Feature Selection for Text Categorization on Imbalanced Data

Zhaohui Zheng
Dept. of Computer Science
and Engineering
University at Buffalo
Amherst, NY 14260
zzheng3@cse.buffalo.edu

Xiaoyun Wu
Dept. of Computer Science
and Engineering
University at Buffalo
Amherst, NY 14260
xwu@cse.buffalo.edu

Rohini Srihari
Dept. of Computer Science
and Engineering
University at Buffalo
Amherst, NY 14260
rohini@cse.buffalo.edu

ABSTRACT

A number of feature selection metrics have been explored in text categorization, among which information gain (IG), chi-square (CHI), correlation coefficient (CC) and odds ratios (OR) are considered most effective. CC and OR are one-sided metrics while IG and CHI are two-sided. Feature selection using one-sided metrics selects the features most indicative of membership only, while feature selection using two-sided metrics implicitly combines the features most indicative of membership (e.g. positive features) and nonmembership (e.g. negative features) by ignoring the signs of features. The former never consider the negative features, which are quite valuable, while the latter cannot ensure the optimal combination of the two kinds of features especially on imbalanced data. In this work, we investigate the usefulness of explicit control of that combination within a proposed feature selection framework. Using multinomial naïve Bayes and regularized logistic regression as classifiers, our experiments show both great potential and actual merits of explicitly combining positive and negative features in a nearly optimal fashion according to the imbalanced data.

1. INTRODUCTION

Feature selection has been applied to text categorization in order to improve its scalability, efficiency and accuracy. Since each document in the collection can belong to multiple categories, the classification problem is usually split into multiple binary classification problems with respect to each category. Accordingly, features are selected locally per category, e.g. local feature selection.

A number of feature selection metrics have been explored, notable among which are Information Gain (IG), Chi-square (CHI), Correlation Coefficient (CC), and Odds Ratio (OR) [8; 9; 10; 12; 15]. CC and OR are one-sided metrics which select the features most indicative of membership for a category only, while IG and CHI are two-sided metrics, which consider the features most indicative of either membership (e.g. positive features) or non-membership (e.g. negative features). A feature selection metric is considered as one-sided if its positive and negative values correspond to positive and negative features respectively. On the other hand, a two-sided metric is non-negative with the signs of features ignored.

One choice in the feature selection policy is whether to rule out all negative features. Some argue that classifiers built from positive features only may be more transferable to new situations where the background class varies. Others believe that negative features are numerous, given the imbalanced data set, and quite valuable in practical experience. Their experiments show that when deprived of negative features, the performance of all feature selection metrics degrades, which indicates negative features are essential to high quality classification [3]. We think that negative features are useful because their presence in a document highly indicates its non-relevance. Therefore, they help to confidently reject non-relevant documents.

The focus in this paper is to answer the following three questions with empirical evidence:

- How sub-optimal are two sided metrics?
- To what extent can the performance be improved by better combination of positive and negative features?
- How can the optimal combination be learned in practice?

The first two questions are concerned with the potential of optimal combination of positive and negative features, and the last with a practical solution.

Imbalanced datasets are commonly encountered in text categorization problems, especially in the binary setting. A two-sided metric implicitly combines the positive and negative features by simply ignoring the signs of features and comparing their values. However, the values of positive features are not comparable with those of negative features due to the imbalanced data. Furthermore, different performance measures attach different weights to positive and negative features: the optimal size ratio between the two kinds of features should also depend on the performance measure. Two-sided metrics cannot ensure the optimal combination of the positive and negative features. However, neither one-sided nor two-sided metrics themselves allow control of the combination.

In order to examine the effect of control of the combination of positive and negative features, this paper presents a novel feature selection framework, in which the positive and negative features are selected separately, and then combined explicitly afterwards. Several standard methods are unified by this framework and a set of new methods is proposed that optimally combines the positive and negative features for each category according to the data characteristics and performance measure.

The rest of the paper is organized as follows. Sections 2 and 3 describe the related work: various feature selection metrics and the imbalanced data problem respectively. In Section 4, we present the new feature selection framework. The experimental setup is reported in Section 5, and results are analyzed in Section 6. The last section concludes.

2. FEATURE SELECTION METRICS

In this section, we present six feature selection metrics (four known measures and two proposed variants), which are functions of the following four dependency tuples:

1. (t, c_i): presence of t and membership in c_i.
2. $(t, \overline{c}_i)$: presence of t and non-membership in c_i.
3. $(\overline{t}, c_i)$: absence of t and membership in c_i.
4. $(\overline{t}, \overline{c}_i)$: absence of t and non-membership in c_i.

where: t and c_i represent a term and a category respectively. The frequencies of the four tuples in the collection are denoted by A, B, C and D respectively. The first and last tuples represent the positive dependency between t and c_i, while the other two represent the negative dependency.

Information gain (IG) Information gain [12; 15] measures the number of bits of information obtained for category prediction by knowing the presence or absence of a term in a document. The information gain of term t and category c_i is defined to be:

$$IG(t, c_i) = \sum_{c \in \{c_i, \overline{c}_i\}} \sum_{t' \in \{t, \overline{t}\}} P(t', c) \cdot log \frac{P(t', c)}{P(t') \cdot P(c)}$$

Information gain is also known as Expected Mutual Information. The Expected Likelihood Estimation (ELE) smoothing technique was used in this paper to handle singularities when estimating those probabilities.

Chi-square (CHI) Chi-square measures the lack of independence between a term t and a category c_i and can be compared to the chi-square distribution with one degree of freedom to judge extremeness [12; 15]. It is defined as:

$$\chi^2(t, c_i) = \frac{N[P(t, c_i)P(\overline{t}, \overline{c}_i) - P(t, \overline{c}_i)P(\overline{t}, c_i)]^2}{P(t)P(\overline{t})P(c_i)P(\overline{c}_i)}$$

where: N is the total number of documents.

Correlation coefficient (CC) Correlation coefficient of a word t with a category c_i was defined by Ng et al. as [12; 10]

$$CC(t, c_i) = \frac{\sqrt{N}[P(t, c_i)P(\overline{t}, \overline{c}_i) - P(t, \overline{c}_i)P(\overline{t}, c_i)]}{\sqrt{P(t)P(\overline{t})P(c_i)P(\overline{c}_i)}}$$

It is a variant of the CHI metric, where $CC^2 = \chi^2$. CC can be viewed as a "one-sided" chi-square metric.

Odds ratio (OR) Odds ratio measures the odds of the word occurring in the positive class normalized by that of the negative class. The basic idea is that the distribution of features on the relevant documents is different from the distribution of features on the non-relevant documents. It has been used by Mladenić for selecting terms in text categorization [8]. It is defined as follows:

$$OR(t, c_i) = log \frac{P(t|c_i)[1 - P(t|\overline{c}_i)]}{[1 - P(t|c_i)]P(t|\overline{c}_i)}$$

Similar to IG, ELE smoothing was used when estimating those conditional probabilities.

According to the definitions, OR considers the first two dependency tuples, and IG, CHI, and CC consider all the four tuples. CC and OR are one-sided metrics, whose positive and negative values correspond to the positive and negative features respectively. On the other hand, IG and CHI are two-sided, whose values are non-negative. We can easily obtain that the sign for a one-sided metric, e.g. CC or OR, is $sign(AD - BC)$.

A one-sided metric could be converted to its two-sided counterpart by ignoring the sign, while a two-sided metric could be converted to its one-sided counterpart by recovering the sign, e.g. CHI vs. CC.

We propose the two-sided counterpart of OR, namely OR-square, and the one-sided counterpart of IG, namely signed IG as follows.

OR-square (ORS) and Signed IG (SIG)

$$ORS(t, c_i) = OR^2(t, c_i),$$

$$SIG(t, c_i) = sign(AD - BC) \cdot IG(t, c_i)$$

The overall feature selection procedure is to score each potential feature according to a particular feature selection metric, and then take the best features. Feature selection using one-sided metrics like SIG, CC, and OR pick out the terms most indicative of membership only. The basic idea behind this is the features coming from non-relevant documents are useless. They will never consider negative features unless all the positive features have already been selected. Feature selection using two-sided metrics like IG, CHI, and ORS, however, do not differentiate between the positive and negative features. They implicitly combine the two.

3. THE IMBALANCED DATA PROBLEM

The imbalanced data problem occurs when the training examples are unevenly distributed among different classes. In case of binary classification, the number of examples in one class is significantly greater than that of the other. Attempts have been made to deal with this problem in diverse domains such as fraud detection [2], in-flight helicopter gearbox fault monitoring [4], and text categorization [6; 1; 9].

When training a binary classifier per category in text categorization, we use all the documents in the training corpus that belong to that category as relevant training data and all the documents in the training corpus that belong to all the other categories as non-relevant training data. It is often the case that there is an overwhelming number of non-relevant training documents especially when there is a large collection of categories with each assigned to a small number of documents, which is typically an "imbalanced data problem". This problem presents a particular challenge to classification algorithms, which can achieve high accuracy by simply classifying every example as negative. To overcome this problem "query zone" and "category zone" have been introduced to select a subset of *most relevant* non-relevant documents as the non-relevant training data [13]. Essentially these techniques try to obtain more balanced relevant and non-relevant training data by under-sampling negative examples.

In this work, we consider the imbalanced data problem from a different perspective. As Forman [3] argued, feature selection should be relatively more important than classification algorithms in highly imbalanced situations. Instead of balancing the training data, our methods actively-select the most useful features, e.g. combine positive and negative features in a nearly optimal fashion, according to the imbalanced data. This provides an alternative to handle the imbalanced data problem. Experimental comparison of our methods and those sampling strategies will be our future research.

The impact of imbalanced data problem on the standard feature selection can be illustrated as follows, which primarily answers the first question of Section 1:

First, for the methods using one-sided metrics (e.g. SIG, CC, and OR), the non-relevant documents are subject to misclassification. It will be even worse for the imbalanced data problem, where non-relevant documents dominate. How to confidently reject the non-relevant documents is important in that case.

Second, given a two-sided metric, the values of positive features are not necessarily comparable with those of negative features. Let us use CHI for example. The upper limit CHI value of a positive or negative feature is N. For the positive feature, it represents the case that the feature appears in every relevant document, but never in any non-relevant document. For the negative features, it means that the feature appears in every non-relevant document, but never in any relevant document. Due to the large amount and diversity of the non-relevant documents in imbalanced data set, it is much more difficult for a negative feature to reach the same maximum that a positive feature does. This extreme example shed light on why the CHI values of positive features are usually much larger than those of negative features. CHI and CC are very similar when the size of the feature set is small and the data set is highly imbalanced.

Third, let TP, FP, FN and TN denote true positives, false positives, false negatives and true negatives respectively. Positive features have more effect on TP and FN, while negative features have more effect on TN and FP. Two-sided metrics attach the same weights to the two kinds of features. Therefore, feature selection using a two-sided metric combines the positive and negative features so as to optimize the accuracy, which is defined to be $\frac{TP+TN}{TP+TN+FP+FN}$. $F1$ has been widely used in information retrieval, which is: $\frac{2 \cdot TP}{2 \cdot TP+FP+FN}$ [11]. In case of imbalanced dataset where TN is much larger than TP, the two measures are quite different. Two-sided metrics cannot ensure the optimal combination of positive and negative features according to $F1$.

Finally, some performance measures themselves, e.g. $F_\beta, \beta \neq 1.0$, attach different weights to precision(p) and recall(r) [11]:

$$\begin{aligned} F_\beta &= \frac{(\beta^2+1)p \times r}{\beta^2 p + r} \\ &= \frac{(\beta^2+1)\ TP}{(\beta^2+1)\ TP + FP + \beta^2\ FN} \end{aligned}$$

where p and r are defined as $\frac{TP}{TP+FP}$ and $\frac{TP}{TP+FN}$ respectively. Therefore, positive and negative features are considered of different importance. Two sided metrics cannot ensure the optimal combination according to those measures, no matter whether the dataset is balanced or not.

4. FEATURE SELECTION FRAMEWORK

Since implicit combination of positive and negative features using two-sided metrics is not necessarily optimal, explicit combination is the choice. In order to examine the effect of control of the combination, we propose separate handling of the two kinds of features in the following framework.

4.1 General formulation

For each category c_i :

- generate a positive-feature set F_i^+ of size l_1 by selecting the l_1 terms with highest $\Im(t, c_i)$. $l_1, 0 < l_1 \leq l$, is a natural number.
- generate a negative-feature set F_i^- of size l_2 by selecting the l_2 terms with highest $\Im(t, \overline{c}_i)$. $l_2 = l - l_1$ is a non-negative integer.
- $F_i = F_i^+ \cup F_i^-$.

where: l is the size of feature set, which is usually predefined. $0 < l_1/l \leq 1$ is the key parameter of the framework to be set. The function $\Im(\ \cdot\ , c_i)$ should be defined so that the larger $\Im(t, c_i)$ is, the more likely the term t belongs to the category c_i. Obviously, one-sided metrics like SIG, CC, and OR can serve as such functions, while two-sided metrics like IG, CHI, and ORS cannot.

In the first step, we intend to pick out those terms most indicative of membership of c_i, while those terms most indicative of non-membership are selected as well in the second step. The final feature set will be the union of the two.

Based on their definition, we can easily obtain:

$$SIG(t, \bar{c}_i) = -SIG(t, c_i),$$

$$CC(t, \bar{c}_i) = -CC(t, c_i),$$

$$OR(t, \bar{c}_i) = -OR(t, c_i),$$

Accordingly, the second step can be rewritten as:

- generate a negative-feature set F_i^- of size l_2 by selecting the l_2 terms with smallest $\Im(t, c_i)$.

Therefore, the framework combines the l_1 terms with largest $\Im(\ \cdot\ , c_i)$ and the $l - l_1$ terms with smallest $\Im(\ \cdot\ , c_i)$.

4.2 Two special cases

The standard feature selection methods generally fall into one of the following two groups:

1. select the positive features only using one-sided metrics, e.g. SIG, CC, and OR. For convenience, we will use CC as the representative of this group.
2. implicitly combine the positive and negative features using two-sided metrics, e.g. IG, CHI, and ORS. CHI will be chosen to represent this group.

The two groups are two special cases of our feature selection framework. The standard feature selection using CC corresponds to the case where $\Im = CC$, $l_1/l = 1$. The standard method using CHI corresponds to the case where $\Im = CC$, and l_1/l is implicitly set as follows. Considering CHI, we have $F_i = Max[\chi^2(\cdot, c_i), l]$. The positive subset of F_i is $F_i^{'} = \{t \in F_i | CC(t, c_i) > 0\}$. The feature set F_i can be equivalently obtained as a combination of the terms most indicative of membership and non-membership:

$$F_i^+ = Max[CC(\cdot, c_i), |F_i^{'}|];$$

$$F_i^- = Min[CC(\cdot, c_i), l - |F_i^{'}|];$$

$$F_i = F_i^+ \cup F_i^-$$

Where: $Max[\Im(\cdot, c_i), l]$ and $Min[\Im(\cdot, c_i), l]$ represent the l features with highest and smallest $\Im$ values respectively.

Note that $F_i^{'} \equiv F_i^+$. It illustrates the standard feature selection using CHI is a special case of the framework, where $l_1 = |F_i^{'}|$ is internally decided by the size of feature set l given the data set.

4.3 Optimization

The feature selection framework facilitates the control on explicit combination of the positive and negative features through the parameter l_1/l.

As we can see, the combination of positive and negative features is solely decided by the size ratio l_1/l given the predefined feature size and metric. One-sided and two-sided metrics actually correspond to two particular values of that ratio, which are not optimal. How to optimize the size ratio is the key. We design the following two scenarios for optimization answering the second and third questions of Section 1 respectively.

Ideal scenario is designed to explore the potential of explicit combination of positive and negative features. In this case, the size ratio for each category is optimized on the test set. That is, we use the training data to learn different models per category according to different size ratios ranging from 0 to 1 and select the ratio having best performance, e.g. F1, on the test set. It represents the "best possible" size ratio for each category.

Practical scenario Certainly, in practice the optimal size ratios cannot be learned from the test set. One simple way we tried in this paper is to empirically select the size ratio per category having best performance on the training set. This scenario is similar to wrapper techniques [5], but use feature selection metrics as heuristics to guide the search more efficiently and effectively.

Accordingly, a set of new methods corresponding to $\Im = SIG, CC$ and OR are proposed for each of the two scenarios, where $l_1/l \in (0, 1]$ is empirically chosen per category according to the scenario. The first set of methods are referred to as improved SIG, CC and OR in ideal scenario while the other set are improved SIG, CC and OR in practical scenario.

The efficient implementation of optimization within the framework is as follows:

- select l positive features with greatest $\Im(t, c_i)$ in a decreasing order.
- select l negative features with smallest $\Im(t, c_i)$ in an increasing order.
- empirically choose the size ratio l_1/l such that the feature set constructed by combining the first $l_1, 0 < l_1 < l$, positive features and the first $l - l_1$ negative features has the optimal performance.

Therefore, during the optimization of size ratio l_1/l for each category, we did not conduct feature selection for each possible l_1/l, but once only.

5. EXPERIMENTAL SETUP

In order to determine the usefulness of the new feature selection methods, we conduct experiments on standard text categorization data using two popular classifiers: naïve Bayes and logistic regression.

5.1 Data collection

Reuters-21578 (ModApte split) is used as our data collection, which is a standard data set for text categorization [14; 15; 16]. This dataset contains 90 categories, with 7769 training documents and 3019 test documents. After filtering out all numbers, stop-words and words occurring less than 3 times, we have 9484 indexing words in the vocabulary. Words in document titles are treated same as in document body.

5.2 Classifiers

On the training algorithms, We used naïve Bayes (NB for short) and regularized logistic regression (LR for short).

The multinomial mixture model of NB (tf representation) and multivariate model of LR (binary representation) are used [7; 17].

A NB score between a document d and the category c_i can be calculated as:

$$Score(d, c_i) = \frac{logP(c_i) + \sum_{f_j} logP(f_j|c_i)}{logP(\overline{c}_i) + \sum_{f_j} logP(f_j|\overline{c}_i)}$$

where: f_j is the feature appearing in the document d, $P(c_i)$ and $P(\overline{c}_i)$ represent prior probabilities of relevant and non-relevant respectively, and $P(f_j|c_i)$ and $P(f_j|\overline{c}_i)$ are conditional probabilities estimated with Laplacian smoothing.

Logistic regression tries to model the conditional probability as:

$$P(y|d, w) = \frac{1}{1 + exp(-y_i w^T d)}$$

The optimization problem for LR is to minimize:

$$w^* = argmin_w \{ \frac{1}{n} \sum_{i=1}^{n} log(1 + exp(-y_i w^T d_i)) + \lambda w^T w \}$$

where d_i is the ith training example, w is the weight vector, $y_i \in \{-1, 1\}$ is the label associated with d_i, and λ is an appropriately chosen regularization parameter, set to be 0.0001 as suggested in [17]. Column relaxation with Gauss-Seidel is used for solving this optimization [17; 16].

5.3 Performance measure

To measure the performance, we use both precision (p) and recall (r) in their combined from $F1 : \frac{2pr}{p+r}$ [11]. To remain compatible with other results, the $F1$ value at Break Even Point (BEP) [17] is reported throughout this paper, which avoids the tuning of thresholds and purely evaluates the direction of the decision hyperplane learned by the linear method itself. BEP is defined to be the point where precision equals recall. It corresponds to the minimum of $|FP - FN|$ in practice. To measure the global performance of different methods, We report both micro and macro-averaged BEP F1.

The micro-averaged F1 is largely dependent on the most common categories while the rare categories influence macro-averaged F1. The two most common categories: **earn** and **acq** contain many more positive documents than the remaining categories and both of them have very good performance on different classifiers (around .95 BEP F1 with both NB and LR). The micro-averaged F1 is dominated by the F1 values of the two categories. Besides, feature selection on imbalanced data is our focus in this study. The top two categories are somewhat balanced. On the other side, for those extremely rare categories containing only a few documents, their F1 values are unstable, which will affect the reliability of macro-averaged F1 [16]. Therefore, we decided to exclude the two most common categories and those categories containing less than ten training documents from our collection, which results in 58 categories from the third to the sixtieth, ranked in decreasing order according to the number of positive examples. Figure 1 show the percentages of positive examples in the training set for the 58 categories. The percentage ranges from a minimum of 0.1% to maximum of 7% in the training set, which indicate high class imbalances of the dataset.

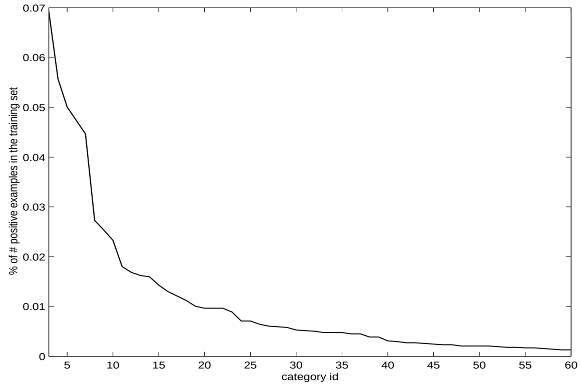

Figure 1: The percentage of # positive examples in the training set (58 categories:3rd-60th)

We also list the maximum IG, CHI and ORS values of positive and negative features for each of the 58 categories in table 7 as attached, which verifies the second illustration of Section 3: the values of positive features are not comparable with those of negative features, given a two-sided metric.

5.4 Optimization

During the optimization of the size ratio per category, we tried different size ratios ranging from 0 to 1: $0, .05, \cdots, .95, 1$ (21 possible ratios) and select the ratio having best BEP F1 on test and training sets corresponding to the two optimization scenarios respectively.

6. PRIMARY RESULTS AND ANALYSIS

In order to compare different feature selection methods, we apply them to text categorization using NB and LR, and compare their performance in terms of micro and macro averaged F1 (BEP). We report the performance of NB and LR with the improved feature selection methods in both optimization scenarios as compared to the standard feature selection methods using one-sided and two-sided metrics.

6.1 Ideal scenario

In order to answer the second question of Section 1, we consider the following three groups of feature selection methods:

- Standard IG, Standard SIG and improved SIG in ideal optimization scenario. The three methods are referred to as IG, SIG and iSIG for notational simplicity.

- Standard CHI, Standard CC and improved CC in ideal optimization scenario, referred to as CHI, CC and iCC respectively.

- Standard ORS, Standard OR and improved OR in ideal optimization scenario, referred to as ORS, OR and iOR respectively.

Table 1: Micro-averaged F1(BEP) values for NB with the feature selection methods at different sizes of features over the 58 categories:3rd-60th. The micro-averaged F1(BEP) for NB without feature selection (using all 9484 features) is .641

$\|F_i\|$	IG	SIG	iSIG	CHI	CC	iCC	ORS	OR	iOR
10	.67	.673	.737	.681	.681	.735	.438	.445	.502
20	.693	**.684**	.751	**.685**	**.685**	.766	.46	.481	.565
30	**.694**	.674	.755	.681	.681	.762	.486	.489	.582
40	.681	.669	**.763**	.676	.676	.769	.514	.509	.615
50	.692	.668	.762	.671	.67	**.77**	.56	.543	.647
100	.678	.657	.751	.668	.655	.767	.651	.597	.725
200	.681	.655	.756	.67	.65	.755	.685	.611	.764
500	.676	.654	.749	.683	.643	.749	**.694**	.623	**.768**
1000	.662	.661	.74	.666	.647	.74	.676	.626	.761
2000	.647	.653	.729	.671	.627	.724	.676	**.628**	.751
3000	.63	.648	.72	.667	.606	.713	.669	.619	.743

Table 2: As table 1, but for macro-averaged F1(BEP). The macro-averaged F1(BEP) for NB without feature selection is .483

$\|F_i\|$	IG	SIG	iSIG	CHI	CC	iCC	ORS	OR	iOR
10	.626	.627	.725	.62	.62	.719	.482	.483	.60
20	.606	.602	.731	.616	.616	.733	.497	.503	.665
30	.603	.588	.728	.595	.595	.731	.50	.499	.677
40	.573	.56	.721	.588	.588	.732	.529	.522	.692
50	.589	.566	.712	.59	.59	.733	.548	.537	.701
100	.579	.542	.703	.545	.54	.724	.579	.541	.721
200	.572	.55	.698	.568	.534	.715	.584	.543	.726
500	.558	.569	.693	.598	.521	.682	.577	.538	.709
1000	.522	.575	.68	.577	.52	.664	.553	.522	.684
2000	.47	.568	.666	.583	.476	.64	.545	.518	.665
3000	.432	.547	.65	.562	.427	.627	.523	.494	.657

The methods are compared to each other within the same group at different sizes of features. Typical size of a local feature set is between 10 and 50 [10]. In this paper, the performance are reported at a much wider range: 10 ∼ 3000.

Tables 1 and 2 list the micro and macro averaged BEP F1 values for naïve Bayes classifiers with the nine different feature selection methods (as listed in the first row) at different sizes of feature set ranging from 10 to 3000 (as listed in the first column); Tables 3 and 4 list the micro and macro averaged BEP F1 values for regularized logistic regression classifiers. The best micro averaged F1 across different sizes of feature set is highlighted for each method. We can see the improved methods in ideal scenario significantly outperform the corresponding one-sided and two-sided methods, which indicates the great potential of optimal combination of positive and negative features.

From tables 1 and 2, we can see it is very useful to conduct feature selection for NB (The micro-averaged F1 for NB without feature selection is .641). On the other side, tables 3 and 4 show that standard feature selection, e.g. IG, SIG, CHI, CC, ORS and OR, will not improve LR's performance (The micro-averaged F1 for LR without feature selection is .766), which confirms the general conseus that standard feature selection will not help the regularized linear methods, e.g. SVMs, LR and ridge regression. However, the best performance of the improved methods in ideal scenario (iSIG: .81, iCC: .812, and iOR: .816) shows that the improved feature selection methods can be very helpful to LR.

Figure 2 show the size ratios implicitly decided by two-sided metrics: IG, CHI and ORS respectively (feature size = 50), which confirms that feature selection using a two-sided metric is similar to its one-sided counterpart(size ratio = 1) when the feature size is small. When using CHI, only positive features are considered for 56 categories(equivalent to CC) and only one negative feature is included for each of the remaining two categories(3rd and 7th).

Figure 3 visualizes the optimization for the first two (*money-fx* and *grain*) and last two categories (*dmk* and *lumber*) using NB, where $\Im = CC$, feature size = 50. Figures 4 and 5 show the optimal size ratios for the 58 categories using NB and LR respectively. Both are quite different from figure 2, which confirms that implicit combination using two-sided metrics are not optimal. The 58 categories are ordered by the numbers of their training documents. Intuitively, given the fixed size of feature set, 50 in this case, the optimal size ratio should decrease with category id increases. We can see in figures 4 and 5 that the optimal size ratios vibrate between 0 and 1 from category to category irregularly though the general trend confirms the intuition. Therefore, besides

Table 3: Micro-averaged F1(BEP) values for LR with the feature selection methods at different sizes of features over the 58 categories. The micro-averaged F1(BEP) for LR without feature selection is .766

$\|F_i\|$	IG	SIG	iSIG	CHI	CC	iCC	ORS	OR	iOR
10	.70	.701	.723	.70	.70	.728	.447	.449	.474
20	.725	.719	.759	.726	.726	.768	.497	.508	.552
30	.735	.728	.776	.73	.73	.78	.527	.527	.575
40	.741	.733	.785	.737	.737	.79	.563	.569	.616
50	.745	.738	.794	.738	.738	.792	.597	.591	.642
100	.755	.74	.80	.746	.74	.799	.663	.666	.721
200	.76	.734	.807	.745	.736	.806	.718	.717	.763
500	.76	.748	**.81**	.756	.747	**.812**	.738	.731	.796
1000	.771	.747	.807	.76	.752	.807	.754	.738	.808
2000	.77	.747	.807	**.762**	.757	.805	.759	.744	.814
3000	**.774**	**.75**	.808	.761	**.759**	.804	**.762**	**.745**	**.816**

Table 4: As table 3, but for macro-averaged F1(BEP). The macro-averaged F1(BEP) for LR without feature selection is .676

$\|F_i\|$	IG	SIG	iSIG	CHI	CC	iCC	ORS	OR	iOR
10	.669	.669	.714	.651	.651	.703	.505	.505	.548
20	.68	.679	.741	.673	.673	.743	.569	.572	.648
30	.677	.673	.743	.675	.675	.756	.582	.583	.654
40	.684	.675	.748	.679	.679	.76	.605	.607	.677
50	.684	.676	.753	.686	.686	.763	.622	.619	.693
100	.698	.68	.759	.68	.678	.761	.651	.652	.741
200	.705	.683	.768	.695	.683	.768	.673	.675	.745
500	.703	.702	.771	.695	.699	.768	.681	.678	.751
1000	.708	.684	.77	.686	.697	.758	.688	.683	.754
2000	.706	.674	.766	.677	.699	.749	.686	.68	.756
3000	.712	.676	.765	.676	.697	.746	.691	.683	.76

number of positive examples, category(domain) characteristics also have effects on optimal feature selection. Our results also show the optimal size ratios learned by NB with iSIG, iCC and iOR (figure 4) are significantly different from those learned by LR (figure 5) respectively. Both results confirm Mladenić's observation [9]: a good feature scoring measure for text should consider domain and classification algorithm characteristics.

6.2 Practical scenario

We consider the improved CC in the practical optimization scenario also. The performance of the improved CC in practical optimization scenario with NB and LR is reported in table 5. The best performance with NB is .74 (feature size = 50), which is 3% lower than the ideal scenario (.77), but significantly (5.5%) better than CHI and CC (both are .685). The advantage of the improved CC in practical scenario over standard CHI and CC is also observed with LR. Wilcoxon signed rank tests show that improved CC in practical scenario significantly outperforms CHI and CC with both NB and LR at the 0.05 significance level. This verifies that the improved methods have not only great potential but practical merits also. Since LR is a highly effective classifier compared to non-regularized methods such as NB, the room for improvement is not that huge. This explains why the performance gain of LR using our new feature selection methods is not as much as that of NB. In this scenario, the size ratios were optimized over the whole training set, which might cause overfitting. We expect more performance gain by n-fold cross validation with the training set.

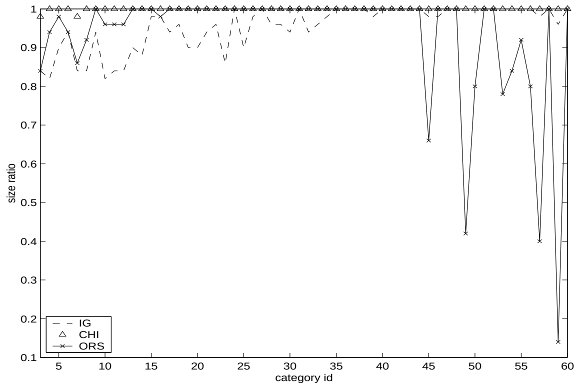

Figure 2: Size ratios implicitly decided by using two-sided metrics: IG, CHI and ORS respectively (58 categories:3rd-60th, feature size = 50)

The micro-averaged F1 values for NB with iCC in ideal and practical scenarios are .77 and .74 respectively (feature size = 50), both approach LR without feature selec-

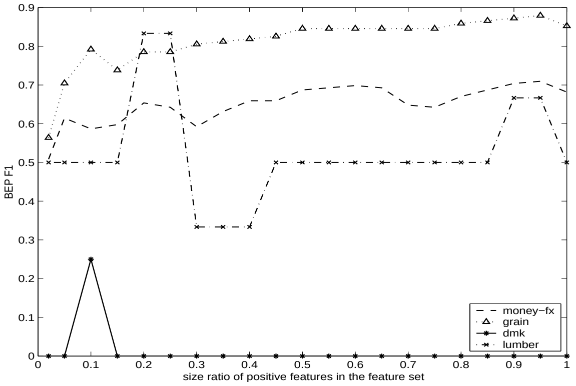

Figure 3: BEP F1 for test over the first two and last two categories (out of 58 categories) at different l_1/l values (NB, $\Im = CC$, feature size = 50). The optimal size ratios for *money-fx, grain, dmk* and *lumber* are 0.95, 0.95, 0.1 and 0.2 respectively

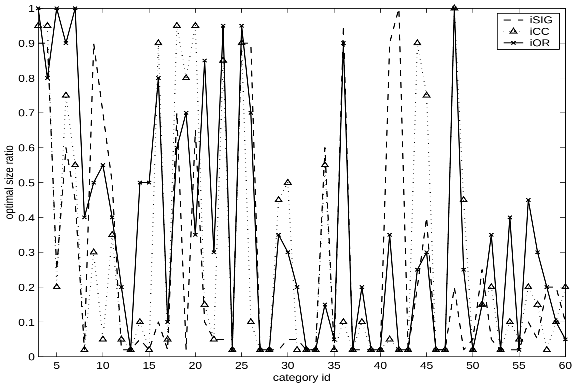

Figure 4: Optimal size ratios of iSIG, iCC and iOR (NB, 58 categories:3rd-60th, feature size = 50)

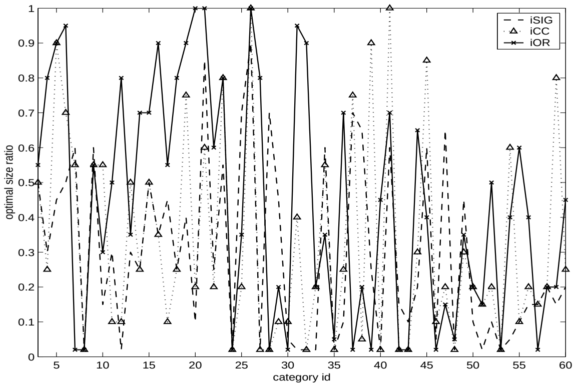

Figure 5: As figure 4, but for LR

Table 5: Micro and macro-averaged F1(BEP) values for NB and LR with the improved CC in practical scenario at different sizes of features over the 58 categories

$\|F_i\|$	NB		LR	
	micro	macro	micro	macro
10	.708	.658	.695	.652
20	.73	.673	.734	.672
30	.719	.657	.741	.679
40	.737	.676	.754	.691
50	**.74**	.68	.758	.696
100	.738	.673	.763	.694
200	.717	.664	.769	.712
500	.714	.633	.77	.702
1000	.693	.593	**.772**	.704
2000	.687	.591	.769	.696
3000	.686	.584	.771	.693

Table 6: BEP F1 values of NB and LR for the two most common categories: 1st and 2nd. iCC and iCC' represent ideal and practical scenarios respectively, feature size = 50

	NB				LR			
	CHI	CC	iCC	iCC'	CHI	CC	iCC	iCC'
earn	.957	.914	.958	.956	.971	.959	.974	.97
acq	.871	.819	.917	.907	.897	.858	.926	.926

tion (.766). This indicates that with the improved feature selection methods such as iCC, NB is competitive with state-of-the-art classification methods such as LR.

Obviously, the efficiency of the improved methods mainly depends on the classification method used to learn the optimal size ratios. For those fast algorithms such as NB, linear regression, etc, the optimization is reasonably efficient. Since the optimization of size ratio for one category is independent with other categories, parallel computing can be performed for those time-consuming classifiers, e.g. kNN, neural networks, SVMs, LR, etc, with many features (feature size > 1000). Possible further work includes an analytic solution to finding optimal size ratios based on the category characteristics.

6.3 Additional results

In order to investigate the effect of our new methods on balanced data, we report in table 6 the performance values for the two most common categories: **earn** and **acq** separately. For **earn**, no performance gain is observed by applying our new methods. It's due to the well-balanced nature of this category: around $\frac{2}{5}$ of its training examples are positive. However, the performance of **acq** was significantly improved. This can be partially explained by the fact that this category is more imbalanced than **earn**: only around $\frac{1}{5}$ of its training examples are positive.

In order to compare this work with others, we also show in figures 6 and 7 the micro-averaged BEP F1 values for the second group of feature selection methods on all 90 cate-

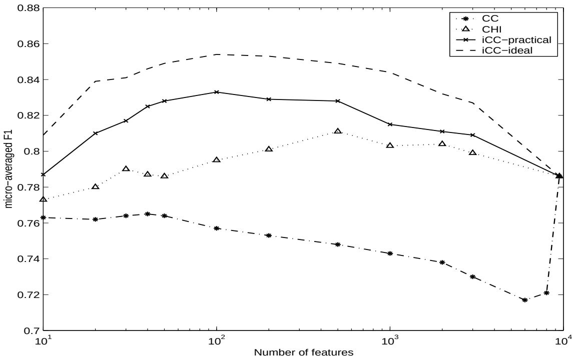

Figure 6: Micro-averaged BEP F1 values for CC, CHI, iCC in practical scenario, and iCC in ideal scenario (NB, all 90 categories)

gories. We can see in figure 6 the micro-averged F1 of all 90 categories for NB using CHI (with 2000 features) is 80.4%. The micro averaged F1(BEP) of LR without feature selection is .854 as shown in figure 7. Both are consistent with previous published results [14; 16; 17]. The micro averaged F1(BEP) score of LR is slightly lower than [17] due to a difference of treating document titles. [17; 16] treated words in document titles as different words in document bodies while we consider them equivalently.

As we expect, the improved methods consistently outperform standard CHI and CC. However, since the micro-averaged F1 over all 90 categories is dominated by the most common category, which happen to be well-balanced, the improvement is not as impressive. We can also see in figures 6 and 7 that CHI is always better than CC even when the feature size is small. For the two most common categories, CHI and CC will choose features quite differently since the CHI values of positive and negative features are comparable on balanced data. In that case, CHI will select useful negative features as well even when the feature size is small. Another interesting point in figure 6 is that with the number of features increases, the performance of CC first increases, then decreases, and finally increases again. The first increase is due to the inclusion of more useful positive features. The afterward decrease is due to the inclusion of noisy or non-indicative (1) positive and (2) negative features. The final increase is because of the inclusion of useful negative features. In contrast with figure 6, figure 7 does not see noticeable performance drop for CHI and CC, which is because that due to the regularization factor of LR, the inclusion of noisy features has much less impact on LR than NB.

7. CONCLUSION

In order to investigate the usefulness of experimenting with the combination of positive and negative features, a novel feature selection framework was presented, in which the positive and negative features are separately selected and explicitly combined. We explored three special cases of the framework:

1. consider the positive features only by using one-sided metrics;

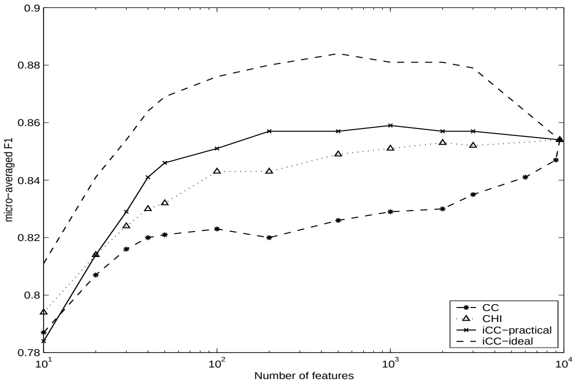

Figure 7: As figure 6, but for LR

2. implicitly combine the positive and negative features by using two-sided metrics;
3. combine the two kinds of features explicitly and choose the size ratio empirically such that optimal performance is obtained.

The first two cases are known and standard, and the last one is new. The main conclusions are:

- Implicitly combining positive and negative features using two-sided metrics is not necessarily optimal, especially on imbalanced data.
- A judicious combination shows great potential and practical merits.
- A good feature selection method should take into consideration the data set, performance measure, and classification methods.
- Feature selection can significantly improve the performance of both naïve Bayes and regularized logistic regression on imbalanced data.

Furthermore, we observe that multinomial naïve Bayes with our proposed feature selection is competitive with the state-of-the-art algorithms such as regularized logistic regression.

8. ACKNOWLEDGEMENTS

We would like to thank Dr. David Pierce, Dr. Miguel Ruiz, Mr. Wei Dai, and the anonymous reviewers for valuable comments and suggestions. Thanks to Mr. Jian Zhang and Dr. Tong Zhang for discussions about the implementation of logistic regression and evaluation issues.

9. REFERENCES

[1] S. Dumais, J. Platt, D. Heckerman, and M. Sahami. Inductive learning algorithms and representations for text categorization. *Proceedings of the Seventh International Conference on Information and Knowledge Management*, pages 148–155, 1998.

[2] T. Fawcett and F. Provost. Adaptive fraud detection. *Data Mining and Knowledge Discovery*, 1(3):291–316, 1997.

[3] G. Forman. An extensive empirical study of feature selection metrics for text classification. *JMLR, Speical Issue on Variable and Feature Selection*, pages 1289–1305, 2003.

[4] N. Japkowicz and S. Stephen. The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6(5), 2002.

[5] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, 1997.

[6] D. Lewis and M. Ringuette. A comparison of two learning algorithms for text categorization. *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 81–93, 1994.

[7] A. McCallum and K. Nigam. A comparison of event models for naive bayes text classification. *AAAI-98 Workshop on Learning for Text Categorization*, 1998.

[8] D. Mladeni. *Machine Learning on non-homogeneous, distributed text data.* PhD Dissertation, University of Ljubljana, Slovenia, 1998.

[9] D. Mladeni and G. Marko. Feture selection for unbalanced class distribution and naive bayes. *The Sixteenth International Conference on Machine Learning*, pages 258–267, 1999.

[10] H. Ng, W. Goh, and K. Low. Feature selection, perceptron learning, and a usability case study for text categorization. *ACM SIGIR Conference on Research and Developement in Information Retrieval*, pages 67–73, 1997.

[11] V. Rijsbergen. *Information Retrieval.* Butterworths, London, 1979.

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

[13] A. Singhal, M. Mitra, and C. Buckley. Learning routing queries in a query zone. In *Proceedings of SIGIR-97, 20th ACM International Conference on Research and Development in Information Retrieval*, pages 25–32, Philadelphia, US, 1997.

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

[15] Y. Yang and J. Pedersen. A comparative study on feature selection in text categorization. *The Fourteenth International Conference on Machine Learning*, pages 412–420, 1997.

[16] J. Zhang and Y. Yang. Robustness of regularized linear classification methods in text categorization. *ACM SIGIR Conference on Research and Developement in Information Retrieval*, 2003.

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

Table 7: The maximum IG, CHI and ORS values of positive (+) and negative (-) features for each of 58 categories.

category id	maxIG		maxCHI		maxORS	
	+	-	+	-	+	-
3	.057	.022	1661	216	30.9	31.8
4	.086	.014	3696	153	44.7	29.3
5	.101	.011	3194	98	54.6	28
6	.082	.016	1950	148	36.9	32.6
7	.056	.014	1569	137	27.8	26.8
8	.105	.009	6334	84	76.3	26.4
9	.05	.008	2833	75	55.8	21
10	.061	.006	4321	58	45.3	20.2
11	.022	.005	891	52	41.3	17.9
12	.043	.005	1762	56	33.4	21.5
13	.067	.005	5929	46	71.8	17
14	.04	.004	3579	43	55.3	16.9
15	.069	.005	6776	43	133.3	20
16	.03	.004	2921	42	50.5	19
17	.05	.003	4464	34	95.6	14.7
18	.025	.003	2983	34	58.4	17.8
19	.043	.003	4800	28	68.7	13.3
20	.024	.003	1414	29	46.5	16.6
21	.013	.003	1633	29	52.9	16.6
22	.03	.002	2520	24	43.7	13
23	.02	.003	1577	42	44.9	15.9
24	.04	.002	6718	21	102.9	14.2
25	.019	.002	796	21	51.9	14.2
26	.019	.002	3373	19	56.7	13.4
27	.033	.001	6075	11	118.8	9.8
28	.028	.002	5104	23	73.1	12.8
29	.021	.002	1470	23	67.3	12.7
30	.011	.002	486	17	40	12
31	.019	.001	2325	13	71.4	8.8
32	.027	.001	5600	12	111.9	10.1
33	.026	.001	5412	14	109.5	11.3
34	.02	.001	3564	12	54.4	10.3
35	.025	.002	5122	20	88	8.4
36	.018	.001	4274	12	69.1	10
37	.024	.001	5335	12	88.6	10
38	.012	.0009	2782	9	57.5	8.5
39	.022	.001	5541	12	111	10
40	.019	.0009	6425	9	124.1	8.2
41	.008	.0009	1432	9	40.2	8.4
42	.014	.0008	3251	8	87	7.9
43	.016	.0005	5256	6	107.4	5.5
44	.005	.0005	1551	5	69.7	5.2
45	.008	.001	1622	12	40.3	7.3
46	.014	.001	5418	13	91	10.6
47	.014	.0007	4988	8	104.3	7.1
48	.009	.0006	3673	7	84.1	6.5
49	.005	.0006	1181	6	62.3	6.5
50	.013	.0006	5400	6	108.9	6.5
51	.004	.0004	2019	4	68.9	4.3
52	.007	.0004	2012	4	63.4	4
53	.006	.0005	2214	5	55.3	5.1
54	.005	.0005	1108	5	64.6	5.8
55	.008	.0004	1191	4	63.6	4.8
56	.007	.0005	4778	5	101.7	5.5
57	.005	.0009	466	10	42.8	9
58	.007	.0007	4081	8	89.9	7.7
59	.005	.0007	2153	8	55.5	8
60	.004	.0004	2483	4	71	4.4