ECAT | **0.306** | 0.267 | 0.198 | 0.224 | 0.096 |
| GPOL | 0.313 | 0.283 | 0.233 | **0.326** | 0.043 |
| Average | **0.494** | 0.477 | 0.387 | 0.485 | 0.272 |

Table 5: RCV1: F1, $|D_L|$= 10

| Class | MNB-FM | SFE | MNB | NBEM | Logist. |
|---|---|---|---|---|---|
| CCAT | **0.797** | 0.793 | 0.624 | 0.713 | 0.754 |
| GCAT | **0.849** | 0.848 | 0.731 | 0.837 | 0.831 |
| MCAT | **0.776** | 0.737 | 0.313 | 0.516 | 0.689 |
| ECAT | **0.463** | 0.317 | 0.017 | 0.193 | 0.203 |
| GPOL | **0.499** | 0.370 | 0.002 | 0.089 | 0.114 |
| Average | **0.677** | 0.613 | 0.337 | 0.470 | 0.518 |

Table 6: RCV1: F1, $|D_L|$= 100

| Method | 1000 | 5000 | $10k$ | $50k$ | $100k$ |
|---|---|---|---|---|---|
| MNB-FM | 1.44 | 1.61 | 1.69 | 2.47 | 5.50 |
| NB+EM | 2.95 | 3.43 | 4.93 | 10.07 | 16.90 |
| MNB | 1.15 | 1.260 | 1.40 | 2.20 | 3.61 |
| Labelprop | 0.26 | 4.17 | 10.62 | 67.58 | - |

Table 7: Runtimes of SSL methods (sec.)

The runtimes of our methods can be seen in Table 7. The results show the runtimes of the SSL methods discussed in this paper as the size of the unlabeled dataset grows. As expected, we find that MNB-FM has runtime similar to MNB, and scales much better than methods that take multiple passes over the unlabeled data.

## 5 Analysis

From our experiments, it is clear that the performance of MNB-FM improves on MNB, and in many cases outperforms all existing SSL algorithms we evaluated. MNB-FM improves the conditional probability estimates in MNB and, surprisingly, we found that it can often improve these estimates for words that do not even occur in the training set.

Tables 8 and 9 show the details of the improvements MNB-FM makes on the feature marginal estimates. We ran MNB-FM and MNB on the RCV1 class MCAT and stored the computed feature marginals for direct comparison. For each word in the vocabulary, we compared each classifier's conditional probability ratios, i.e. $\theta^+/\theta^-$, to the true value over the entire data set. We computed which classifier was closer to the correct ratio for each word. These results were averaged over 5 iterations. From the data, we can see that MNB-FM improves the estimates for many words *not* seen in the training set as well as the most common words, even with small training sets.

### 5.1 Ranking Performance

We also analyzed how well the different methods *rank*, rather than classify, the test documents. We evaluated ranking using the R-precision metric, equal to the precision (i.e. fraction of positive documents classified correctly) of the $R$ highest-ranked test documents, where $R$ is the total number of positive test documents.

Logistic Regression performed particularly well on the R-Precision Metric, as can be seen in Tables 10, 11, and 12. Logistic Regression performed less well in the F1 metric. We find that NB+EM

| | Fraction Improved vs MNB | | | Avg Improvement vs MNB | | | Probability Mass | | |
|---|---|---|---|---|---|---|---|---|---|
| Word Freq. | Known | Half Known | Unknown | Known | Half Known | Unknown | Known | Half Known | Unknown |
| 0-$10^{-6}$ | - | 0.165 | **0.847** | - | -0.805 | **0.349** | - | 0.02% | 7.69% |
| $10^{-6}$-$10^{-5}$ | 0.200 | 0.303 | **0.674** | **0.229** | -0.539 | **0.131** | 0.00% | 0.54% | 14.77% |
| $10^{-5}$-$10^{-4}$ | 0.322 | 0.348 | **0.592** | -0.597 | -0.424 | **0.025** | 0.74% | 10.57% | 32.42% |
| $10^{-4}$-$10^{-3}$ | **0.533** | **0.564** | 0.433 | **0.014** | **0.083** | -0.155 | 7.94% | 17.93% | 7.39% |
| $> 10^{-3}$ | - | - | - | - | - | - | - | - | - |

Table 8: Analysis of Feature Marginal Improvement of MNB-FM over MNB ($|D_L| = 10$). "Known" indicates words occurring in both positive and negative training examples, "Half Known" indicates words occurring in only positive or negative training examples, while "Unknown" indicates words that never occur in labelled examples. Data is for the RCV1 MCAT category. MNB-FM improves estimates by a substantial amount for unknown words and also the most common known and half-known words.

| | Fraction Improved vs MNB | | | Avg Improvement vs MNB | | | Probability Mass | | |
|---|---|---|---|---|---|---|---|---|---|
| Word Freq. | Known | Half Known | Unknown | Known | Half Known | Unknown | Known | Half Known | Unknown |
| 0-$10^{-6}$ | **0.567** | 0.243 | **0.853** | **0.085** | -0.347 | **0.143** | 0.00% | 0.22% | 7.49% |
| $10^{-6}$-$10^{-5}$ | 0.375 | 0.310 | **0.719** | -0.213 | -0.260 | **0.087** | 0.38% | 4.43% | 10.50% |
| $10^{-5}$-$10^{-4}$ | 0.493 | 0.426 | **0.672** | -0.071 | -0.139 | **0.067** | 18.68% | 20.37% | 4.67% |
| $10^{-4}$-$10^{-3}$ | **0.728** | **0.669** | - | **0.233** | **0.018** | - | 31.70% | 1.56% | - |
| $> 10^{-3}$ | - | - | - | - | - | - | - | - | - |

Table 9: Analysis of Feature Marginal Improvement of MNB-FM over MNB ($|D_L| = 100$). Data is for the RCV1 MCAT category (see Table 8). MNB-FM improves estimates by a substantial amount for unknown words and also the most common known and half-known words.

performs particularly well on the R-precision metric on ApteMod, suggesting that its modelling assumptions are more accurate for that particular data set (NB+EM performs significantly worse on the other data sets, however). MNB-FM performs essentially equivalently well, on average, to the best competing method (Logistic Regression) on the large RCV1 data set. However, these experiments show that MNB-FM offers more advantages in document classification than in document ranking.

The ranking results show that LR may be preferred when ranking is important. However, LR underperforms in classification tasks (in terms of F1, Tables 4-6). The reason for this is that LR's learned classification threshold becomes less accurate when datasets are small and classes are highly skewed. In these cases, LR classifies too frequently in favor of the larger class which is detrimental to its performance. This effect is visible in Tables 5 and 6, where LR's performance significantly drops for the ECAT and GPOL classes. ECAT and GPOL represent only 14.91% and 7.07% of the RCV1 dataset, respectively.

| Class | MNB-FM | SFE | MNB | NBEM | LProp | Logist. |
|---|---|---|---|---|---|---|
| Apte (10) | 0.353 | 0.304 | 0.359 | **0.631** | 0.490 | 0.416 |
| Apte (100) | 0.555 | 0.421 | 0.343 | **0.881** | 0.630 | 0.609 |
| Apte (1k) | 0.723 | 0.652 | 0.532 | **0.829** | 0.754 | 0.795 |
| Amzn (10) | 0.536 | 0.527 | 0.516 | 0.481 | 0.535* | **0.544** |
| Amzn (100) | 0.614 | 0.562 | 0.517 | 0.480 | 0.573* | **0.639** |
| Amzn (1k) | 0.717 | 0.650 | 0.562 | 0.483 | 0.639* | **0.757** |
| RCV1 (10) | 0.505 | 0.480 | 0.421 | 0.450 | - | **0.512** |
| RCV1 (100) | 0.683 | 0.614 | 0.474 | 0.422 | - | **0.689** |
| RCV1 (1k) | 0.781 | 0.748 | 0.535 | 0.454 | - | **0.802** |
| * Limited to 5 of 10 Amazon categories | | | | | | |

Table 10: R-Precision, training size in parentheses

## 6 Related Work

To our knowledge, MNB-FM is the first approach that utilizes a small set of statistics computed over

| Data Set | MNB-FM | SFE | MNB | NBEM | Logist. |
|---|---|---|---|---|---|
| CCAT | 0.637 | 0.631 | 0.620 | 0.498 | **0.653** |
| GCAT | 0.663 | 0.711 | 0.600 | **0.792** | 0.671 |
| MCAT | 0.580 | 0.492 | 0.477 | 0.510 | **0.596** |
| ECAT | 0.291 | 0.217 | 0.214 | 0.111 | **0.297** |
| GPOL | **0.354** | 0.352 | 0.193 | 0.341 | 0.341 |
| Average | 0.505 | 0.480 | 0.421 | 0.450 | **0.512** |

Table 11: RCV1: R-Precision, $D_L$= 10

| Class | MNB-FM | SFE | MNB | NBEM | Logist. |
|---|---|---|---|---|---|
| CCAT | 0.805 | 0.797 | 0.765 | 0.533 | **0.809** |
| GCAT | 0.849 | 0.858 | 0.780 | **0.869** | 0.843 |
| MCAT | **0.782** | 0.753 | 0.579 | 0.533 | 0.774 |
| ECAT | 0.471 | 0.293 | 0.203 | 0.119 | **0.498** |
| GPOL | 0.509 | 0.370 | 0.042 | 0.056 | **0.520** |
| Average | 0.683 | 0.614 | 0.474 | 0.422 | **0.689** |

Table 12: RCV1: R-Precision, $D_L$= 100

a large unlabeled data set as constraints to improve a semi-supervised classifier. Our experiments demonstrate that MNB-FM outperforms previous approaches across multiple text classification techniques including topic classification and sentiment analysis. Further, the MNB-FM approach offers scalability advantages over most existing semi-supervised approaches.

Current popular Semi-Supervised Learning approaches include using Expectation-Maximization on probabilistic models (e.g. (Nigam et al., 2000)); Transductive Support Vector Machines (Joachims, 1999); and graph-based methods such as Label Propagation (LP) (Zhu and Ghahramani, 2002) and their more recent, more scalable variants (e.g. identifying a small number of representative unlabeled examples (Liu et al., 2010)). In general, these techniques require passes over the entirety of the unlabeled data for each new learning task, intractable for massive unlabeled data sets. Naive implementations of LP cannot scale to large unlabeled data sets, as they have time complexity that increases quadratically with the number of unlabeled examples. Recent LP techniques have achieved greater scalability through the use of parallel processing and heuristics such as Approximate-Nearest Neighbor (Subramanya and Bilmes, 2009), or by decomposing the similarity matrix (Lin and Cohen, 2011). Our approach, by contrast, is to pre-compute a small set of marginal statistics over the unlabeled data, which eliminates the need to scan unlabeled data for each new task. Instead, the complexity of MNB-FM is proportional only to the number of unique words in the labeled data set.

In recent work, Su et al. propose the Semi-supervised Frequency Estimate (SFE), which like MNB-FM utilizes the marginal probabilities of features computed from unlabeled data to improve the Multinomial Naive Bayes (MNB) classifier (Su et al., 2011). SFE has the same scalability advantages as MNB-FM. However, unlike our approach, SFE does not compute maximum-likelihood estimates using the marginal statistics as a constraint. Our experiments show that MNB-FM substantially outperforms SFE.

A distinct method for pre-processing unlabeled data in order to help scale semi-supervised learning techniques involves dimensionality reduction or manifold learning (Belkin and Niyogi, 2004), and for NLP tasks, identifying word representations from unlabeled data (Turian et al., 2010). In contrast to these approaches, MNB-FM preserves the original feature set and is more scalable (the marginal statistics can be computed in a single pass over the unlabeled data set).

## 7 Conclusion

We presented a novel algorithm for efficiently leveraging large unlabeled data sets for semi-supervised learning. Our MNB-FM technique optimizes a Multinomial Naive Bayes model to accord with statistics of the unlabeled corpus. In experiments across topic classification and sentiment analysis, MNB-FM was found to be more accurate and more scalable than several supervised and semi-supervised baselines from previous work.

In future work, we plan to explore utilizing richer statistics from the unlabeled data, beyond word marginals. Further, we plan to experiment with techniques for unlabeled data sets that also include continuous-valued features. Lastly, we also wish to explore ensemble approaches that combine the best supervised classifiers with the improved class-conditional estimates provided by MNB-FM.

## 8 Acknowledgements

This work was supported in part by DARPA contract D11AP00268.

## References

Mikhail Belkin and Partha Niyogi. 2004. Semi-supervised learning on riemannian manifolds. *Machine Learning*, 56(1):209–239.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Association for Computational Linguistics*, Prague, Czech Republic.

O. Chapelle, B. Schölkopf, and A. Zien, editors. 2006. *Semi-Supervised Learning*. MIT Press, Cambridge, MA.

Thorsten Joachims. 1999. Transductive inference for text classification using support vector machines. In *Proceedings of the Sixteenth International Conference on Machine Learning*, ICML '99, pages 200–209, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

David D Lewis, Yiming Yang, Tony G Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. *The Journal of Machine Learning Research*, 5:361–397.

Frank Lin and William W Cohen. 2011. Adaptation of graph-based semi-supervised methods to large-scale text data. In *The 9th Workshop on Mining and Learning with Graphs*.

Wei Liu, Junfeng He, and Shih-Fu Chang. 2010. Large graph construction for scalable semi-supervised learning. In *ICML*, pages 679–686.

Gideon S. Mann and Andrew McCallum. 2010. Generalized expectation criteria for semi-supervised learning with weakly labeled data. *J. Mach. Learn. Res.*, 11:955–984, March.

Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. 2000. Text classification from labeled and unlabeled documents using em. *Mach. Learn.*, 39(2-3):103–134, May.

Jiang Su, Jelber Sayyad Shirab, and Stan Matwin. 2011. Large scale text classification using semisupervised multinomial naive bayes. In Lise Getoor and Tobias Scheffer, editors, *ICML*, pages 97–104. Omnipress.

Amar Subramanya and Jeff A. Bilmes. 2009. Entropic graph regularization in non-parametric semi-supervised classification. In *Neural Information Processing Society (NIPS)*, Vancouver, Canada, December.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. *Urbana*, 51:61801.

Yiming Yang and Xin Liu. 1999. A re-examination of text categorization methods. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 42–49. ACM.

X. Zhu and Z. Ghahramani. 2002. Learning from labeled and unlabeled data with label propagation. Technical report, Technical Report CMU-CALD-02-107, Carnegie Mellon University.

Xiaojin Zhu. 2006. Semi-supervised learning literature survey.