Proceedings of International Joint Conference on Neural Networks, Orlando, Florida, USA, August 12-17, 2007

Large Scale Imbalanced Classification with Biased Minimax Probability Machine

Xiang Peng and Irwin King

***Abstract*— The Biased Minimax Probability Machine (BMPM) constructs a classifier which deals with the imbalanced learning tasks. It provides a worst-case bound on the probability of misclassification of future data points based on reliable estimates of means and covariance matrices of the classes from the training data samples, and achieves promising performance. In this paper, we apply the biased classification model to large scale imbalanced classification problem, and develop a critical extension to train the BMPM efficiently which is a novel training algorithm based on Second Order Cone Programming (SOCP). By removing some crucial assumptions in the original solution to this model, we make the new method more accurate and efficient. We outline the theoretical derivatives of the biased classification model, and reformulate it into a SOCP problem which could be efficiently solved with global optima guarantee. We evaluate our proposed SOCP-based BMPM ($BMPM_{SOCP}$) scheme in comparison with traditional solutions on text classification tasks where negative training documents significantly outnumber the positive ones. Empirical results have shown that our method is more effective and robust to handle imbalanced classification problems than traditional classification approaches.**

I. Introduction

Biased classifiers have many applications [5], [8], [12], [13]. The goal of constructing a two-category biased classifier is to make the accuracy of the important class, instead of the overall accuracy, as high as possible, while maintaining the accuracy of the less important class at an acceptable level. Recently, a novel biased classification model, Biased Minimax Probability Machine (BMPM), provides a worst-case bound on the probability of misclassification of future data points based on reliable estimates of means and covariance matrices of the classes from the training data points and achieves promising performance [4], [6].

In this paper, we extend the model of BMPM to the problem of large scale imbalanced classification, and propose a new training algorithm to tackle the complexity and accuracy issues in BMPM learning task. This model is transformed into a Second Order Cone Programming (SOCP) problem instead of a Fractional Programming (FP) one. Under this new proposed framework, the large scale imbalanced classification problem could be modelled and solved efficiently.

The rest of this paper is organized as follows. Section II reviews the concept of Biased Minimax Probability Machine (BMPM) and related work on it. Section III presents an effective learning algorithm based on Second Order Cone Programming for BMPM. Section IV presents the results of our empirical study on the derived learning scheme. Conclusion and future work are given in Section V.

II. Biased Minimax Probability Machine

In this section, we present the biased minimax framework, designed to achieve the goal of the imbalanced classification. We first introduce the model definition of linear Biased Minimax Probability Machine (BMPM), and then review the original method to solve the optimization.

A. Problem Definition

We assume two random vectors $\mathbf{x}$ and $\mathbf{y}$ represent two classes of data with mean and covariance matrices as $\{\overline{\mathbf{x}}, \Sigma_{\mathbf{x}}\}$ and $\{\overline{\mathbf{y}}, \Sigma_{\mathbf{y}}\}$, respectively in a two-category classification task, where $\mathbf{x}$, $\mathbf{y}$, $\overline{\mathbf{x}}$, $\overline{\mathbf{y}} \in \mathbb{R}^n$, and $\Sigma_{\mathbf{x}}$, $\Sigma_{\mathbf{y}} \in \mathbb{R}^{n\times n}$. For convenience, in the following, we use $\mathbf{x}$ and $\mathbf{y}$ to represent the corresponding class of the $\mathbf{x}$ data and the $\mathbf{y}$ data respectively.[1]

Assuming $\{\overline{\mathbf{x}}, \Sigma_{\mathbf{x}}\}$, $\{\overline{\mathbf{y}}, \Sigma_{\mathbf{y}}\}$ for two classes of data are reliable, Biased Minimax Probability Machine (BMPM) attempts to determine the hyperplane $\mathbf{a}^T\mathbf{z} = b$ ($\mathbf{a} \neq \mathbf{0}$, $\mathbf{z} \in \mathbb{R}^n$, $b \in \mathbb{R}$) with $\mathbf{a}^T\mathbf{z} > b$ being considered as class $\mathbf{x}$ and $\mathbf{a}^T\mathbf{z} < b$ being judged as class $\mathbf{y}$ to separate the important class of data $\mathbf{x}$ with a maximal probability while keeping the accuracy of less important class of data $\mathbf{y}$ acceptable. We formulate this objective as follows:

$$\begin{array}{cc} \max\limits_{\alpha,\beta,b,\mathbf{a}\neq\mathbf{0}} & \alpha \\ s.t. & \inf\limits_{\mathbf{x}\sim(\overline{\mathbf{x}},\Sigma_{\mathbf{x}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{x} \geq b\} \geq \alpha, \\ & \inf\limits_{\mathbf{y}\sim(\overline{\mathbf{y}},\Sigma_{\mathbf{y}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{y} \leq b\} \geq \beta, \\ & \beta \geq \beta_0, \end{array} \tag{1}$$

where α and β represent the lower bounds of the accuracy for future data classification, namely, the worst-case accuracy. Meanwhile, β_0 is a pre-specified positive constant which represents an acceptable accuracy for the less important class.

This optimization will maximize the accuracy for the biased class $\mathbf{x}$ (the probability α) while maintaining the class $\mathbf{y}$'s accuracy at an acceptable level by setting a lower bound β_0 as indicated in the third constraint of optimization problem (1). The hyperplane $\mathbf{a}^{*T}\mathbf{z} = b^*$ given by the solution of this optimization will favor the classification of the important class $\mathbf{x}$ over the class $\mathbf{y}$, and will be more suitable in handling biased classification tasks.

Xiang Peng and Irwin King is with Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong (phone: 852-2609-8431; fax: 852-2603-5024; email: {xpeng, king}@cse.cuhk.edu.hk).

[1] The reader may refer to [9] for a more detailed and complete description.

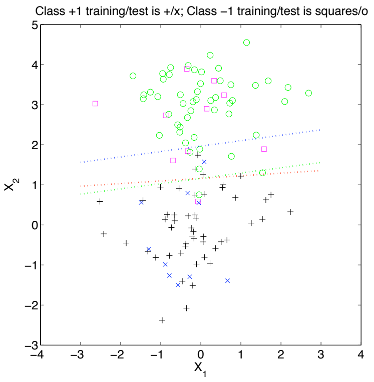

Fig. 1. Decision lines comparison: MPM decision line (dotted red line), BMPM decision line (dotted green line), SVM decision line (dotted blue line).

B. Solving the Biased Minimax Probability Machine

In order to give a comprehensive comparison between our proposed strategy and its original solution, we present the solvability of this optimization problem here. According to the research effort by Huang [7], [8], we adopt Lemma 1 from [10], and obtain the following transformed optimization problem by using the lemma:

$$\max_{\alpha,\beta,b,\mathbf{a}\neq\mathbf{0}} \quad \alpha \tag{2}$$

$$\text{s.t.} \quad -b+\mathbf{a}^T\overline{\mathbf{x}} \geq \kappa(\alpha)\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}\ , \tag{3}$$

$$b-\mathbf{a}^T\overline{\mathbf{y}} \geq \kappa(\beta)\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}\ , \tag{4}$$

$$\beta \geq \beta_0\ , \tag{5}$$

where $\kappa(\alpha)=\sqrt{\frac{\alpha}{1-\alpha}}$, $\kappa(\beta)=\sqrt{\frac{\beta}{1-\beta}}$.

From constraints (3) and (4), we eliminate b from this optimization problem. Without considering the influence of magnitude of $\mathbf{a}$ on the optimal solution for the above problem, we set $\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}})=1$. In addition, since $\kappa(\alpha)$ increases monotonically with α, maximizing α is equivalent to maximizing $\kappa(\alpha)$. Thus the problem can be finally transformed to the Fractional Programming problem

$$\max_{\mathbf{a}\neq\mathbf{0}} \quad \frac{1-\kappa(\beta_0)\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}}{\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}} \tag{6}$$

$$\text{s.t.} \quad \mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}})=1\ , \tag{7}$$

$$\kappa(\beta)\geq\kappa(\beta_0)\ , \tag{8}$$

where the objective function is a linear function with respect to $\kappa(\beta)$, and $\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}$ is a positive term.

In the earlier work of this model, Rosen Gradient projection method [2] is employed to find the solution of this concave-convex FP problem. Furthermore it is observed the inequalities in (3, 4) will become equalities at the optimal point. The optimal b will thus be obtained by

$$b^*=\mathbf{a}^{*T}\overline{\mathbf{y}}+\kappa(\beta_0)\sqrt{\mathbf{a}^{*T}\Sigma_{\mathbf{y}}\mathbf{a}^*}=\mathbf{a}^{*T}\overline{\mathbf{x}}-\kappa(\alpha^*)\sqrt{\mathbf{a}^{*T}\Sigma_{\mathbf{x}}\mathbf{a}^*}$$

III. Efficient BMPM Training

In this section, we present our research effort on the efficient training issue on BMPM model.

A. Motivation

Biased Minimax Probability Machine (BMPM) has been extensively studied as a state-of-the-art learning techniques in various areas, such as bioinformatics [7], [8], information retrieval [12], [13] and statistical learning [5]. Most of recent studies on BMPM are generally based on the Fractional Programming problem (we name it $BMPM_{FP}$) which could be solved by Rosen Gradient method. However the problem reformulation has some crucial assumption which would lead to failure of the model solution. Another issue is that when applying the Fractional Programming based $BMPM_{FP}$ into large real-world classification problems, it would be very sensitive to data dimension and very time consuming.

Motivated from the serious defects of FP-based BMPM solution, we formulate the model into a Second Order Cone Programming (SOCP) problem without any loss of model information. Based on the efforts, the BMPM could be efficiently trained and applied into large scale learning problems.

B. Proposed Strategy

Our main result is stated below.

Theorem 1: If $\overline{x}=\overline{y}$, then the minimax probability decision problem (1) does not have a meaningful solution: the optimal worst-case misclassification probability that we obtain is 1 - a^* =1. Otherwise, an optimal hyperplane $H(a^*,b^*)$ exists, and can be determined by solving the convex optimization problem:

$$\begin{aligned} \min_{t,\mathbf{a}\neq\mathbf{0}} \quad & t-\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}}) \\ s.t. \quad & \|\ \Sigma_{\mathbf{x}}^{\frac{1}{2}}\mathbf{a}\ \| \leq 1, \\ & \|\ \Sigma_{\mathbf{y}}^{\frac{1}{2}}\mathbf{a}\ \| \leq \sqrt{\frac{1-\beta_0}{\beta_0}}t, \end{aligned} \tag{9}$$

and setting b to the value

$$b^*=\mathbf{a}^{*T}\overline{\mathbf{y}}+\kappa(\beta_0)\sqrt{\mathbf{a}^{*T}\Sigma_{\mathbf{y}}\mathbf{a}^*}=\mathbf{a}^{*T}\overline{\mathbf{x}}-\kappa(\alpha^*)\sqrt{\mathbf{a}^{*T}\Sigma_{\mathbf{x}}\mathbf{a}^*},$$

where $\mathbf{a}^*$ is the optima of (9), and $t\in\mathbb{R}$ is a new optimization variable. The optimal worst-case misclassification probability for class $\mathbf{x}$ and $\mathbf{y}$ is

$$\mathbf{Pr}(Misclassification_{\mathbf{x}})=1-\alpha^*, \tag{10}$$

$$\mathbf{Pr}(Misclassification_{\mathbf{y}})=1-\beta_0, \tag{11}$$

respectively. Furthermore, if either $\Sigma_{\mathbf{x}}$ or $\Sigma_{\mathbf{y}}$ is positive definite, the optimal hyperplane is unique.

Proof: It is observed that the optimization problem of (1) could be transformed to the following format:

$$\begin{aligned} \max_{\alpha,b,\mathbf{a}\neq\mathbf{0}} \quad & \alpha \\ \text{s.t.} \quad & \inf_{\mathbf{x}\sim(\overline{\mathbf{x}},\Sigma_{\mathbf{x}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{x}\geq b\}\geq\alpha, \\ & \inf_{\mathbf{y}\sim(\overline{\mathbf{y}},\Sigma_{\mathbf{y}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{y}\leq b\}\geq\beta_0. \end{aligned} \tag{12}$$

By using Lemma 1 in [10], the above optimization becomes:

$$\begin{array}{ll}\max\limits_{\alpha,\mathbf{a}\neq\mathbf{0}} & \alpha \\ \text{s.t.} & \sqrt{\frac{\alpha}{1-\alpha}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}+\sqrt{\frac{\beta_0}{1-\beta_0}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}\leq\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}}).\end{array}$$

Since $\sqrt{\frac{\alpha}{1-\alpha}}$ is a monotonic increasing function of α, we can change variables and rewrite our problem as

$$\begin{array}{ll}\max\limits_{\alpha,\mathbf{a}\neq\mathbf{0}} & \sqrt{\frac{\alpha}{1-\alpha}} \\ \text{s.t.} & \sqrt{\frac{\alpha}{1-\alpha}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}+\sqrt{\frac{\beta_0}{1-\beta_0}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}\leq\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}}).\end{array}$$

Considering $\Sigma_{\mathbf{x}}$ and $\Sigma_{\mathbf{y}}$ can be viewed as positive definite matrices, we formulate the optimization as following:

$$\begin{array}{ll}\max\limits_{\alpha,\mathbf{a}\neq\mathbf{0}} & \sqrt{\frac{\alpha}{1-\alpha}} \\ \text{s.t.} & \sqrt{\frac{\alpha}{1-\alpha}}\leq\frac{\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}})-\sqrt{\frac{\beta_0}{1-\beta_0}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}}{\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}},\end{array}$$

which allow us to eliminate $\sqrt{\frac{\alpha}{1-\alpha}}$,

$$\max_{\mathbf{a}\neq\mathbf{0}}\ \frac{\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}})-\sqrt{\frac{\beta_0}{1-\beta_0}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}}{\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}}. \tag{13}$$

It is observed that optimization problem (13) is equivalent to bound the denominator to 1, and then maximize its numerator. Otherwise if the denominator has no bound, we would have no way to get the optimal solution[2]. Furthermore maximization of a item is equivalent to minimize its opponent. Hence, we could obtain the transformed problem as

$$\begin{array}{ll}\min\limits_{\mathbf{a}\neq\mathbf{0}} & -\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}})+\sqrt{\frac{\beta_0}{1-\beta_0}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}} \\ \text{s.t.} & \sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}\leq 1.\end{array} \tag{14}$$

And it could be further transformed to

$$\begin{array}{ll}\min\limits_{t,\mathbf{a}\neq\mathbf{0}} & t-\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}}) \\ \text{s.t.} & \sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}\leq 1, \\ & \sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}\leq\sqrt{\frac{1-\beta_0}{\beta_0}}t.\end{array} \tag{15}$$

It is exactly a Second Order Cone Programming problem in the form of:

$$\begin{array}{ll}\min\limits_{t,\mathbf{a}\neq\mathbf{0}} & t-\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}}) \\ \text{s.t.} & \|\ \Sigma_{\mathbf{x}}^{\frac{1}{2}}\mathbf{a}\ \|\leq 1, \\ & \|\ \Sigma_{\mathbf{y}}^{\frac{1}{2}}\mathbf{a}\ \|\leq\sqrt{\frac{1-\beta_0}{\beta_0}}t.\end{array} \tag{16}$$

The above problem is convex, feasible, and its objective is linear, therefore there exists an optimal point, $\mathbf{a}^*$. The linearity of the objective function which is strict convex implies that the optimal point is unique. This ends our proof of the theorem. ∎

Lemma 1 The Second Order Cone Programming problem with linear objective function and norm constraints is a convex optimization problem and thus is solvable.

Proof: This can be directly observed from the properties of convex optimization. ∎

Many methods or packages can be used to solve this problem. For example, SeDuMi can solve this problem efficiently with global optima guarantee [14].

C. Kernelized Biased Minimax Probability Machine and Its Solution

We use the kernelization technique to map the n-dimensional data points into a high-dimensional feature space $\mathbb{R}^f$, in which a linear classifier corresponds to a nonlinear hyperplane in the original space.

Assuming the training data points are represented by $\{\mathbf{x}_i\}_{i=1}^{N_{\mathbf{x}}}$ and $\{\mathbf{y}_j\}_{j=1}^{N_{\mathbf{y}}}$ for class $\mathbf{x}$ and class $\mathbf{y}$, respectively, we can formulate the kernel mapping as:

$$\begin{array}{l}\mathbf{x}\rightarrow\varphi(\mathbf{x})\sim(\overline{\varphi(\mathbf{x})},\mathbf{\Sigma}_{\varphi(\mathbf{x})})\ , \\ \mathbf{y}\rightarrow\varphi(\mathbf{y})\sim(\overline{\varphi(\mathbf{y})},\mathbf{\Sigma}_{\varphi(\mathbf{y})})\ ,\end{array}$$

where $\varphi:\mathbb{R}^n\rightarrow\mathbb{R}^f$ is a mapping function. The corresponding linear classifier in $\mathbb{R}^f$ is $\mathbf{a}^T\varphi(\mathbf{z})=b$, where $\mathbf{a}$, $\varphi(\mathbf{z})\in\mathbb{R}^f$ and $b\in\mathbb{R}$. Similarly, the transformed SOCP optimization in BMPM can be written as:

$$\begin{array}{ll}\min\limits_{t,\mathbf{a}\neq\mathbf{0}} & t-\mathbf{a}^T(\overline{\varphi(\mathbf{x})}-\overline{\varphi(\mathbf{y})}) \\ \text{s.t.} & \|\ \mathbf{\Sigma}_{\varphi(\mathbf{x})}^{\frac{1}{2}}\mathbf{a}\ \|\leq 1, \\ & \|\ \mathbf{\Sigma}_{\varphi(\mathbf{y})}^{\frac{1}{2}}\mathbf{a}\ \|\leq\sqrt{\frac{1-\beta_0}{\beta_0}}t.\end{array} \tag{17}$$

To make the kernel work, we represent the final decision hyperplane and the optimization into a kernel form, $K(\mathbf{z}_1,\mathbf{z}_2)=\varphi(\mathbf{z}_1)^T\varphi(\mathbf{z}_2)$, namely an inner product form of the mapping data points. Due to the restriction of paper space, we will not present a detailed kernelization procedure here. It's a similar way as described in [9]. Readers interested in the details can refer to [9].

We give out the kernelized optimization function for Biased Minimax Probability Machine as follows:

$$\begin{array}{ll}\min\limits_{t,\mathbf{a}\neq\mathbf{0}} & t-\mathbf{w}^T(\tilde{\mathbf{k}}_{\mathbf{x}}-\tilde{\mathbf{k}}_{\mathbf{y}}) \\ \text{s.t.} & \sqrt{\frac{1}{N_{\mathbf{x}}}\mathbf{w}^T\tilde{\mathbf{K}}_{\mathbf{x}}^T\tilde{\mathbf{K}}_{\mathbf{x}}\mathbf{w}}\leq 1, \\ & \sqrt{\frac{1}{N_{\mathbf{y}}}\mathbf{w}^T\tilde{\mathbf{K}}_{\mathbf{y}}^T\tilde{\mathbf{K}}_{\mathbf{y}}\mathbf{w}}\leq\sqrt{\frac{1-\beta_0}{\beta_0}}t,\end{array} \tag{18}$$

which is a Second Order Cone Program (SOCP) that has the similar form as the SOCP in (9) and can thus be solved in a similar way.

Remark. We omit the introduction of some notations here due to the space limitations. Interested readers could refer to [7].

IV. Experimental Results

In this section we discuss the experimental evaluation of our proposed biased learning algorithm in comparison to the state-of-the-art approaches. For a consistent evaluation, we conduct our empirical comparisons on two standard datasets for text document classification: Reuters-21578 dataset, and 20-Newsgroup data collection. For both datasets, the same

[2] This is a common technique to tackle optimization problems

class	number of total samples
earn	3964
acq	2369
money-fx	717
grain	582
crude	578
trade	485
interest	478
wheat	283
ship	286
corn	237

TABLE I

AN OVERVIEW OF REUTERS-21578 DATASET WITH 10 MAJOR CLASSES

data pre-processing procedure is applied: the stopwords and numerical words are removed from the documents, and all the words are stemmed and further converted into the lower cases. In order to remove the uninformative word features for dimension reduction, feature selection is conducted using the Information Gain criterion [15].

A. Experimental Testbeds

1) Reuters-21578 Corpus Dataset: It has been broadly used as a benchmark dataset for evaluating classification algorithms. In our experiments, the ModApte split of the Reuters-21578 is used. There are a total of 10,788 text documents in this collection. Table I shows a list of the 10 most frequent topics contained in the dataset [3]. Due to the scope coverage of this paper, we only consider the binary text classification problem, i.e., justifying a text document as relevant or irrelevant to a particular group without consideration of document be assigned to multiple categories. We conduct 3 groups of evaluations on three predefined classes, i.e., *earn*, *grain* and *ship*, which are considered as the positive classes in each group respectively.

2) 20-Newsgroup Data Collection: The 20-Newsgroup dataset is a collection of approximately 20,000 newsgroup documents, partitioned nearly evenly across 20 different newsgroups. Among these different groups, each one corresponding to a different topic. Some of the newsgroups are very closely related to each other, e.g., *comp.sys.ibm.pc.hardware* vs. *comp.sys.mac.hardware*, while others are highly unrelated, e.g., *talk.politics.guns* vs. *comp.graphics*. Considering this fact, we select 3 out of 20 newsgroups with related topics and define them as the interested class in our study, which is *talk.politics.misc*, *talk.politics.guns* and *talk.politics.mideast*. Apart from that, the others are regarded as uninterested.

B. Experimental Settings

Applying BMPM-based technique in text classification is a very straightforward task, where we just need to assume the interested documents to be the more important class ($\mathbf{x}$) in the biased classification framework while assuming the uninterested ones to be the less important class ($\mathbf{y}$).

For performance measurement, the Receiver Operating Characteristic (ROC) curve analysis is employed as our evaluation metric. The ROC curve plots a series of sensitivities against the corresponding one minus specificities, or the true positive rates versus the false positive rates for short. Moreover, if the ROC curves are generated with good shapes evenly distributed along their length, they can be used to evaluate biased learning algorithms by using the area under the curve. The larger the area under the curves, the higher the sensitivity for a given specificity, and hence the better the method's performance [7].

Two other measurements are used to demonstrate the efficiencies of our proposed model and strategy. They are training time performance and Test-Set Accuracy which consists of three measurements, i.e., Test-Set Accuracy on Class $\mathbf{x}$ ($TSA_{\mathbf{x}}$), Test-Set Accuracy on Class $\mathbf{y}$ ($TSA_{\mathbf{y}}$) and the overall Test-Set Accuracy on both classes (TSA).

To examine the effectiveness and efficiency of the learning model and proposed solving strategy, three reference models are used in our experiments. The first reference model is the Support Vector Machine (SVM)[3] which is a state-of-the-art text classification technique. The second reference model is based on *k*NN[4] which is a traditional classification tool. We also include Minimax Probability Machine (MPM)[5] for performance comparison intention. Finally, BMPM has been conducted based on both FP and SOCP frameworks. By comparing with these three models, we are able to determine the BMPM model is more reliable to handle the imbalanced text classification model, and the advantages of our proposed training strategy.

To implement the SOCP-based BMPM algorithm, we adopt the standard optimization package, i.e., SeDuMe [14] and YALMIP [11], to solve the Second Order Cone Programming problem in our algorithm. The FP-based BMPM framework is based on the Rosen Gradient Projection method described in [7].

C. Performance Evaluation

1) Test-Set Accuracy Comparison: Table II shows the experimental results of Test-Set Accuracy (TSA) performance averaging over 3 groups of evaluation, each of which is associated with a predefined positive class in Reuters-21578 dataset.

First, as listed in the first and the second colummns of Table II, we observe that the performance of the two classifiers, $BMPM_{SOCP}$ and $BMPM_{FP}$, outperform the other three models. Take the parameter α for example, $BMPM_{SOCP}$ and $BMPM_{FP}$ achieves noticeably better performance than MPM, which makes the worst-case (maximum) misclassification probability much lower with the value $1-\alpha$ reduced. Second, we compare the performance of the two BMPM classifiers with the traditional classifiers, i.e., SVM and kNN. The results are listed in the fourth and fifth columns of Table II. We find that the average TSA performance, which is indicated as TSA in the table, of these two learning

[3] http://svmlight.joachims.org/

[4] http://people.revoledu.com/kardi/tutorial/KNN/resources.html

[5] http://cosmal.ucsd.edu/∼gert/publications.html

	$BMPM_{SOCP}$	$BMPM_{FP}$	MPM	SVM	kNN
α	**81.42** ± **0.22** ↑	80.35 ± 0.13 ↑	76.30 ± 0.28	-	-
β	**70.00** ± **0.00**	70.00 ± 0.00	76.30 ± 0.34	-	-
TSA_x	**83.10** ± **0.60** ↑	81.07 ± 0.63 ↑	74.91 ± 0.61	73.23 ± 1.59	71.60 ± 0.38
TSA_y	**72.61** ± **0.84**	74.48 ± 0.69	75.20 ± 0.62	74.60 ± 0.47	69.40 ± 0.60
TSA	**77.85** ± **0.04**	77.70 ± 0.21	75.05 ± 0.37	73.90 ± 0.44	70.50 ± 0.55

TABLE II

LOWER BOUND α AND TEST-SET ACCURACY ON THE REUTER-21578 DATASET (%)

	$BMPM_{SOCP}$	$BMPM_{FP}$	MPM	SVM	kNN
α	**78.41** ± **0.46** ↑	78.20 ± 0.55 ↑	74.62 ± 0.33	-	-
β	**70.00** ± **0.00**	70.00 ± 0.00	74.60 ± 0.39	-	-
TSA_x	**76.20** ± **0.72** ↑	75.40 ± 0.79 ↑	73.40 ± 1.02	54.20 ± 0.49	53.90 ± 0.37
TSA_y	**71.40** ± **1.59**	70.50 ± 1.37	75.81 ± 0.36	79.60 ± 1.13	78.41 ± 0.33
TSA	**73.80** ± **1.35**	72.95 ± 1.26	**74.60** ± **0.37**	66.92 ± 0.64	66.15 ± 0.17

TABLE III

LOWER BOUND α AND TEST-SET ACCURACY ON THE 20-NEWSGROUP DATASET (%)

methods becomes closer than the BMPM models. But for the TSA of the more important class indicated as TSA_x is much lower than BMPM models. For example, the TSA_x of $BMPM_{SOCP}$ is much better than kNN though it shows the shortage in the TSA measurement. Finally, we compare the performance of the proposed Second Order Cone Programming based algorithm, i.e., $BMPM_{SOCP}$, to the Fractional Programming based methodology $BMPM_{FP}$. It is evident that the proposed learning algorithm outperforms its original approach.

In order to evaluate the performance substantially, the classification results of the 20-Newsgroup dataset is listed in Table III. From the experimental results, we can see that our two BMPM models achieve better performances than the other algorithms in most of the cases while the $BMPM_{SOCP}$ generally outperforms the $BMPM_{FP}$ method.

2) ROC Curve Analysis: Note that we do not involve MPM and SVM for comparison here, since it cannot easily generate the ROC curves for SVM and MPM due to their model settings.

It is observed that the $BMPM_{SOCP}$ and $BMPM_{FP}$ perform better than the kNN classifier for the two data collections, since the BMPM curves are above of the kNN method at most cases. In addition, usually not all the portions of the ROC curve are of great interest. In general, those with a small false positive rate and a high true positive rate are most important. In light of this, we show the critical portions of Fig. 3 in Fig. 4 detailedly when the false positive rate is in the range of 0.0 to 0.5 and the true positive rate is in the range of 0.5 to 1.0 respectively. In this critical region, most parts of the ROC curve of BMPMs are above the corresponding curve of kNN model in both datasets along with the $BMPM_{SOCP}$ curve is above the one of $BMPM_{FP}$, which again demonstrated the superiority of the BMPM models and our proposed $BMPM_{SOCP}$ algorithm.

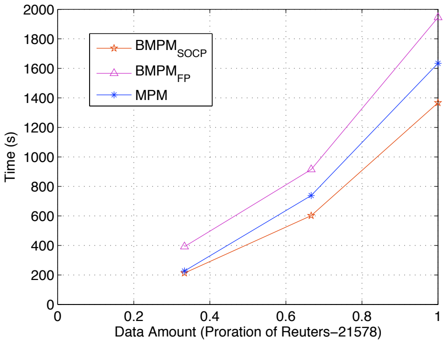

Fig. 2. Training time performance of different models based on Matlab for Three-Phase Reuters-21578 dataset (*sec.*GHz*)

3) Training Time Comparison: We record the runtime when conducting experiments on the Reuters-21578 data collection. We divided the whole dataset into three roughly equivalent portions. We run the experiments three phases stage by stage: first we examine the runtime on one third of the whole dataset; following that we add another one third and record the time consumption; finally we conduct the evaluation on the whole dataset. All these steps are deployed three times given by three predefined positive classes respectively, and we get the averaged performance.

Fig. 2 compares the CPU-time of two $BMPMs$ and MPM on these tasks described above. It could be observed that $BMPM_{SOCP}$ is substantially faster than the other two models on all problems. From the experimental results, we can see that our proposed strategy outperforms its original solution and MPM in training time comparison while MPM is generally faster than $BMPM_{FP}$. We also found that the improvement of our algorithm is more evident comparing with the other two approaches when the size of training

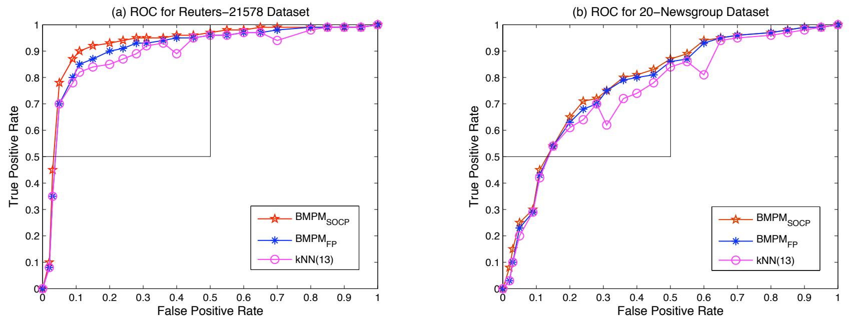

Fig. 3. Full range of the ROC curves on two datasets

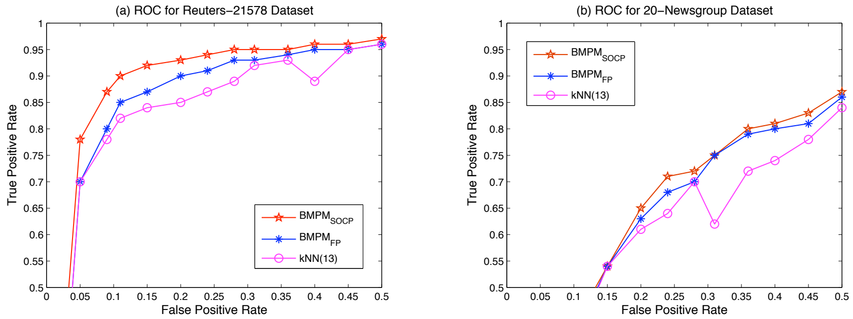

Fig. 4. Crucial part of the ROC curves on two datasets

instances is larger. This is because the larger the size of the problem, the better the performance we could expect. When more examples are conducted, the gap for future improvement begins to increase. As a result, the difference between the two algorithms for BMPM starts to become obvious. It is a crucial point for large scale imbalanced text classification problems. This makes the BMPM conducted on large scale classification problems practical.

V. Conclusion and Future Work

The computational complexity of our method for Biased Minimax Probability Machine (BMPM) is comparable to the quadratic program that one has to solve for the support vector machine (SVM) and Minimax Probability Machine (MPM). While we have viewed this model from the viewpoint of a convex optimization problem, we believe that there is much to gain from exploiting analogies to the SVM and developing specialized optimization procedures for our model. Another direction that we are currently investigating is the extension of our model to multiway classification.

Acknowledgments

We thank researchers from Carnegie Mellon University and the University of Massachusetts, Amherst, for sharing the library and documentation of Lemur Project [1]. The work described in this paper is supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK4235/04E) and is affiliated with the Microsoft-CUHK Joint Laboratory for Human-centric Computing and Interface Technologies.

References

[1] The lemur toolkit. http://www.lemurproject.org.
[2] D. P. Bertsekas. *Nonlinear Programming: 2nd Edition.* Athena Scientific, April 2004.
[3] S. C. Hoi, R. Jin, and M. R. Lyu. Large-scale text categorization by batch mode active learning. In *Proc. of WWW*, pages 633–642, 2006.
[4] K. Huang, H. Yang, I. King, and M. Lyu. Learning classifiers from imbalanced data based on biased minimax probability machine. In *Proc. of CVPR*, volume 2, pages 558–563, 2004.
[5] K. Huang, H. Yang, I. King, and M. Lyu. Learning large margin classifiers locally and globally. In *Proc. of ICML*, pages 51–59, 2004.
[6] K. Huang, H. Yang, I. King, and M. Lyu. Imbalanced learning with a biased minimax probability machine. *IEEE Trans. on SMC (Part B)*, 36(4):913–923, 2006.
[7] K. Huang, H. Yang, I. King, and M. Lyu. Maximizing sensitivity in medical diagnosis using biased minimax probability machine. *IEEE Trans. on Biomedical Engineering*, 53(5):821–831, 2006.
[8] K. Huang, H. Yang, I. King, M. Lyu, and L. Chan. Biased minimax probability machine for medical diagnosis. In *Proc. of AMAI*, pages 1103–1110, 2004.
[9] K. Huang, H. Yang, I. King, M. Lyu, and L. Chan. The minimum error minimax probability machine. *Journal of Machine Learning Research*, 5:1253–1286, 2004.
[10] G. Lanckriet, L. E. Ghaoui, C. Bhattacharyya, and M. Jordan. A robust minimax approach to classification. *Journal of Machine Learning Research*, 3:555–582, 2003.
[11] J. Lofberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *Proc. of CACSD*, pages 284–289, 2004.
[12] X. Peng and I. King. Biased minimax probability machine active learning for relevance feedback in content-based image retrieval. In *Proc. of IDEAL*, pages 953–960, 2006.
[13] X. Peng and I. King. Imbalanced learning in relevance feedback with biased minimax probability machine for image retrieval tasks. In *Proc. of ICONIP*, pages 342–351, 2006.
[14] J. F. Sturm. Using sedumi 1.02: A matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11:625–653, 1999.
[15] Y. Yang and J. Pedersen. A comparative study on feature selection in text categorization. In *Proc. of ICML*, pages 412–420, 1997.