Biased Classification for Relevance Feedback in Content-based Image Retrieval

PENG, Xiang

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Master of Philosophy
in
Computer Science and Engineering

May 2007

Abstract of thesis entitled:

Biased Classification for Relevance Feedback in Content-based Image Retrieval

Submitted by PENG, Xiang

for the degree of Master of Philosophy

at The Chinese University of Hong Kong in May 2007

Content-based Image Retrieval (CBIR) is a very popular research topic in Information Retrieval and Pattern Recognition. Although extensive studies have been conducted, it is still a difficult task to find the user preferred images from image databases. Relevance feedback, as an alternative and more powerful technique for iterative image retrieval, has been investigated in recent years. Many statistical learning techniques have been employed to solve the relevance feedback problem. However, all these techniques are typically used as binary classifiers in which the imbalanced input dataset problem is ignored. In other words, they do not consider the imbalanced dataset problem in relevance feedback where the irrelevant images extremely outnumber the relevant images. This imbalanced dataset setting would lead the positive information (relevant images) to be overwhelmed by the negative information (irrelevant images). Furthermore, how to reduce the number of iterations in order to achieve the optimal boundary between relevant and irrelevant images during the learning procedure is also a critical problem for image retrieval from large datasets.

In this dissertation, we first apply the framework of Biased Minimax Probability Machine (BMPM) to the relevance feed-

back task in CBIR. Conventional relevance feedback techniques typically treat equally with relevant and irrelevant images. However, the irrelevant instances often overnumber the relevant instances practically. In order to handle the imbalanced problem, we present a BMPM-based methodology to capture the user's preference in the relevance feedback process. The proposed scheme is evaluated against both synthetic and real-world datasets, and promising results are obtained.

Second, we extend our BMPM learning strategy with active learning to deal with the iteratively learning problem. Traditional CBIR systems usually need a number of feedback iterations to achieve a satisfactory performance in the learning procedure. In order to overcome the problem, an active learning framework with BMPM is proposed in this thesis. The suggested scheme is validated by promising experimental results.

Finally, an efficient training algorithm for BMPM based on Second Order Cone Programming (SOCP) is presented in order to tackle the large scale learning problems. Experimental results are reported to demonstrate the effectiveness of the suggested algorithm.

偏差分類在基於內容的圖像檢索中相關性反饋技術之應用

摘要

基於內容的圖像檢索(CBIR)是一個非常普遍的研究題目，並且在電腦領域吸引了大量的研究興趣。雖然到目前爲止進行了廣泛的研究，從圖像資料庫中檢索到渴望的圖像仍然是一個困難和開放的問題。相關性反饋，作為重複圖像檢索的一個更加可行和強大的技術，近年來被密集地研究了。許多統計學習技術被用來解決相關性反饋問題。但是，所有這些技術典型地被用作對稱的二進制分類器而輸入資料集之非對稱問題被忽略。換句話說，他們沒有考慮在相關性反饋中的不對稱資料集的問題，毫不相關的圖像在數量上遠遠超過相關的圖像。這個非對稱資料集設置會使得正面資訊被消極資訊所淹沒。此外，怎麼減少為了達到相關和毫不相關的圖像之間優選的界限所需的機器學習疊代的數量是從大資料集中進行圖像檢索的一個重要問題。

在這份論文中，我們首先應用偏差最大最小概率機(BMPM)去對付 CBIR 中的非對稱學習問題。正規的相關性反饋技術通常會平衡的處理相關和無關的訓練樣本。然而， 在實際的相關性反饋任務裏，無關的訓練樣本數目通常要遠超出相關樣本的數目。爲了處理這個非對稱的學習問題，我們提議在相關性反饋過程中使用基於 BMPM 的方法來獲取用戶的首選項。在模擬數據集和真實世界數據集上的實驗結果表明我們提出的方案能有效地改善檢索的性能。

其次，我們擴大基於 BMPM 的學習策略並以活躍學習理論來應付重復學習問題。在相關性反饋過程中，傳統的 CBIR 系統通常需要很大數量的反饋疊代以達到優選的界限。爲了克服這個問題，在這份論文中我們提議一個集成了活躍學習理論和 BMPM 的學習框架，並且通過了實驗核實其有效性。

最後，我們提出一個基於二次錐規劃(SOCP)的高效 BMPM 訓練算法，以用來處理大規模學習問題。頗佳的實驗結果展示了我們所提出方法的有效性。

Acknowledgement

There are many people that I want to thank. First, I would like to take this opportunity to express my gratitude to my supervisor, Prof. Irwin King, for his patient guidance and encouragement during my M.Phil study. The inspiring advices and insightful criticisms from Prof. Irwin King are extremely essential and valuable in my research papers and my thesis. Another professor I want to thank is Prof. Michael Lyu. I am grateful for the valuable suggestions and comments that Prof. Michael Lyu has given to me in reviewing my term papers, term presentations and my thesis. The knowledge I acquired from him is not only beneficial to my research, but also to my whole life. I am also grateful for the help from Prof. M. C. Lee. This dissertation could not be completed without his effort.

I also want to give thanks to my fellow colleagues in the Department of Computer Science and Engineering in The Chinese University of Hong Kong. They helped me in solving technical problems, enlightened me with new research ideas, and gave me encouragement. It is they that have made my two years' research time joyful and wonderful.

My special thanks must to my wife, Jessie Li, who has given me the greatest support and encouragement, so that I can keep concentrated on my postgraduate study.

To Jessie.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Owing to the rapid growth of digital devices, capturing and storing large amounts of multimedia data has become common [82]. Both government and commercial equipments generate gigabytes of image, video, and audio data, or a combination of them [84, 90]. An extremely large amount of information is out there. Growing needs of efficient retrieving, searching and browsing such data is a natural conclusion of the requirement for database systems. Multimedia information retrieval has been a very active research topic in recent years, among which image retrieval has become one of the most important and challenging problems [28, 29, 82]. In image retrieval, there are two research communities, database management and computer vision, studying the same topic from two different viewpoints, text-based and content-based approaches [84, 90].

In a traditional image retrieval system, it uses text keywords or text descriptors for indexing and retrieval. However, there are two main difficulties in keyword-based image retrieval. On the one hand, there are differences in the interpretation of image content. There are always inconsistencies in keyword assignments, since different users may use different keywords to describe the same image concept. On the other hand, large amount of manual effort is required to annotate the images in database.

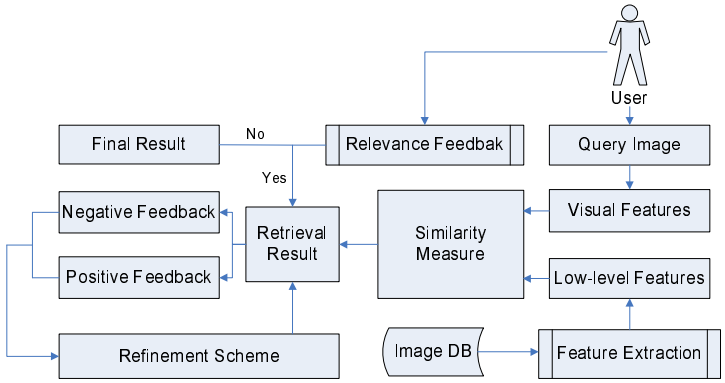


Figure 1.1: A Representative CBIR Framework.

To overcome the difficulties of keyword-based image retrieval approach, Content-based Image Retrieval (CBIR) has been proposed [8, 9]. In contrast to the keyword-based approach, CBIR uses the visual feature of images, such as color, texture, and shape feature, for indexing and retrieval. This greatly reduces the difficulties of the keyword-based approach, since the feature extraction process can be made automatic and the image's own content is always consistent [10]. The CBIR process can be summarized as follows:

1. **Feature Extraction:** Image Processing and Computer Vision techniques are used to extract low-level visual features from images. Image features include color, texture, and shape, etc. These features are usually represented by high-dimensional vectors in the real domain.

2. **Retrieval:** For a given feature, a notation of similarity measure is determined. The similarity measure is used to rank the images in the collection.

Despite the extensive research efforts, the retrieval techniques used in CBIR systems have a very limited recall even when the best feature extraction and similarity measure algorithms are used. That is only a very limited relevant items retrieved to the

user in response to the initial query. This problem is recognized as a major difficulty in information retrieval. There are two major reasons that lead to this problem; they are, (1) the gap between high-level concepts and low-level features; (2) subjectivity of human perception. Therefore, for a particular image, different users or the same user under different circumstances may perceive differently [27, 29]. Thus, it is almost impossible to find a feature extraction or similarity measure algorithm to satisfy all situations.

In light of this, researchers figure out that refinements of the query and similarity measurement during the retrieval process are required to further improve the retrieval performance. Relevance feedback is suggested as a solution for the problem of user subjectivity. The goal of relevance feedback is to learn the users' preference from their interaction, and it is a powerful technique to improve the retrieval result in CBIR. Under this framework, a set of images are presented to the user according to the query. The user marks those images as either relevant or irrelevant and then feeds back this information into the system. Based on this feedback information, the system presents another set of images to the user. The system learns user's preference through this iterative process and improves the retrieval performance. From the experimental results of various CBIR systems, it shows that relevant feedback is a promising direction for CBIR.

1.1 Problem Statement

Most of the current relevance feedback systems are based on the statistical learning approach. Under this framework, the system refines the query and improves the retrieval result by using the feedback information provided by the user.

In the past years, relevance feedback techniques have evolved from early heuristic weighting adjustment techniques to various

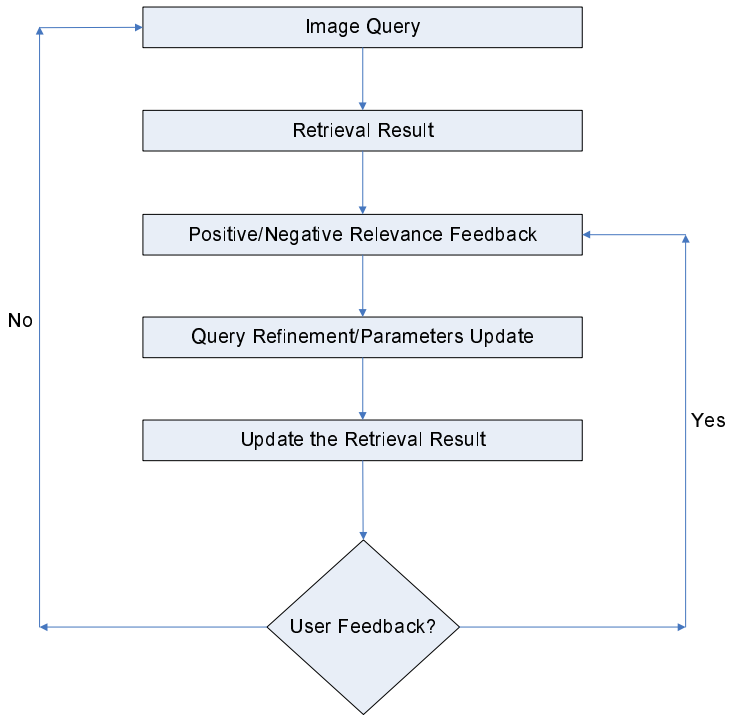


Figure 1.2: Relevance Feedback Flow Chart.

machine learning techniques recently [27, 29, 82, 107]. In [52], Self-Organizing Map (SOM) was proposed to construct the relevance feedback algorithm. Besides the SOM, many popular machine learning techniques were also suggested, such as Decision Tree [60], Artificial Neural Network [90], and Bayesian learning [111], etc. Moreover, many state-of-the-art classification techniques were proposed to attack the relevance feedback, such as Nearest-Neighbor classifiers [105], Bayesian classifiers [13] and Support Vector Machines [29, 32, 107], etc. Typical relevance feedback approaches by these classification models are based on strict binary classifications [29, 32, 105] or one-class classifications [11]. However, the strict binary classifications treat the relevance feedback problem as a strict binary classification problem, and they do not consider the imbalanced dataset problem in relevance feedback, in which the number of irrelevant images is significantly larger than the relevant images. This imbalanced

dataset problem will lead the positive data (relevant images) to be overwhelmed by the negative data (irrelevant images). The one-class technique seems to avoid the imbalanced dataset problem, but the relevance feedback scheme cannot be done properly without the help of negative information [27].

Recently, researchers proposed a novel classification model for imbalanced dataset learning problems, named Biased Minimax Probability Machine (BMPM) [36, 38]. The BMPM constructs a classifier which deals with 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. Our experiments show that the use of BMPM to handle the relevance feedback problem can further improve the retrieval results compared to the traditional techniques. However, since the relevance feedback involves large volume of images, the learning process need a number of iterations to get the optimal boundary. The system users may not have the patience to wait for such a long learning time when they do the image query. How to reduce the number of iterations in the relevance feedback procedure is also a crucial problem. On the other hand, the original solvability of BMPM model has some assumptions, which would lead to the failure of the optimization problem.

Thus, the goals we want to achieve are:

1. To develop a relevance feedback framework that has the advantages of the existing relevance feedback techniques, and is able to address the imbalanced dataset problem.

2. To improve the retrieval result and reduce the number of iterations required during the relevance feedback procedure in Content-based Image Retrieval.

3. To make the solvability of Biased Minimax Probability Ma-

chine more efficient and accurate, and remove the crucial assumptions in its original solution.

1.2 Major Contributions

The main contributions of our work are:

1. **Biased Minimax Probability Machine-based Framework for Relevance Feedback.** We apply the framework of BMPM to the imbalanced learning problem in CBIR. We propose a BMPM-based methodology to capture the user's preference in the relevance feedback process in which BMPM addresses the imbalanced dataset problem. Our strategy is to construct a biased classifier so that the positive examples would not be overwhelmed by the negative examples since the negative ones are extremely larger than the positive ones.

2. **Biased Minimax Probability Machine Active Learning for Relevance Feedback.** Traditional CBIR systems usually need a large number of feedback iterations to achieve the optimal boundary during the learning procedure. How to reduce the iteration number is a crucial problem for retrieving image from large dataset. We propose an active learning framework with BMPM to tackle this problem.

3. **Second Order Cone Program for Biased Minimax Probability Machine.** Training a BMPM on a dataset of huge size with hundreds of thousands of samples is a challenging problem. We propose an efficient algorithm to solve this problem, which reformulate the BMPM framework into a Second Order Cone Program. Our analysis of the proposed algorithm shows that it is more efficient and accurate than its original solution.

Our evaluations show that:

1. BMPM produces better retrieval performance than traditional statistical learning models to handle the imbalance learning task for relevance feedback in CBIR, including Nearest Neighborhood and Support Vector Machines based approaches in the literatures.

2. The retrieval performance of BMPM-based framework can be further improved by incorporating active learning theory.

3. It is more efficient and accurate to solve the BMPM problem for large dataset from the convex optimization angle. Extensive experiments have been conducted to study various appealing properties of the proposed algorithm. Compared with its original solution, the proposed algorithm has a much higher training accuracy, especially on datasets of a huge size with hundreds of thousands of samples.

1.3 Thesis Outline

In the next chapter, we review the current progress of CBIR and relevance feedback research. Furthermore, we present the related works on imbalanced learning, active learning and convex optimization. Chapter 3 presents how the imbalanced learning model, named BMPM, is applied to the problem of relevance feedback. In Chapter 4, we conduct the research work of BMPM active learning on relevance feedback in CBIR. We propose a novel approach for the efficiency and accuracy issues on BMPM for large scale learning problem in Chapter 5. Lastly, we conclude the thesis and describe some potential research directions in Chapter 6.

Each chapter of the thesis is intended to be self-contained. Thus, in some chapters, some definitions, formulas, lemmas,

theorems or illustrative figures that have already appeared in previous chapters, may be briefly reiterated for consistency and completeness.

□ **End of chapter.**

Chapter 2

Background Study

Traditional database management systems are designed for managing numerical and textual data, and searching such data is usually based on straightforward comparisons of text and numerical values. However, the simple method of retrieval is no longer sufficient for the multimedia data, since the digitized representations of image, video, audio, or data themselves do not convey the reality of these media items [10, 103]. As a result, content-based retrieval for multimedia data is given more and more awareness [29, 84, 90]. Implementation of the content-based retrieval facility is not based on a single representation, but is closely related to an underlying data model, a priori knowledge of the area of interest, and the framework for representing queries. In this chapter we survey recent studies on content-based retrieval for multimedia databases. Throughout the discussion, we assume databases that manage only image data though other nontextual information are also in the category of multimedia databases.

2.1 Content-based Image Retrieval

There has seen a rapid growth in the size of digital image collections in recent years. A large number of image data are being

generated by both commercial and individual entities everyday. However, we cannot access or make use of the information unless it is organized so as to allow efficient browsing, searching, and retrieval. With the efforts from two research communities, Database Management and Computer Vision, image retrieval has been a very important research topic since the 1970s. These two research communities study image retrieval from different angles, one being text-based and the other being content-based.

Under the text-based setting, the general framework of image retrieval is to first annotate the images by keyword and then use text-based database management systems to perform image retrieval. Representatives of this approach are [9, 103]. Much research efforts have been pursued along this research direction, such as feature extraction, data modelling, multi-dimensional indexing, and query evaluation. Nevertheless, there exists two major difficulties, especially when the volume of image collections is large with hundreds of thousands samples. One is the huge amount of human labor required in manual image annotation. The other difficulty comes from the rich content in the images and the subjectivity of human perception which is more essential. That is to say, for the same image content different people may perceive it differently. The perception subjectivity and annotation impreciseness may cause unrecoverable mismatches in later retrieval processes.

In the early 1990s, the two difficulties faced by the manual annotation approach when it became more and more crucial owing to the emergence of large scale image collections. In order to tackle these problems, Content-based Image Retrieval (CBIR) was proposed. Under the CBIR framework, images would be indexed by their own visual content, such as color and texture. Since then, many research efforts have been conducted and many image retrieval systems, both for research and commercial, have been developed [66, 73, 83, 95, 96]. This approach has also estab-

lished a general scheme of image retrieval from a new perspective, although there are still many open problems to be handled before such retrieval systems can be put into practice.

There are three fundamental aspects for Content-based Image Retrieval, i.e., *Image Representation*, *High Dimensional Indexing*, and *Image Retrieval Systems Design*. In the following subsections, we will give a brief review of these three parts.

2.1.1 Image Representation

Feature extraction is the fundamental task in Content-based Image Retrieval [84]. Normally features may include both text-based features (keywords) and visual features (color, texture, shape, etc). We confine ourselves in the techniques of visual feature extraction since the key point of feature extraction in Content-based Image Retrieval does not lie in text-based feature extraction [8, 82].

It is obvious that no single best presentation for a particular feature owing to the human perception subjectivity. For any given feature, there exists multiple representatives which characterize the feature from different perspectives.

Color Feature

Color feature, which is independent of image size and orientation and relatively robust to background variety, is one of the most important and widely used visual features in image retrieval [62, 64, 114]. *Color Histogram*, *Color Moments*, and *Color Sets* are three most popular color features in the research communities of Computer Vision and Pattern Recognition.

Color Histogram is the most commonly used color feature representation in image retrieval. It denotes the joint probability of the intensities of the three color channels. Swain and Ballard [101] propose Histogram Intersection, a L_1 metric, as the

similarity measure for the Color Histogram. In order to take into account the similarities between similar but not identical colors, Ioka and Niblack *et al.* [41, 66] introduce a L_2 metric in comparing the histograms. In addition, since most Color Histograms are very sparse and thus sensitive to noise, Stricker and Orengo [97] propose to use the cumulated Color Histogram.

To overcome the quantization effects as in Color Histogram, Stricker and Orengo [97] propose to use *Color Moments* approach. The mathematical foundation can be characterized by the moments. Only the first moment (mean), the second, and third central moments (variance and skewness) are extracted as the color feature representation since most of the information is concentrated on the low-order moments. Weighted Euclidean distance is employed to measure the color similarity.

Smith and Chang [92, 93] propose *Color Sets* as an approximation to Color Histogram in order to facilitate fast search over large scale image collections. They first transform the (R, G, B) color space into a perceptually uniform space (i.e., H, S, V), and then quantize the transformed color space into M bins. A Color Set is defined as a selection of the colors from the quantized color space. Thus a binary search tree can be constructed to allow fast search since Color Set feature vectors are binary.

Texture Feature

Texture refers to the visual patterns that have properties of homogeneity that do not result from the presence of only a single color or intensity, and it is an innate property of virtually all surfaces. It contains important information about the structural arrangement of surfaces and their relationship to the surrounding environment. Extensive research results on this topic have been reported in the past decades due to its importance and usefulness in Computer Vision and Pattern Recognition. Now, it further finds its way in image retrieval tasks.

In [26], Haralick *et al.* propose the *co-occurrence matrix representation* of texture feature. Under this framework, it first constructs a co-occurrence matrix based on the orientation and distance between image pixels, and then extracts meaningful statistics from the matrix as the texture representation. Many other researchers follow their footprint and further propose enhanced versions. For example, Gotlieb and Kreyszig [23] find out that contrast, inverse difference moment and entropy had the biggest discriminatory power.

Tamura *et al.* [102] explore the texture representation from a different viewpoint. They develop computational approximations to the visual texture properties found to be important in psychology studies. The six visual texture properties are *coarseness*, *contrast*, *directionality*, *linelikeness*, *regularity*, and *roughness*. The major distinction between the *Tamura texture representation* and the *co-occurrence matrix representation* is that all the texture properties in Tamura representation are visually meaningful whereas some in co-occurrence matrix representation may not. This makes the Tamura texture representation very attractive in image retrieval, and it has been employed in many real image retrieval systems such as QBIC [17] and MARS [40, 67].

After Wavelet transform was introduced and its theoretical framework was established, many researchers begin to employ Wavelet transform in texture representation. Smith and Chang [94] use the statistics (mean and variance) extracted from the Wavelet subbands as the texture representation. This approach achieves impressive results in real image dataset. Wavelet transform is also associated with other techniques to achieve better performance. Gotlieb and Kreyszig use Wavelet transform to perform texture analysis together with KL expansion and Kohonen maps [23].

Shape Feature

In some image retrieval applications, it requires the shape representation to be invariant to translation, rotation, and scaling. Generally the shape representations can be divided into two categories, boundary-based and region-based. The former uses only the outer boundary of the shape while the latter uses the entire shape region. *Fourier Descriptor* and *Moment Invariants* are the most successful representatives for these two categories.

The core of Fourier Description is to use the Fourier transformed boundary as the shape feature. There are some comprehensive literature reviews on this topic in [74, 123]. In order to take into account the digitization noise in the image domain, Rui *et al.* propose a modified Fourier Descriptor which is both robust to noise and invariant to geometric transformations [85].

The concept of Moment Invariant is to use region-based moments, which are invariant to transformations as the shape feature. Hu [33] identifies seven such moments. Since then, many improved versions emerged based on his work. Yang and Albregtsen [121] propose a fast method of computing moments in binary images based on the discrete version of Green's theorem. Motivated by the fact that most useful invariants were found by extensive experience, Kapur *et al.* develope algorithms to systematically generate and search for a given geometry's invariants [46].

Apart from 2D shape representations, there exist many methods developed for 3D shape representations. Wallace and Wintz [113] present a technique for normalizing Fourier Descriptors which retained all shape information, and was computationally efficient. They also propose to use a hybrid structural/statistical local shape analysis algorithm for 3D shape representation. Furthermore, Taubin and Cooper [104] propose to use a set of Algebraic Moment Invariants to represent both 2D and 3D shapes, which greatly reduced the computation required for shape matching.

Summary

Many visual features have been explored, both previously in Computer Vision applications and currently in Image Retrieval applications as we could see from the above review. For each visual feature, there exist multiple representations, which model the human perception of that feature from different perspectives. There is also a need of developing new frameworks to organize feature representations so that we could conduct efficient retrieval from image databases.

2.1.2 High Dimensional Indexing

Efficient *Multi-dimensional Indexing* techniques need to be explored in order to make the Content-based Image Retrieval scalable to large scale image collections. There are two key challenges in such an exploration for image retrieval: *High Dimensionality,* and *Non-Euclidean Similarity Measurement.* It is observed that Euclidean measurement may not effectively model human perception of a certain visual content [82]. Thus various other similarity measures need to be supported, such as Histogram Intersection, Cosine, Correlation, etc.

Towards overcoming these obstacles, one promising approach is to first perform dimension reduction and then use appropriate multi-dimensional indexing techniques.

Even though the dimension of the feature vectors in image retrieval is normally very high, the embedded dimension is much lower. It is beneficial to perform dimension reduction before we utilize any indexing technique. Two very popular approaches appeared in the literature, say Karhunen-Loeve Transform (KLT) [7] and column-wise clustering [87]. Interested readers may refer to the literatures for more details.

After we identify the embedded dimension of the feature vectors, we need to select appropriate multi-dimensional indexing

algorithms to index the reduced but still high dimensional feature vectors. Currently there are three major research communities contributing in this area, i.e., Computational Geometry, Database Management, and Pattern Recognition. The existing popular multi-dimensional indexing techniques include Bucketing algorithm, k-d tree, priority k-d tree, quad-tree, K-D-B tree, hB-tree, R-tree and its variants R^+-tree and R^*-tree [24, 25, 88]. Furthermore, clustering and Neural Networks which are widely used in Pattern Recognition also find their way in multi-dimensional indexing area.

2.1.3 Image Retrieval Systems Design

Content-based Image Retrieval has become a very active research area recent years [84, 90]. Many image retrieval system have been built for both commercial and research objectives. Most image retrieval systems support one or more of the following options: *Random Browsing*, *Search by Example*, *Search by Sketch*, *Search by Text*. We have been provided a rich set of search options today, but systematic studies involving actual users in practical applications are still a long way off [29, 82]. In the subsequent sections we will review a few representative systems and highlight their distinct characteristics.

QBIC

URL: http://wwwqbic.almaden.ibm.com/

QBIC [66] is developed in IBM and is the first commercial Content-based Image Retrieval system, which stands for Query By Image Content. It has profound significance on later Image Retrieval systems.

In QBIC, it supports queries based on example images, user-constructed drawings and sketches, and selected color and texture patterns, etc. The color feature used in QBIC are the aver-

age (R, G, B), MTM (mathematical transform to Munsell) coordinates, and a k-element color histogram. An improved version of the Tamura texture representation is employed as its texture feature. Its shape feature consists of shape area, circularity, eccentricity and a set of algebraic moments invariants. QBIC is one of the few systems which take into account the high dimensional feature indexing.

Photobook

URL: http://vismod.media.mit.edu/demos/photobook/

Photobook [73] is developed at MIT Media Lab which is a set of interactive tools for browsing and searching images. There are three sub-books in Photobook, from which shape, texture, and face features are extracted respectively. Human can then query each of the three sub-books based on corresponding features.

It is observed that there was no single feature which can best model images from each and every domain. In order to tackle this problem, Picard and Minka [75] proposed to include human in the image annotation and retrieval iterations in the more recent version of Photobook. They also proposed a "society of model" approach to incorporate the human factor due to the subjectiveness of human. Experimental results demonstrated the effectiveness of their approach in interactive image annotation.

VisualSEEk

URL: http://www.ctr.columbia.edu/VisualSEEk/

VisualSEEk [95] is developed at Columbia University, which is a visual feature search engine. Main research features are spatial relationship query of image regions and visual feature extraction from compressed domain.

Color Set and Wavelet Transform based texture feature are

the image features employed in the system. They also developed binary tree based indexing algorithms to speed up the retrieval process. It supports queries based on both visual features and their spatial relationships in VisualSEEk. This enables a user to submit a query by its sketch.

WebSEEk

URL: http://persia.ee.columbia.edu:8008/

WebSEEk [96] is also developed at Columbia University, which is a World Wide Web oriented text/image search engine. It consists of three main modules, i.e., image/video collecting module, subject classification and indexing module, and search, browse and retrieval module. Both keywords and visual content based queries are provided in the system.

MARS

URL: http://jadzia.ifp.uiuc.edu:8000/

MARS [83] is developed at University of Illinois at Urbana Champaign, which stands for Multimedia Analysis and Retrieval System. MARS differs from other image retrieval systems in both the research scope and the techniques used. It is an interdisciplinary research effort involving multiple computer research communities: Computer Vision (CV), Database Management System (DBMS), and Information Retrieval (IR). The system characteristics of MARS are the integration of DBMS and IR, integration of indexing and retrieval, and integration of computer and human. The primary focus of MARS is not on finding a single "best" feature representation, but rather on how to organize various visual features into a helpful retrieval architecture which can dynamically adapt to different applications and different users. Specifically MARS formally proposes a relevance feedback architecture in image retrieval and integrates

such technique at various levels during retrieval.

2.2 Relevance Feedback

Relevance feedback takes advantage of human-machine interaction to refine high-level queries represented by low-level features [13, 29, 69, 70]. It is employed in conventional document retrieval for automatically adjusting an existing query using information fed back from the user. In image retrieval applications, the user selects relevant images from previous retrieved results and provides a preference weight for each relevant image. The weights for the low-level feature, i.e., color and texture, etc., are dynamically updated based on the user's feedback. The user is no longer required to specify a precise weight for each low-level feature to formulate the query model. Based on the feedback, the hight-level concepts implied by the feature weights and relevant feedbacks are automatically refined.

The similarities between the query and those images in the database are computed during the process of relevance feedback. The similarity between an image I in the database and the query is calculated by:

$$S(I) = \sum_f w_f F_f(I), \tag{2.1}$$

where $F_f(I)$ measures the similarity of the image I to the query in feature level (e.g. color, texture, etc). Mahalanobis distance is employed for feature similarity measurement:

$$F_f(I) = (\vec{x}_f - \vec{q}_f)^T C_f^{-1} (\vec{x}_f - \vec{q}_f), \tag{2.2}$$

where $\vec{x}_f$ is the f-th feature vector of the image I, $\vec{q}_f$ is the f-th feature vector of the query and C_f is the covariance matrix of the f-th feature components of the query. $\vec{q}_f$ and C_f are decided

by Eq. (2.3) and Eq.(2.4) respectively

$$\vec{q}_f = \frac{\sum_{k=1}^{N} v_k \vec{m}_{kf}}{\sum_{k=1}^{N} v_k}, \tag{2.3}$$

$$C_f = \frac{\sum_{k=1}^{N} v_k (\vec{m}_{kf} - \vec{q}_f)(\vec{m}_{kf} - \vec{q}_f)^T}{\sum_{k=1}^{N} v_k}, \tag{2.4}$$

where N is the number of relevant images, v_k is the preference weight for the k-th relevant image (positive feedback), and $\vec{m}_{kf}$ is the f-th feature vector of the k-th relevant image.

The low-level feature weight w_f in Eq. (2.1) is updated by:

$$w_f = \frac{\sum_{k=1}^{N} v_k}{\sum_{k=1}^{N} v_k F_f(K)}. \tag{2.5}$$

The concept behind Eq. (2.5) is that: the smaller the average feature distance over the relevant images, the better the feature represents the query concept. Therefore, higher weight is given to the feature that has smaller average feature distance over the relevant images.

Currently, the image retrieval system requires the user to manually provide a preference weight v_k for each relevant image, which denotes the degree of how much the user likes the image. Here only positive examples are used [13]. However, there exist examples that are not desired by the user but closer to the query than some of the relevant images based on the above calculation in some cases. Those examples will be retrieved, and their ranks may remain higher than some relevant image during the whole interactive retrieval process. Hence, it is important to use the information implied by the negative examples. Moreover, expressing the perception subjectivity via providing numerical preference weights is a difficult task for the users from time to time.

2.2.1 Self-Organizing Map in Relevance Feedback

A system named PicSOM involves Self-Organizing Map (SOM) as a relevance feedback technique [52]. The technique introduced in the PicSOM system implements relevance feedback and simultaneously facilitates automatic combination of the responses from multiple Tree Structured SOMs and all their hierarchical levels. This mechanism aims at autonomous adaptation to the user's behaviour in selecting which images resemble each other in the particular sense the user seems to be interested in. In the subsequent section we will first introduce the theory of SOM, and then give a brief introduction of its application in PicSOM.

The Self-Organising Map (SOM) [47] is an unsupervised and self-organising neural algorithm which is widely used to visualize and interpret large high-dimensional datasets. It can be used to visualize multi-dimensional data, usually on a two-dimensional grid. The SOM consists of a two-dimensional lattice of units. A model vector m_i is associated with each map unit i. The map attempts to represent all the available observations x with optimal accuracy by using the map units as a restricted set of models. During the training phase, the models become ordered on the grid so that similar models are close to and dissimilar models far from each other [49].

The fitting of the model vectors is usually carried out by a sequential regression process, where $t = 0, 1, \ldots, t_{max} - 1$ is the step index: For each input sample $x(t)$, first the index $c(x)$ of the Best-Matching Unit (BMU) or the *winner* model $m_{c(x)}(t)$ is identified by the condition

$$\forall i : \| x(t) - m_{c(x)}(t) \| \leq \| x(t) - m_i(t) \| . \tag{2.6}$$

The usual distance metric used here is Euclidean one [48]. After finding the BMU, a subset of the model vectors constituting a neighborhood centered around node $c(x)$ are updated as

$$m_i(t+1) = m_i(t) + h(t; c(x), i)(x(t) - m_i(t)). \tag{2.7}$$

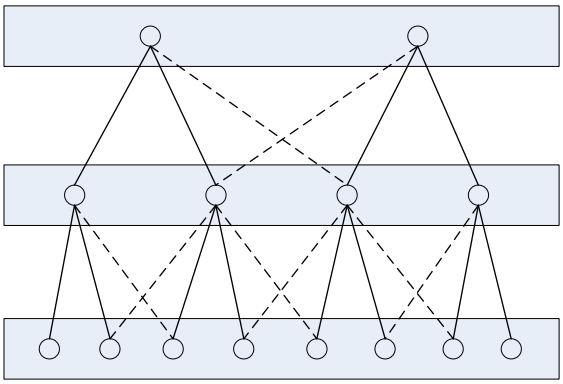


Figure 2.1: The structure of a three-level one-dimensional TS-SOM. The solid lines present parent-child relations and the dash lines represent neighbouring nodes included in the BMU search space [52].

Here $h(t; c(x), i)$ is the 'neighborhood function', a decreasing function of the distance between the ith and $c(x)$th nodes on the map grid.

To speed up the search of the BMU, Koikkalainen [48] introduced a variant of SOM called the Tree Structured Self-Organising Map (TS-SOM). TS-SOM is a tree-structured vector quantization algorithm that uses normal SOMs at each of its hierarchical levels. The structure of a TS-SOM in one-dimensional case with three SOM levels is illustrated in Fig 2.1.

The PicSOM system presents the user in each round of the image query a set of images s/he has not seen before. S/he then marks the relevant images, and the system implicitly interprets the unmarked images as negative ones. Because all the database images have been previously mapped in their best-matching SOM units at the time the SOMs were trained, it is now easy to locate both the positive and negative images on each level of every TS-SOM in use [49]. The map units are scored with a fixed positive value for each positive image mapped in them. Likewise, negative images contribute negative values. These values are selected so that the sum of all positive values equals plus one, and the sum of all negative values equals minus one. The

total sum of all values on each map is thus equal to zero.

The system remembers all image responses the user has given since the query was started. Information on all the images seen and the user's opinions on them thus becomes stored in every single SOM in the system. And this is the point where relevance feedback enters the play. The basic idea is simple: the formation of an SOM brings similar images in nearby map units. If a particular SOM unit has been the best-matching one for many positive images and for none or only few negative ones, it can be deduced that its content coincides with the user's opinion well [47, 51]. By assumption, the neighbouring SOM units are similar to it, and the images mapped in them can likewise be supposed to be relevant for the user.

2.2.2 Decision Tree in Relevance Feedback

In data mining and machine learning, a decision tree is a predictive model; that is, a mapping from observations about an item to conclusions about its target value [63, 79]. More descriptive names for such tree models are classification tree or reduction tree. In these tree structures, leaves represent classifications and branches represent conjunctions of features that lead to those classifications. The machine learning technique for inducing a decision tree from data is called decision tree learning.

Recently, researchers present a relevance feedback retriever that learns decision trees from feedback information [60]. Based on the learned Relevance Feedback Decision Trees (RFDT), inferences are made about which images the user would most like to see on a subsequent retrieval iteration. The retrieval precision increases after only one or two iterations, requiring that the user provide feedback on only a handful of images.

The algorithm behind the Relevance Feedback Decision Tree (RFDT) retriever operates as follows. On the first iteration,

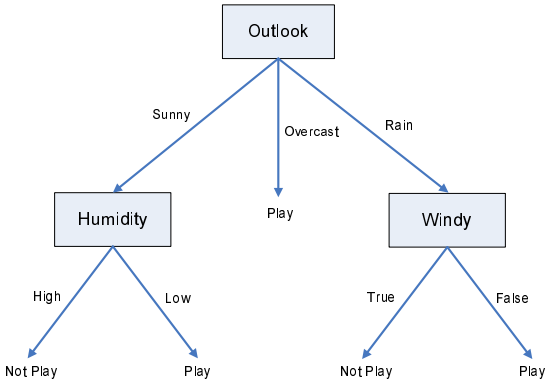


Figure 2.2: A simple Decision Tree illustration. Each internal node tests an attribute, and each branch corresponds to a possible value for that attribute, and each leaf node provides a classification, and each tree path corresponds to a rule.

no feedback information exists, so the retriever performs an unweighted k-Nearest Neighbor retrieval. The user then marks the retrieved images as relevant or irrelevant. This feedback is relayed back to the system and the second iteration begins. On the second iteration, the algorithm is presented with the $k+1$ labeled images. From these $k+1$ training instances, a decision tree is generated via $C4.5$. A decision tree is a method for recursively partitioning a feature space such that each partition is labeled by a single class value. The criteria for making sequential "cuts" in the space is a product of information theory called "entropy" [89]. The algorithm continues to make select cuts until all instances within a partition are of the same class; the partition is then labeled with that class value. Once the tree is formed, it can be used to select the next set of k images to present to the user. To this end, the entire database of feature vectors can be classified via the learned tree. Thus, a leaf has a record of all of the images that have been routed to it. When all instances in the database have been filtered through the decision tree, the instances contained in the leaves labeled with class "relevant" are assembled into a list. From this list,

the k images closest to the query are retrieved by executing an unweighted k-Nearest Neighbor retrieval on the list. On the next iteration, the retriever's operation is identical to that on the second iteration, except that now there are a total of $2k$ instances labeled with user feedback, plus the query, from which the system will induce a decision tree. Thus, each subsequent iteration allows the retriever to learn from k more images than the previous iteration [60]. This process continues until the user becomes satisfied with the result or until the user's patience is expended.

2.2.3 Bayesian Classifier in Relevance Feedback

Considering the vector $\mathbf{x}$ in $\mathbb{R}^n$ that obeys Gaussian distribution, the probability density function of $\mathbf{x}$ is:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{d}{2}} \mid \sum \mid} e^{-\frac{1}{2}(\mathbf{x}-\varepsilon)^T \sum^{-1}(\mathbf{x}-\varepsilon)}, \tag{2.8}$$

where $\mathbf{x} = [x_1, \ldots, x_n]$, $\varepsilon = [\varepsilon(x_1), \ldots, \varepsilon(x_n)]$, and $\sum = \varepsilon\{(\mathbf{x} - u)(\mathbf{x} - u)^T\}$ [16, 99].

We can get the following Bayesian decision boundary function that is the probability of $\mathbf{x}$ belongs to the ith class C_i:

$$g_i(\mathbf{x}) = \lg p_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\varepsilon_i)^T\Sigma_i^{-1}(\mathbf{x}-\varepsilon_i) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln \mid \Sigma_i \mid + \ln P(w_i). \tag{2.9}$$

Bayesian classifier can be used to deal with the feedback process in CBIR, and it treats positive and negative feedback examples with different strategies. For positive examples, a Bayesian classifier is used to determine the distribution of the query space [99]. A 'dibbling' process is applied to penalize images that are near the negative examples in the query and retrieval refinement process. The proposed algorithm also has the progressive learning capability that utilize past feedback information to help the current query.

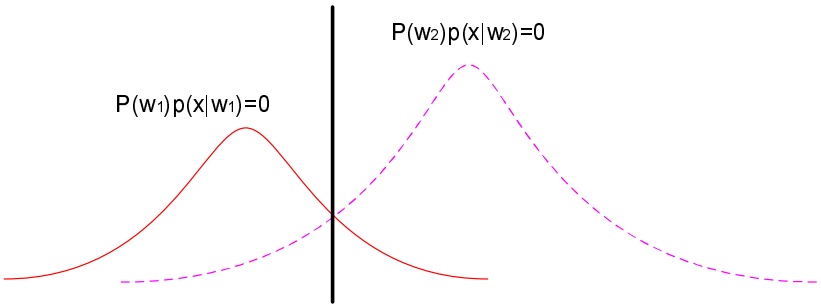


Figure 2.3: The decision boundary of the Bayes classifier is located at the point of intersection of the two curves.

In the algorithm the probabilistic property of each image is used in the relevance feedback process. This property contains the conditional probability of each attribute value given the image and can be updated on the fly by users' feedbacks. It describes a single decision boundary through the feature space. Another key idea is to treat positive and negative examples in the feedback differently in the query refinement process, as positive examples often are semantically similar, while negative examples are not [12, 99].

2.2.4 Nearest Neighbor Search in Relevance Feedback

In Pattern Recognition, the k-Nearest Neighbor algorithm (kNN) is a method for classifying objects based on closest training examples in the feature space [86, 115]. kNN is a type of instance-based learning, or lazy-learning where the function is only approximated locally and all computation is deferred until classification. The training examples are mapped into multi-dimensional feature space. The space is partitioned into regions by class labels of the training samples. A point in the space is assigned to the class c if it is the most frequent class label among the k nearest training samples. Usually Euclidean distance is used [105, 119].

The training phase of the algorithm consists only of storing

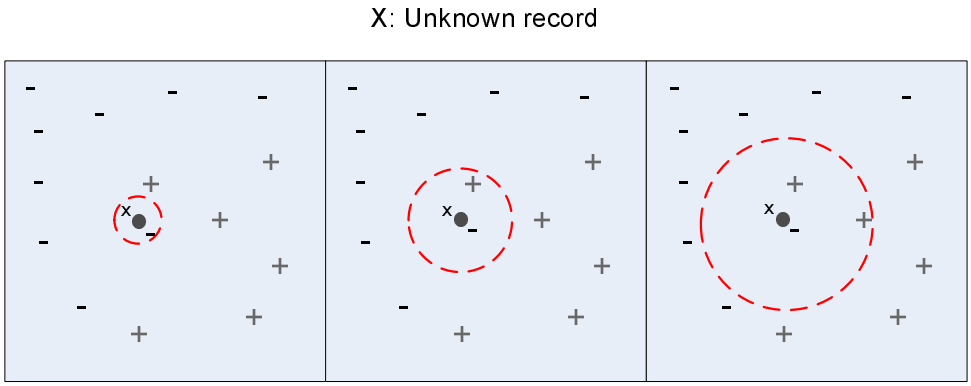


Figure 2.4: k-Nearest Neighbors of a record $\mathbf{x}$ are data points that have the k smallest distance to $\mathbf{x}$. 1-Nearest Neighbor (Left). 2-Nearest Neighbor (Middle). 3-Nearest Neighbor (Right).

the feature vectors and class labels of the training samples. In the actual classification phase, the same features as before are computed for the test sample whose class is not known. Distances from the new vector to all stored vectors are computed and k closest samples are selected. The new point is predicted to belong to the most numerous class within the set.

The best choice of k depends upon the data; generally, larger values of k reduce the effect of noise on the classification, but make boundaries between classes less distinct [2]. A good k can be selected by parameter optimization using, for example, cross-validation. The special case where the class is predicted to be the class of the closest training sample (i.e., when $k = 1$) is called the nearest neighbor algorithm.

The basic idea behind Nearest Neighbor Search for relevance feedback is to constrain the search space for the nearest neighbors for the next iteration using the current set of nearest neighbors [105]. Usually one need compute the nearest neighbors of the query feature vector in the corresponding feature space in order to retrieve images that are similar in texture or color in content-based retrieval. User identifies a set of retrieval examples relevant to the image, and that information is then used

to compute a new set of retrievals. In order to compute the new set of retrievals closer to the user's expectation we modify the similarity metric used in computing the distances between the query and the database items. The distance between two feature vectors is typically calculated as a quadratic distance of the form $(Q-F)^T W(Q-F)$ where Q is a query vector, F is a database feature vector, and W is a positive semi-definite matrix. During each iteration, the weight matrix is updated based on a user's feedback [105]. Given the updated weight matrix, the next set of nearest neighbors is then computed. There are also more recent methods, such as kernel-based ones that appear to be more effective in learning but computationally prohibitive for large scale datasets.

2.2.5 Support Vector Machines in Relevance Feedback

The method of incorporating Support Vector Machines (SVM) into CBIR with relevant feedback is very sound [15, 27, 28]. The information carried by positive and negative examples are used to automatically update preference weights for positive relevant images. This not only releases the users from providing accurate preference weight for each positive relevant image but also utilizes the negative information. Reasonable better results are obtained compared to those of using positive feedbacks only. We first give a brief introduction of SVM, and then describe how it can be applied to Content-based Image Retrieval task.

Support Vector Machine (SVM) [112] is an approximate implementation of the structural risk minimization principle. It creates a classifier with minimized Vapnik-Chervonenkis dimension. SVM minimizes an upper bound on the generalization error rate. The SVM can provide a good generalization performance on pattern classification problems without incorporating problem domain knowledge. Consider the problem of separating

Kernel Function	Inner Product Kernel: $K(\vec{\mathbf{x}}, \vec{\mathbf{x}_i}), i = 1, 2, \ldots, N$
Gaussian RBF	$k(\mathbf{x}, \mathbf{y}) = exp(\frac{-\|\mathbf{x}-\mathbf{y}\|^2}{c})$
Polynomial	$((\mathbf{x} \cdot \mathbf{y}) + \theta)^d$
Sigmoidal	$tanh(k(\mathbf{x} \cdot \mathbf{y}) + \theta)$
Inverse Multiquadric	$\frac{1}{\sqrt{\|\mathbf{x}-\mathbf{y}\|^2+c^2}}$

Table 2.1: Types of Kernel Functions

the set of training vectors belonging to two classes:

$$\{(\vec{x}_i, y_i)\}_{i=1}^{N}, \qquad y_i = +1/-1 \tag{2.10}$$

where $\vec{x}_i$ is an input pattern, and y_i is the label, $+1$ denotes positive example, -1 denotes the negative example. If those two classes are linearly separate, the hyperplane that does the separation can be easily calculated by:

$$\vec{w}^T \vec{x} + b = 0 \tag{2.11}$$

where $\vec{x}$ is an input vector, $\vec{w}$ is a weight vector, and b is a bias. The goal of SVM is to find the parameters $\vec{w}_0$ and b_0 for the optimal hyperplane to maximize the distance between the hyperplane and the closest data point:

$$\vec{w}_0^T \vec{x}_i + b_0 \geq 1 \qquad for\ y_i = +1 \tag{2.12}$$

$$\vec{w}_0^T \vec{x}_i + b_0 < -1 \qquad for\ y_i = -1 \tag{2.13}$$

A linear separable example in 2D is illustrated in Fig. 2.5. If the two classes are non-linearly separable, the input vectors should be nonlinearly mapped to a high-dimensional feature space by an inner-product kernel function $K(\vec{x}, \vec{x}_i)$. Table 2.1 shows four typical kernel functions [32, 65]. An optimal hyperplane is constructed for separating the data in the high-dimensional feature space. This hyperplane is optimal in the sense of being a maximal margin classifier with respect to the training data.

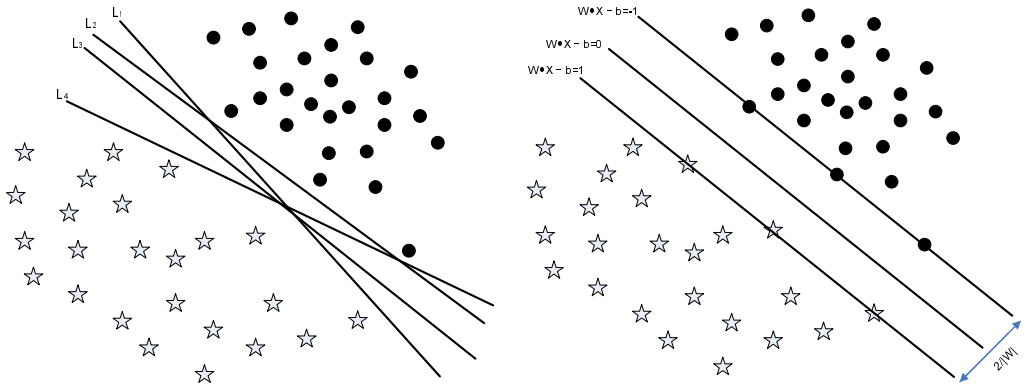


Figure 2.5: Many linear classifiers separate the data. However, only one achieves maximum separation (Left). Maximum-margin hyperplanes for a SVM trained with samples from two classes. Samples along the hyperplanes are called the support vectors. (Right)

Usually the problem to separate the negative examples from the positive examples turns out to be finding a nonlinear classifier. SVM can be used in this task, and it provides a good generalization performance at the same time. Given $\vec{w}_0$ and b_0, the distance of a point $\vec{x}$ from the optimal hyperplane is defined as

$$d(\vec{w}_0, b_0, \vec{x}) = \frac{\mid \vec{w}_0^T \vec{x} + b_0 \mid}{\parallel \vec{w}_0 \parallel}. \tag{2.14}$$

The distance indicates how much an example belonging to one class is different from the other one. These motivate us to use SVM for automatically generating preference weights for relevant images. Intuitively, the farther the positive examples from the hyperplane, the more distinguishable they are from the negative examples. Thus, when we decide their preference weights, they should be assigned with larger weights. Currently, we simply set the relation between the preference weights and the distance as a linear function in the numerical calculation. It can be easily extended to nonlinear relation [32]. During the iterative query procedure, the positive and negative examples selected in the history are collected for learning at each query time.

2.3 Imbalanced Classification

Recently many real-world machine learning and data mining problems are characterized by imbalanced learning data, where at least one class is under-represented relative to others. Examples include fraud detection [19], medical diagnosis [37], bioinformatics [61], text categorization [71, 72] and etc, but are not limited to these. The problem of imbalanced data is often associated with asymmetric costs of misclassifying samples of different categories. Additionally the distribution of the test data may differ from that of the learning sample and the true misclassification costs may be unknown at learning time. Although much awareness of the issues related to data imbalance has been raised, many of the key problems still remain open. In this thesis, we concentrate on the two-category case.

The problem of learning from imbalanced datasets occurs when the number of samples in one class is significantly greater than that of the others. Breiman et al. [5] discussed the relationship between the prior probability of a class and its error cost. Categories with fewer examples in the training set have a lower prior probability and a lower error cost. This is problematic when true error cost of the minority class is higher than is implied by the distribution of examples in the training set.

When learning methods are conducted to skewed datasets, some algorithms will find an acceptable trade-off between the true-positive and false-positive rates. However, others learn simply to predict the majority classes. Indeed, classifiers that always predict the majority class can obtain higher classification accuracies than those that predict both classes equally well. Several research efforts have been proposed for coping with skewed data sets. For instance, in order to overcome the difficulty of imbalanced datasets problems we could over-sample (i.e., duplicate) examples of the minority class, under-sample (i.e., remove)

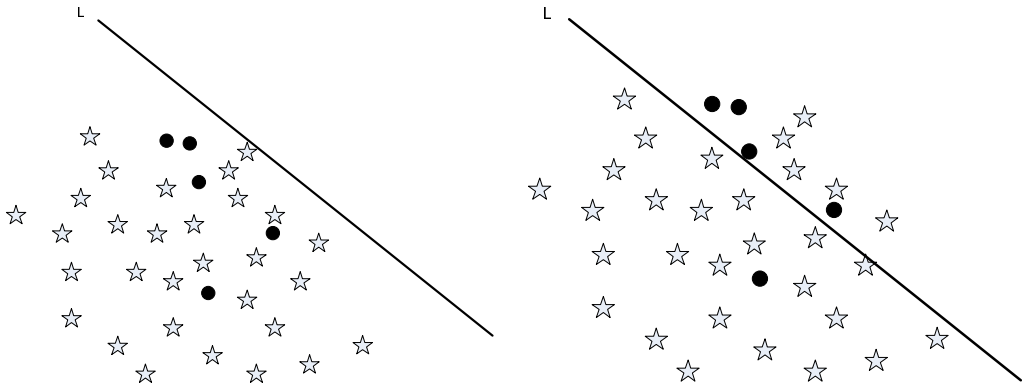


Figure 2.6: Imbalanced Classification Illustration. The overall classification accuracy is 86%, while the accuracy for the more important class is 0% (labeled as '•') (Left). The overall classification accuracy is 83%, while the accuracy for the more important class is 80% (labeled as '•') (Right)

examples of the majority class, or both. We can also learn to predict the minority class with the majority class as the default prediction. Solutions also exist to weight examples in an effort to bias the performance element toward the minority class and to weight the rules themselves. Schemes of boosting the examples of the minority class have also been proposed.

In [78], Provost gave an impressive summary about the related issues for imbalance dataset classification. He pointed out that it would be a critical mistake to use the classifiers produced by standard machine learning algorithms without adjusting the output threshold when studying problems with imbalanced data. He also pointed out that the normal classifier would cause problems for the imbalanced dataset by operating on data drawn from the same distribution as the training data. Weiss and Provost [116] conducted an empirical study for imbalanced dataset learning problems. From their research efforts, it is evident that the natural data distribution usually is not the best distribution for learning and a different class distribution should generally be chosen when the dataset size is limited.

A synthetic dataset is employed to take a systematic research

effort about the imbalance classification problem on the specific case [43, 44, 45]. It is evident that the performance of imbalance dataset problems is related with three factors: complexity of the problem, training set size and degree of the imbalance. They found that independently of the training size, linearly separable domains are not sensitive to imbalance. It also compared the effectiveness of several techniques for the imbalance problem: *over-sampling*, *under-sampling* and *cost modifying*. They concluded that the bad performance of the imbalance dataset problems is usually caused by small disjuncts that can not be classified accurately.

In conclusion, the methods aiming to tackle with the imbalance dataset problem can be summarized into three big categories [1]: *Algorithm specific approach* [80, 117], *Pre-processing for the data* (under-sample, over-sample, progressive, active, etc.) [1] and *Post-processing for the learned model* [18].

2.4 Active Learning

The fundamental objective of machine learning and data mining is to obtain the general patterns from a limited amount of data [108, 109]. For a long time, the majority of statistical machine learning scenarios commonly fall into one of two learning tasks: *supervised learning* or *unsupervised learning*. Under both learning frameworks, usually we first collect a significant quantity of data that is randomly sampled from the underlying data distribution, and we then induce a classifier or model. This methodology is referred as passive learning. A passive learner receives a random selected dataset from the world and then output a classifier or model [106, 110].

The supervised learning task is to predict some additional aspects of an input object [106]. The training data consist of pairs of input objects, and desired outputs. The output of the

Figure 2.7: A Representative Passive Learner.

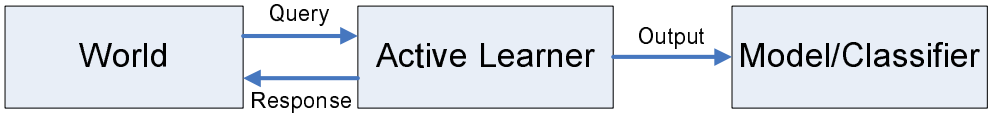


Figure 2.8: A Representative Active Learner.

function can be a continuous value (called regression), or can predict a class label of the input object (called classification). The objective of the supervised learner is to predict the value of the function for any valid input object after having seen a number of training examples (i.e., pairs of input and target output). To achieve this, the learner has to generalize from the presented data to unseen situations in a reasonable way. Examples of such a learning scenario include the simple problem of trying to predict an employee's salary given his/her occupation and the more complex task of trying to predict the subject of an image given the raw pixel values. Classification is the fundamental area of supervised learning tasks. The objective of classification is to create a mapping from input objects to labels. A typical example of a classification task is text categorization, where we want to automatically assign labels to a new text document based on their textual content. The statistical learning approach to tackle this problem is to collect a training set by manually labeling some number of documents. Then we employ a learner together with the labeled documents to generate a mapping from documents to labels. We call this mapping a classifier. And we can use the classifier to predict new and unseen documents.

The other major area of machine learning is unsupervised learning [106]. Compared with supervised learning, the essence of unsupervised learning is that we are not given any concrete

information as to how well we are performing. Unsupervised learning encompasses clustering (where we try to find groups of data instances that are similar to each other) and model building (where we try to build a model from our data). One major topic of model building in machine learning is parameter estimation which is central to statistics. Here, we have a statistical model of a domain which contains a number of parameters that need estimating. By collecting a number of data instances we can use a learner to estimate these parameters. Collecting experimental data is crucial for accomplishing this task. Often the most time-consuming and costly task in these applications is the gathering of data. We have very limited resources for collecting such data in many cases. Therefore, it is particularly valuable to chose methods in which we can make use of these resources as much as possible. We assume that we randomly gather data instances that are IID (i.e., independent and identically distributed) in all settings. However, in many situations we may have a way of guiding the sampling process. For example, in the document classification task it is often easy to gather a large pool of unlabeled documents. Now, instead of randomly picking documents to be manually labeled for our training set, we have the option of more carefully choosing (or querying) documents from the pool that are to be labeled by choosing candidates that fit certain profiles (e.g., "the most uncertain samples"). Furthermore, we need not declare our desired queries before hand. We can determine our next query based upon the answers to our previous queries instead. This process of guiding the sampling process by querying for certain types of instances based upon the data that we have seen so far is called active learning [106, 108].

In the subsequent sections we present two active learning strategies: *uncertainly-based sampling*, which selects the samples for which the relevance function is most uncertain about, and *error reduction*, which aims at minimizing the generaliza-

tion error of the classifier. We also present a strategy for batch selection.

2.4.1 Uncertainly-based Sampling

This strategy aims at selecting unlabeled instances that the learner is most uncertain about [22]. The methodology is to compute a probabilistic output for each instance, and select the unlabeled instances with the probabilities closest to 0.5 [55]. Similar strategies have been also proposed on SVM classifier with a theoretical justification [57, 68, 110].

In all cases, a function may be computed. This function can be a distribution, a utility function, or a fellowship to a class (e.g., distance to the hyperplane for SVM) [107]. Therefore a function $f_y : x \rightarrow [-1, 1]$ is trained with some adaptation, where the most uncertain documents have an output close to 0. The cost function to minimize is then $g(x) =\mid f(x) \mid$. The efficiency of a method depends on the accuracy of the function estimation close to 0 under such a strategy. This is the area where it is the most difficult to perform a good evaluation [22].

2.4.2 Error Reduction

Active learning strategies based on error reduction select instances that, once added to the training set, minimize the error of generalization [22, 81]. Let $P(c \mid x)$ the (unknown) probability of an instance x to be in class c, and $P(x)$ the (also unknown) distribution of the instances. With a training set $\mathcal{A}$ with pairs (x, c) sampled from $P(x)$, $P(c \mid x)$ provides the estimation $\hat{P}_{\mathcal{A}}(c \mid x)$ of $P(c \mid x)$. The expected error of generalization can be written as:

$$E_{\hat{P}_{\mathcal{A}}} = \int L(P(c \mid x), \hat{P}_{\mathcal{A}}(c \mid x))dP(x) \tag{2.15}$$

with L a loss function which evaluates the loss between the estimated distribution $\hat{P}_{\mathcal{A}}(c \mid x)$ and the true distribution $P(c \mid x)$.

The optimal pair $(x^\star, c^\star)$ is the one which minimizes this expectation:

$$\forall(x,c) \qquad E_{\hat{P}_{\mathcal{A}^\star}} < E_{\hat{P}_{\mathcal{A}+(x,c)}}, \tag{2.16}$$

with $\mathcal{A}^\star = \mathcal{A} + (x^\star, c^\star)$.

Roy and McCallum [81] propose to estimate the probability $P(c \mid x)$ with the function provided by the classifier, and estimate $P(x)$ over X. The estimation of the expectation becomes the following with a maximum loss function:

$$\hat{E}_{\hat{P}_{\mathcal{A}^\star}} = \frac{1}{\mid J \mid} \sum_{x \in J} (1 - \max_{c \in \{-1,1\}} \hat{P}_{\mathcal{A}^\star}(c \mid x)), \tag{2.17}$$

with J the set of unlabeled instances. But We don't know the label of each candidate. Roy and McCallum compute the expectation for each possible label, which finally gives the following cost function:

$$g(x) = \sum_{c \in \{-1,1\}} E_{\hat{P}_{\mathcal{A}+(x,c)}} \hat{P}_{\mathcal{A}}(c \mid x), \tag{2.18}$$

with $\hat{P}_{\mathcal{A}}(c \mid x)$ estimated with the function $f_y(x)$:

$$\hat{P}_{\mathcal{A}}(c \mid x) = \frac{c}{2}(f_y(x) + c), \tag{2.19}$$

with $f_y(x)$ such as y encodes the training set $\mathcal{A}$.

2.4.3 Batch Selection

In many real cases, it is often necessary to select batches of new training examples [22, 30, 31]. Many active learning strategies are made to select only one new training example. With no particular extension, these methods can select several instances very close in the feature space. In view of the power of current

classification techniques, labeling a batch of very close instances or only one of them always gives the same classification.

In [110], Tong and Koller propose to select batches yielding minimum worst-case version space volume. However, this method requires a lot of computations which make it infeasible in practice. Based on the diversity of angles between the hyperplanes in the version space, Brinker proposes a fast approximation of this strategy in [6]. The method selects instances close to the SVM boundary one far from another, and also far from the current training data.

2.5 Convex Optimization

2.5.1 Overview of Convex Optimization

Many machine learning and data mining problems can be formulated as constrained optimization problems which can sometimes be expressed in convex form with proper mathematical manipulations. These kinds of convex problems can be solved very efficiently in practice. In addition, interior-point methods are often employed to solve these problems to a specified accuracy within a polynomial operations of the problem dimensions. Interested readers may refer to [4] for more details about convex optimization theory.

First we would like to introduce some basic definitions of convex problems.

Definition 2.1 ***Convex Set:*** *A set S is convex if the line segment between any two points in S lies in S, i.e., if for any x_1, $x_2 \in S$ and any θ with $0 \leq \theta \leq 1$, we have*

$$\theta x_1 + (1 - \theta) x_2 \in S. \tag{2.20}$$

Definition 2.2 ***Convex Function:*** *A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if* ***dom****f is a convex set and if for all x_1, $x_2 \in$* ***dom****f,*

and any θ with $0 \leq \theta \leq 1$, the following inequality holds:

$$f(\theta x_1 + (1-\theta)x_2) \leq \theta f(x_1) + (1-\theta)f(x_2). \tag{2.21}$$

Definition 2.3 ***Convex Problem:*** *A convex optimization problem is defined as one being in the following form:*

$$\begin{aligned} \min_{x} \quad & f_0(x) \\ s.t. \quad & f_i(x) \leq 0 \qquad 1 \leq i \leq m, \\ & h_i(x) = 0 \qquad 1 \leq i \leq k, \end{aligned} \tag{2.22}$$

where $x \in \mathbb{R}^n$ is the optimization variable, $f_0, \ldots, f_m$ are convex functions, and $h_0, \ldots, h_k$ are affine functions.

In the above definitions, the function f_0 is usually called the objective function or cost function. The inequalities are called inequality constraints and the equations are called equality constraints. If there is no constraint, the problem is an unconstrained problem. The subsequent parts review several types of convex optimization problems.

2.5.2 Linear Program

Definition 2.4 ***Linear Program:*** *A convex optimization problem is called a Linear Program (LP) when the objective and constraint functions are all affine. The Linear Program problem has the following general form:*

$$\begin{aligned} \min_{x} \quad & c^T x + d \\ s.t. \quad & G\,x \preceq h, \\ & A\,x = b, \end{aligned} \tag{2.23}$$

where $G \in \mathbb{R}^{m \times n}$ and $A \in \mathbb{R}^{p \times n}$.

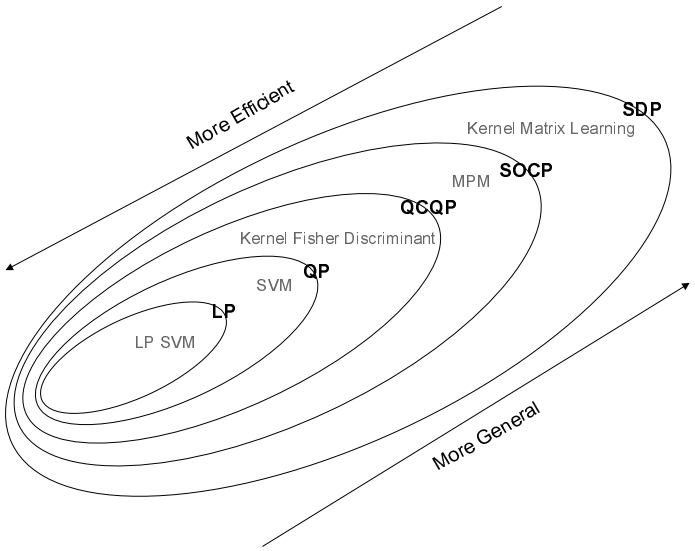


Figure 2.9: Convex Optimization Problems Relational Graph. The Key: Convex Optimization can be solved effectively in Polynomial time.

2.5.3 Quadratic Program

Definition 2.5 ***Quadratic Program:*** *A convex optimization problem is called a Quadratic Program (QP) if the objective function is (convex) quadratic, and the constraint functions are affine. The Quadratic Program problem is generally expressed as:*

$$\begin{aligned} \min_{x} \quad & \tfrac{1}{2}x^T P x + q^T x + r \\ s.t. \quad & G\,x \preceq h, \\ & A\,x = b, \end{aligned} \tag{2.24}$$

where $P \in \mathbb{S}^n_+$, $G \in \mathbb{R}^{m\times n}$ *and* $A \in \mathbb{R}^{p\times n}$.

From the above definitions, one can see that quadratic programs include linear programs as a special case by taking $P = 0$.

2.5.4 Quadratically Constrained Quadratic Program

Definition 2.6 ***Quadratically Constrained Quadratic Program:*** *A convex optimization problem is called a Quadratically*

Constrained Quadratic Program (QCQP) if the objective function and the constraint functions are all (convex) quadratic. The Quadratically Constrained Quadratic Program Problem is generally expressed as:

$$\begin{array}{ll} \min\limits_{x} & \frac{1}{2}x^T P_0 x + q_0^T x + r_0 \\ s.t. & \frac{1}{2}\, x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m \\ & A\, x = b, \end{array} \tag{2.25}$$

where $P_i \in \mathbb{S}_+^n$, $i = 0, \ldots, m$.

It is evident that quadratically constrained quadratic programs include quadratic programs and linear programs as special cases.

2.5.5 Cone Program

In addition to convex optimization problems of standard forms, another type of very useful generalizations are convex optimization problems with generalized inequality constraints. One of the simplest case is the Cone Program (CP), which is defined as follows:

Definition 2.7 ***Cone Program:*** *A convex optimization problem with generalized inequalities is called a Cone Program (CP) if the objective function is linear and the inequality constraint functions are affine:*

$$\begin{array}{ll} \min\limits_{x} & c^T x \\ s.t. & F\, x + g \preceq_K 0 \\ & A\, x = b, \end{array} \tag{2.26}$$

where $K \subseteq \mathbb{R}^k$ *is a proper cone.*

2.5.6 Semi-definite Program

Definition 2.8 ***Semi-definite Program:*** *A convex optimization problem with generalized inequalities is called a Semi-definite Program (SDP) if the objective function is linear and the inequality constraint functions are affine with the cone of positive semi-definite $k \times k$ matrices, i.e., K is $\mathbb{S}_+^k$. The Semi-definite Program has the standard form as:*

$$\begin{array}{ll} \min\limits_{x} & c^T x \\ s.t. & x_1 F_1 + \ldots + x_n F_n + G \preceq_K 0 \\ & A\, x = b, \end{array} \tag{2.27}$$

where $G, F_1, \ldots, F_n \in \mathbb{S}^k$ and $A \in \mathbb{R}^{p \times n}$.

Similarly, SDP problems include LP, QP and QCQP as special cases. As a comparison of computational complexity, all of them can be solved efficiently in polynomial time. Among these four types of problems, in general, SDP is the hardest problem, QCQP is easier than SDP, QP is easier than QCQP, and LP is the easiest one.

□ **End of chapter.**

Chapter 3

Imbalanced Learning with BMPM for CBIR

In recent years, Minimax Probability Machines (MPMs) have demonstrated excellent performance in a variety of pattern recognition problems. At the same time various machine learning methods have been applied on relevance feedback tasks in Content-based Image Retrieval (CBIR). One of the problems in typical techniques for relevance feedback is that they treat the relevant feedback and irrelevant feedback equally. Since the negative instances largely outnumber the positive instances, the assumption that they are balanced is incorrect as the data is biased. In this chapter we study how Biased Minimax Probability Machine (BMPM), a variation of MPM, can be applied for relevance feedback in image retrieval tasks. Different from previous methods, this model directly controls the accuracy of classification of the future data to construct biased classifiers. Hence, it provides a rigorous treatment on imbalanced dataset. Mathematical formulation and explanations are provided to demonstrate the advantages. Experiments are conducted to evaluate the performance of our proposed framework, in which encouraging and promising experimental results are obtained.

3.1 Research Motivation

With the recent progress of hardware for capturing and storing image data, CBIR has attracted a lot of research interests in the past decades [84]. However two semantically similar images may be located far from each other in the feature space, while two absolutely different images may lie close to each other [83]. This is known as the problem of *semantic gap* between low-level features and high-level concepts [90]. Relevance feedback has been shown to be a powerful tool to address this problem and improve retrieval performance in CBIR [84]. Recently, researchers proposed a number of classification techniques to attack relevance feedback tasks including some state-of-the-art models such as Support Vector Machines (SVMs) [27, 29, 32]. However most of the classification techniques treat the relevance feedback problem as a strict binary classification problem and they do not consider the imbalanced dataset problem, which means the number of irrelevant images are significantly larger than the number of relevant images. This imbalanced dataset problem would lead the positive data (relevant images) to be overwhelmed by the negative data (irrelevant images). An illustration has been shown in Fig. 2.6.

MPM has been used as a novel and important tool to perform classification tasks [53]. Compared with traditional classification models, it has a promising accuracy performance on pattern recognition tasks. In order to tackle the problem of imbalanced dataset in CBIR, we propose to use a modified Minimax Probability Machine, called Biased Minimax Probability Machine which can better model the relevance feedback problem and reduce the accuracy degradation caused by the imbalanced dataset problem [34, 36, 37].

The rest of this chapter is organized as follows. Section 3.2 reviews some related research efforts on relevance feedback and

MPM. Section 3.3 formulates the relevance feedback technique employing BMPM and shows the benefits compared with the conventional techniques. We present the experimental results and performance evaluation in Section 3.4. Finally, Section 3.5 concludes the work of this chapter.

3.2 Background Review

Here we will give a brief introduction on the related work of relevance feedback in CBIR, the theory of MPM and its variations, Minimum Error Minimax Probability Machine (MEMPM) and Biased Minimax Probability Machine (BMPM).

3.2.1 Relevance Feedback for CBIR.

Relevance feedback techniques have been used as a powerful tool for Content-based Image Retrieval [84, 120]. There are various of methodologies involving in that research area such as Self-Organizing Map [52], Decision Tree [60], Artificial Neural Network [21], and Bayesian Learning Network [100], etc. Moreover many popular classification techniques have been employed to tackle the relevance feedback problem, such as Bayesian classifiers and SVM [27, 32], etc. Among them, SVM-based techniques are the most promising and effective techniques to solving the relevance feedback task.

3.2.2 Minimax Probability Machine

We here introduce the basic concept of MPM [53]. In pattern classification problems, MPM provides very good empirical generalization performance.

Let us illustrate MPM in a binary classification case. Suppose two random n-Dimensional vectors, $\mathbf{x}$ and $\mathbf{y}$, represent two classes of data, where $\mathbf{x}$ belongs to the family of distributions

with a given mean $\overline{\mathbf{x}}$ and a covariance matrix $\Sigma_{\mathbf{x}}$, denoted as $\mathbf{x} \sim (\overline{\mathbf{x}}, \Sigma_{\mathbf{x}})$; similarly, $\mathbf{y}$ belongs to the family of distributions with a given mean $\overline{\mathbf{y}}$ and a covariance matrix $\Sigma_{\mathbf{y}}$, denoted as $\mathbf{y} \sim (\overline{\mathbf{y}}, \Sigma_{\mathbf{y}})$. Here $\mathbf{x}$, $\mathbf{y}$, $\overline{\mathbf{x}}$, $\overline{\mathbf{y}} \in \mathbb{R}^n$, and $\Sigma_{\mathbf{x}}$, $\Sigma_{\mathbf{y}} \in \mathbb{R}^{n \times n}$. In the following discussion of this thesis, $\mathbf{x}$ represents the relevance image class and $\mathbf{y}$ represents the irrelevance image class.

The MPM attempts to determine the hyperplane $\mathbf{a}^T\mathbf{z} = b$ ($\mathbf{a} \in \mathbb{R}^n$, $\mathbf{z} \in \mathbb{R}^n$, $b \in \mathbb{R}$) which can separate two classes of data with maximal probability. The formulation for the MPM model [54] is written as follows:

$$\begin{aligned} \max_{\alpha, \mathbf{a} \neq \mathbf{0}, b} \quad & \alpha \quad s.t. \\ & \inf_{\mathbf{x} \sim (\bar{\mathbf{x}}, \Sigma_{\mathbf{x}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{x} \geq a\} \geq \alpha, \\ & \inf_{\mathbf{y} \sim (\bar{\mathbf{y}}, \Sigma_{\mathbf{y}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{y} \leq b\} \geq \alpha, \end{aligned} \tag{3.1}$$

where α represents the lower bound of the accuracy for future data. Future points $\mathbf{z}$ for which $\mathbf{a}^T\mathbf{z} \geq \alpha$, are then classified as the class $\mathbf{x}$; otherwise they are judged as the class $\mathbf{y}$.

Later, Huang *et al.* [36] improved the model by removing away the assumption that these two classes have the same importance, and furthermore adding a bias to the more important class. As we could observe from the above formulation, this model actually assumes that two classes have the same importance. Hence it makes the worst-case accuracies for two classes the same. However, in real applications, especially in relevance feedback of Content-based Image Retrieval, two classes of data are usually biased, i.e., the relevant class is often more important than the irrelevance class and the quantities of both classes are imbalanced. Therefore it is more appropriate to take the inherited bias into account in this context. In the following section, we will introduce Huang's developments, two extensions of MPM, i.e., MEMPM and BMPM.

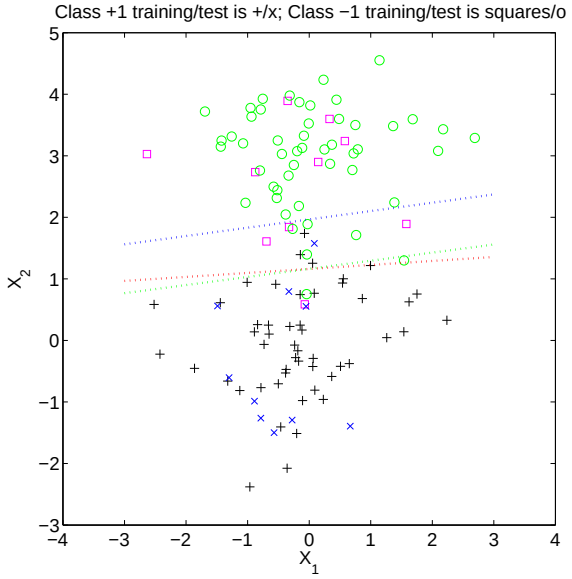


Figure 3.1: Decision lines comparison: MPM decision line (dotted red line), BMPM decision line (dotted green line), SVM decision line (dotted blue line).

3.2.3 Extensions of Minimax Probability Machine

With exactly the same scenario as MPM, the mathematical model of MEMPM is as following:

$$\begin{aligned} \max_{\alpha,\beta,b,\mathbf{a}\neq\mathbf{0}} \quad & \theta\alpha + (1-\theta)\beta \quad s.t. \\ & \inf_{\mathbf{x}\sim(\bar{\mathbf{x}},\Sigma_{\mathbf{x}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{x} \geq b\} \geq \alpha, \\ & \inf_{\mathbf{y}\sim(\bar{\mathbf{y}},\Sigma_{\mathbf{y}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{y} \leq b\} \geq \beta, \end{aligned} \tag{3.2}$$

while BMPM is defined as:

$$\begin{aligned} \max_{\alpha,\beta,b,\mathbf{a}\neq\mathbf{0}} \quad & \alpha \quad s.t. \\ & \inf_{\mathbf{x}\sim(\bar{\mathbf{x}},\Sigma_{\mathbf{x}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{x} \geq b\} \geq \alpha, \\ & \inf_{\mathbf{y}\sim(\bar{\mathbf{y}},\Sigma_{\mathbf{y}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{y} \leq b\} \geq \beta, \\ & \beta \geq \beta_0. \end{aligned} \tag{3.3}$$

3.3 Relevance Feedback using BMPM

In this section, we will first give a more detailed introduction on BMPM. Next, we show the benefits of applying BMPM in relevance feedback in Content-based Image Retrieval. We then present how the BMPM based approach can be employed for relevance feedback tasks in Section 3.3.3.

3.3.1 Model Definition

Given reliable $\{\overline{\mathbf{x}}, \Sigma_{\mathbf{x}}\}$, $\{\overline{\mathbf{y}}, \Sigma_{\mathbf{y}}\}$ for two classes of data, we try to find a hyperplane $\mathbf{a}^T\mathbf{z} = b(\mathbf{a} \neq 0, \mathbf{z} \in \mathbb{R}^n, b \in \mathbb{R})$ 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.[1] We formulate this objective as follows:

$$\begin{aligned} \max_{\alpha,\beta,b,\mathbf{a}\neq\mathbf{0}} \quad & \alpha \quad s.t. \\ & \inf_{\mathbf{x}\sim(\bar{\mathbf{x}},\Sigma_{\mathbf{x}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{x} \geq b\} \geq \alpha, \\ & \inf_{\mathbf{y}\sim(\bar{\mathbf{y}},\Sigma_{\mathbf{y}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{y} \leq b\} \geq \beta, \\ & \beta \geq \beta_0, \end{aligned} \tag{3.4}$$

where α represents the lower bound of the accuracy for the classification, or the worst-case accuracy of future data points $\mathbf{x}$; the similar for β. The parameter β_0 is a pre-specified positive constant, which represents an acceptable accuracy level for the less important class $\mathbf{y}$.

The above formulation is derived from MPM, which requires the probabilities of correct classification for both classes to be an equal value α. Through this formulation, the BMPM model can handle the imbalanced classification in a direct way by changing the value of α and $\beta_{\mathbf{0}}$. This model provides a different treatment

[1] The readers may refer to [39] for a more detailed and complete description.

on different classes, i.e., 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 less important class $\mathbf{y}$. Furthermore, the derived decision hyperplane is directly associated with two real accuracy indicators of classification of the future data, i.e., α and β_0, for each class.

3.3.2 Advantages of BMPM in Relevance Feedback

From the above formulations, one could see that the optimization in BMPM is similar to the one in the MPM, which is in convex optimization format and could be efficiently solved in polynomial time. Now, we show the mathematical differences and the advantages of our proposed BMPM framework from an analytical perspective for solving the relevance feedback problem compared with SVMs and other conventional learning methods.

Obviously we see that BMPM is with the following constraints, in contrast to the one of MPM in Eq. (3.1) and the one of MEMPM in Eq. (3.2)

$$\begin{aligned} \inf_{\mathbf{x}\sim(\bar{\mathbf{x}},\Sigma_{\mathbf{x}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{x} \geq b\} &\geq \alpha, \\ \inf_{\mathbf{y}\sim(\bar{\mathbf{y}},\Sigma_{\mathbf{y}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{y} \leq b\} &\geq \beta, \\ \beta &\geq \beta_0. \end{aligned} \tag{3.5}$$

The difference indicates that the proposed BMPM framework tries to improve the accuracy of relevant images while maintaining an acceptable specificity of irrelevant ones. This methodology provides a rigorous way to handle the relevance feedback problem by directly control the classification accuracy and this is powerful for solving the imbalanced dataset problem in CBIR. However, SVMs and other traditional learning models treat the two classes equally without any bias or direct control, which is not effective to model and solve the relevance feedback problem.

3.3.3 Relevance Feedback Framework by BMPM

In this section we describe how to formulate the relevance feedback algorithm by employing the BMPM technique. Applying BMPM based technique in relevance feedback is similar to the conventional classification tasks. However, the relevance feedback needs to construct an iterative function to produce the retrieval results. The following strategy is our proposed approach for retrieval task in CBIR.

Algorithm *BMPM-based Relevance Feedback*
Input: $\mathbf{Q}_{im}$ (query image)
Output: $\mathbf{R}_{im}$ (images belong to the relevant class with similar semantic content)
1. $\mathbf{F}_q \Leftarrow \mathbf{Q}_{im}$ /* Feature extraction for query image */
2. $\mathbf{F}_q \Leftarrow \mathbf{x}/\mathbf{y}$ /* Assign label to query image */
3. **For** i = 1: MaxIt
4. $\quad \mathbf{R}_{im} \Leftarrow \mathbf{R}^i_{im}$ /* Update based on similarity measurement */
5. $\quad$ Involve feedback information using BMPM
6. $\quad\quad \mathbf{R}^1_{im}, \mathbf{R}^2_{im} \Leftarrow \mathbf{R}_{im}$ /* Separate returned images into two sets */
7. $\quad\quad \mathbf{R}^1_{im}, \mathbf{R}^2_{im} \Leftarrow \{\mathbf{x}, \mathbf{y}\}$ /* Assign labels to classes by experts */
8. $\quad\quad$ Classification task by BMPM
9. $\quad i \Leftarrow i + 1$
10. **End For**
11. **Return** $\mathbf{R}_{im}$

Figure 3.2: BMPM-based Relevance Feedback

After a certain number of iterations of relevance feedback, our proposed strategy returns the *Top-n* most relevant images and also learns a reasonable classifier to classify the imbalanced image dataset.

3.4 Experimental Results

We implement BMPM-based learning scheme and apply to relevance feedback in Content-based Image Retrieval. In this section, we describe the iterative framework and show the experimental results. We compare the performance of our proposed

Dataset	#Instances	#Features	#Classes
Synthetic Dataset	1000	2	10

Table 3.1: Overview of synthetic dataset.

Classes	*Mean*	*Covariance*
Class 1	[0,-3]	[1, 0;0, 1.5]
Class 2	[0,-4]	[1, 0;0, 1.5]
Class 3	[0,-5]	[1, 0;0, 1.5]
Class 4	[1,-1]	[1, 0;0, 1.5]
Class 5	[-1,1]	[1, 0;0, 1.5]
Class 6	[-2,0]	[1, 0;0, 1.5]
Class 7	[2,0]	[1, 0;0, 1.5]
Class 8	[1,0]	[1, 0;0, 1.5]
Class 9	[0,1]	[1, 0;0, 1.5]
Class 10 (Relevant Class)	[0,5]	[1.5, 0;0, 1.5]

Table 3.2: Detailed information of synthetic dataset

approach with two classification models for relevance feedback: MPM and SVM. The SVM algorithm deployed in our experiments is based on modifying the codes in the *libsvm* library, and MPM and BMPM schemes are adopted from the MPM and BMPM packages respectively. Furthermore we choose the same kernel and parameters (such as α, β_0, etc.) for all the settings. The experiments are evaluated on both a synthetic dataset and a real-world image dataset. All our works are done on a 3.2GHz machine with Intel Pentium 4 processor and 1Gb RAM.

3.4.1 Experiment Datasets

Synthetic Dataset

We generate a synthetic dataset to simulate the real-world images. The dataset consists 10 categories, 9 of which contains

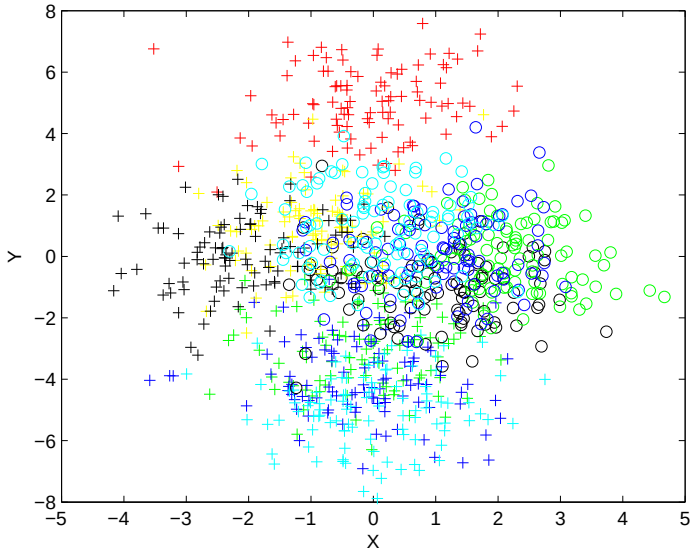


Figure 3.3: Visualization of synthetic dataset. Relevant class: red plus; Irrelevant class: others. In the experiments, we use one versus all strategy for the multi-class classification problem.

100 data points randomly generated by Gaussian Distribution with different means and covariance matrices in a 2-dimensional space. The remaining class contains 100 instances generated by another mean and covariance matrix, representing the relevant samples.

Real-world Dataset

The real-world dataset is chosen from the COREL image CDs. We organize one dataset which contains various images with different semantic meanings, such as *bird*, *pyramid*, *model*, *autumn*, *dog*, and *glacier*, etc. It is with 6 categories (we name it 6-Bird). The class of *bird* which we recognize as the positive class contains 50 images. The other 5 categories with each including 100 instances are regarded as the negative class.

Here we extract three different features to represent the images: *color*, *shape*, and *texture*. The color feature employed is the

color histograms. We quantize the number of pixels into 10 bins for each color channel (Hue, Saturation, and Value) respectively. Then we obtain a 30-dimensional color histogram. We use edge direction histogram as shape feature to represent the images [42]. First we compute the edge images by the Canny edge detector and obtain the edge direction histogram by quantizing the results into 15 bins of 24 degrees. Therefore a 15-dimensional edge direction histogram is used as the edge feature. We apply the wavelet-based texture in our experiments [91]. Gabor Wavelet Decomposition [20] is first performed and we compute the features for each Gabor filter output afterwards. Following this approach we obtain a 16-dimensional vector to represent the texture information for each image.

3.4.2 Performance Evaluation

Results on Synthetic Dataset

In our experiments, a category is first picked from the dataset randomly, and this category is assumed to be the user's desired query target. The frameworks then improve retrieval results by user's feedback. During each iteration of the relevance feedback procedure, 10 instances are picked from the dataset and labelled as either relevant or irrelevant samples based on the ground truth of the dataset. For the first iteration, three positive and seven negative samples are randomly picked out and the three learning schemes are applied with this initial set. For the iterations afterward, each model selects 10 samples and the number of the samples in the positive and negative regions are recorded. The precision of each model is then computed. Figure 3.4 shows the evaluation results of top-50 returned results. We can observe that BMPM outperforms the other models. The SVM based approach achieves better performance than the one of MPM in the application of relevance feedback in CBIR.

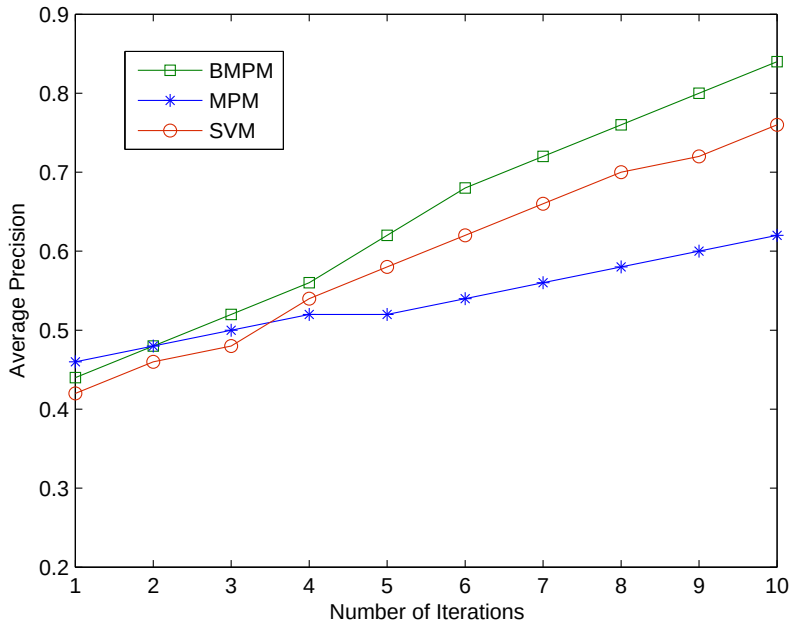


Figure 3.4: The experimental results for three models on the synthetic dataset: Top-50 returned samples are evaluated.

Results on Real-world Dataset

In the following, we present the experimental results by the algorithms on real-world images. The metric of evaluation is the *Average Precision* which is defined as the average ratio of the number of relevant images in the returned images over the total number of the returned images.

In the real-world dataset experiments, the iteration is similar as the one of synthetic dataset, except that we need to first perform feature extraction for the query images and image database. In each iteration of the feedback process, 10 images are picked from the database and labelled as either relevant or irrelevant based on the ground truth of the database. The precision is then recorded, and the whole process is repeated for 10 times to produce the average precision in each iteration for the proposed method.

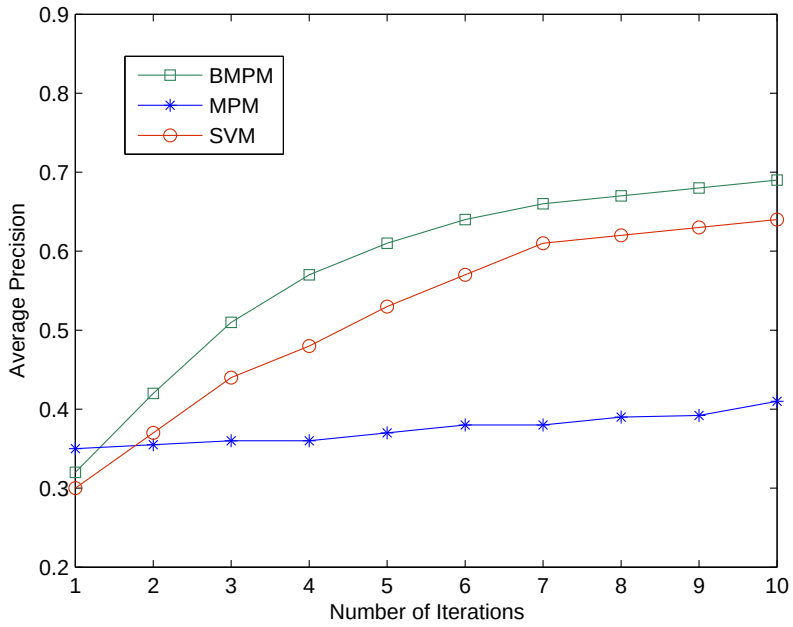


Figure 3.5: The experimental results for three models on the 6-Bird dataset: Top-50 returned images are evaluated.

Figure 3.5 shows the evaluation results on the 6-Bird dataset. From the results on the real-world dataset, we can observe that our proposed BMPM-based methodology outperforms other approaches such as MPM and SVM. We also notice that the performance of SVM is very competitive to BMPM, and MPM achieves the worst performance in these three models. The reason is that MPM cannot model the relevance feedback problem as good as SVM and BMPM due to its assumption that both positive and negative feedbacks are equal. From here we can see how the bias works. In order to know the detailed comparison of the three methods after a set number of iterations, we list the retrieval results in Tables 3.3 and 3.4. From the results, we can also see the similar results which verify our hypothesis.

Learning	Number of Iterations				
Models	0	1	3	7	10
BMPM	5	12	16	**26** ↑	**33** ↑
MPM	5	13	16	18	21
SVM	5	11	14	23	29

Table 3.3: Number of relevant images in Top-50 returned images.

Learning Models	Top50@6-Bird	Top30@6-Bird	Top20@6-Bird
BMPM	**0.68** ↑	**0.71** ↑	**0.75** ↑
MPM	0.42	0.47	0.55
SVM	0.63	0.66	0.70

Table 3.4: Average Precision after 10 Iterations.

3.4.3 Discussions

We have observed that the proposed BMPM-based scheme performs better than the conventional approaches from the experimental results. The traditional classification approaches, such as regular SVM and MPM, without considering the bias in the retrieval tasks is not appropriate in solving the relevance feedback problem. Furthermore, we know there are other methods to address the imbalanced dataset problem in literature [18] [118]. We can also consider to include them in our scheme in the future. Nevertheless, we have observed the promising results in demonstrating the effectiveness of our proposed BMPM technique for the relevance feedback problem in image retrieval.

3.5 Summary

In this chapter, we address the problem of imbalanced classification needed with the relevance feedback in CBIR and present a novel learning framework, the BMPM-based approach, to treat

this problem more precisely. In contrast to the traditional methods, the BMPM does not adopt an indirect approach, but directly controls the worst-case classification accuracy in order to impose a certain bias in favor of the relevant images. This provides a more effective way to handle imbalanced classification problems. We evaluate the performance of the BMPM-based relevance feedback on the synthetic dataset and the COREL Image Dataset. The results on both datasets show that the BMPM outperforms the other learning models on the problem of relevance feedback.

□ **End of chapter.**

Chapter 4

BMPM Active Learning for CBIR

In this chapter we apply Biased Minimax Probability Machine (BMPM) Active Learning to address the problem of relevance feedback in Content-based Image Retrieval (CBIR). In our proposed methodology we treat relevance feedback tasks in CBIR as an imbalanced learning task which is more reasonable than traditional methods since the negative instances largely outnumber the positive instances. Furthermore we incorporate active learning in order to improve the framework performance, i.e., try to reduce the number of iterations used to achieve the optimal boundary between relevant and irrelevant images. Different from previous work, this model builds up a biased classifier and achieves the optimal boundary using fewer iterations. Experiments are performed to evaluate our method with promising experimental results.

4.1 Problem Statement and Motivation

CBIR has attracted a lot of research interests in the past decades [90]. For CBIR, i.e., searching in image database based on their content, the focus was on Query By Example (QBE). A representative CBIR system contains four major parts: (1) image rep-

resentation, (2) high-dimensional image indexing, (3) similarity measurement between images, and (4) system design [124]. At the early stage of CBIR research, researchers mainly focused on the feature extraction for the best representation of the content of images. However these features are often low-level features. Therefore two semantically similar objects may locate far from each other in the feature space, while two absolutely different images may lie close to each other [124]. This is known as the problem of *semantic gap* between low-level features and high-level concepts and the subjectivity of human perception [27]. Although many features have been investigated for some CBIR systems, and some of them have demonstrated good performance, the problem has been the major encumbrance to more successful CBIR systems.

Relevance feedback has been shown to be a powerful tool to address the problem of the semantic gap and the subjectivity of human perception in CBIR [27]. Widely used in text retrieval, relevance feedback was first introduced by Rui *et al.* [84] as an iterative tool in CBIR. Since then it has become a major research topic in this area. Recently, researchers proposed a number of classification techniques to attack relevance feedback tasks, among which Support Vector Machine (SVM) based techniques are considered as the most promising and effective ones [27]. The major SVM techniques treat the relevance feedback problem as a strict binary classification problem. However, these methods do not consider the imbalanced dataset problem, which means the number of irrelevant images are significantly larger than the relevant ones. This imbalanced dataset problem would lead the positive data (relevant images) be overwhelmed by the negative data (irrelevant images). Furthermore, how to reduce the number of iterations in order to achieve the optimal boundary in this learning task is also a critical problem for image retrieval from large datasets.

In this chapter, we propose a relevance feedback technique to incorporate both Biased Minimax Probability Machine and active learning to attack these two problems, which can better model the relevance feedback problem and reduce the number of iterations in the learning interaction.

4.2 Background Review

In text retrieval, relevance feedback was used early on and had been proven to improve results significantly. The adoption of relevance feedback in CBIR is more recent, and it has evolved to incorporate various machine learning techniques into applications recently. In [60], Decision Tree was employed to model the relevance feedback task. In [13], Bayesian learning was conducted to attack the problem of relevance feedback. Apart from these, many other conventional machine learning methods were also proposed, e.g., Self-Organizing Map [52], Artificial Neural Network [90], etc. Furthermore, many state-of-the-art classification algorithms were suggested to model and solve the relevance feedback problem, e.g., Nearest Neighborhood classifier [105] and SVM [107], etc. Among these techniques, SVM-based techniques are the most effective ones to address the relevance feedback task in CBIR.

However, conventional relevance feedback techniques by SVMs or other learning models are based on strict binary classification tasks. In other words, they do not consider the imbalanced dataset problem in relevance feedback. Moreover, these techniques always consume a number of iterations to obtain an optimal boundary which is not suitable for searching images from large datasets. In order to address this imbalance classification task and make relevance feedback more efficient, we propose the Biased Minimax Probability Machine Active Learning to construct the relevance feedback technique in CBIR.

4.3 Relevance Feedback by BMPM Active Learning

In this section, we introduce the concepts of active learning and BMPM. We then present and formulate our proposed BMPM methodology with active learning, applying to relevance feedback.

4.3.1 Active Learning Concept

In supervised learning, often the most time-consuming and costly process in designing classifiers is instance labelling when we face large scale learning tasks. Instead of randomly picking objects to be manually labelled for training, active learning is a novel mechanism for selecting unlabelled objects based on the result of past labelled objects. Under this framework, the learner could construct a classifier with much fewer manually labelled samples (i.e., optimal data).

Based on the different criteria for optimal data, there are three main types of active learning method: *Most Informative*, *Minimizing the Expected Error* and *Farthest First*. In each iteration of learning, the samples with highest classification uncertainty is chosen for manual labelling [124]. Then the classification model is retrained with additional labelled samples. The key challenge in active learning for relevance feedback is how to measure the information associated with an unlabelled images. In [56], various of distinct classifier models were first generated. Then, the classification uncertainty of a test image is measured by the amount of disagreement among the test images. Another batch of methodologies measure the information associated with a test sample by how far the sample is away from the classification boundary. One of the most promising approaches within this group is the SVM active learning developed by Tong and

Chang [107].

4.3.2 General Approaches for Active Learning

Generally speaking the most important step in active learning is to define a notion of a model M and the model loss $Loss(M)$. The definition of a model and the associated model loss can be tailored to match the particular task at hand. Under this framework, the next query that will result in the future model with the lowest model loss is chosen. It is straightforward to extend this framework to batch mode active learning. However, in many situations this type of active learning is computationally infeasible. Thus we shall just consider the simplified schema.

Algorithm *General Schema for Active Learning*
Input: *M, MaxIt, pQueries*
Output: *M*
1. **For** *i=1:MaxIt* /* we do MaxIt iterations */
2. **For** *q* in *pQueries* /* for each query in the potential queries */
3. Evaluate *Loss(q)*
4. **End For**
5. Select *q* whose *Loss(q)* is lowest
6. Update *M* with *q* and *x*; /* update with the query and its response */
7. *i++*;
8. **End For**
9. **Return** *M*

Figure 4.1: General Schema for Active Learning.

When we are asking a potential query, $\mathbf{q}$, we need to assess the loss of the subsequent model, M'. The posterior model M' is the original model M updated with query $\mathbf{q}$ and response $\mathbf{x}$. Since we do not know what the true response $\mathbf{x}$ to the potential query will be, we propose to maintain a distribution over the possible responses to each query. After asking a query where we take the expectation over the possible responses to the query we can then compute the expected model loss:

$$Loss(\mathbf{q}) = E_{\mathbf{x}} Loss(M') \tag{4.1}$$

If we use this definition in the active learning algorithm we would then choose the query that results in the minimum expected model loss.

In general, the common approach for active learning is as follows. We first choose a model and model loss function appropriate for the learning task. Given a potential query we also choose a method for calculating the potential model loss. For each potential query we then evaluate the potential loss incurred and we then chose to ask the query which gives the lowest potential model loss. This general schema is outlined in Fig. 4.1.

4.3.3 Biased Minimax Probability Machine

We assume two random vectors $\mathbf{x}$ and $\mathbf{y}$ represent two classes of data with means 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 $\Sigma_{\mathbf{x}}, \Sigma_{\mathbf{y}} \in \mathbb{R}^{n\times n}$. We also use $\mathbf{x}$ and $\mathbf{y}$ to represent the corresponding class of the $\mathbf{x}$ data and the $\mathbf{y}$ data respectively.

With given reliable $\{\overline{\mathbf{x}}, \Sigma_{\mathbf{x}}\}$, $\{\overline{\mathbf{y}}, \Sigma_{\mathbf{y}}\}$ for two classes of data, we try to find a hyperplane $\mathbf{a}^T\mathbf{z} = b$ ($\mathbf{a} \neq 0, \mathbf{z} \in \mathbb{R}^n, b \in \mathbb{R}$, here the superscript T denotes the transpose) 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. The problem is formulated as:

$$\begin{aligned} \max_{\alpha,\beta,b,\mathbf{a}\neq\mathbf{0}} \quad & \alpha \\ s.t. \quad & \inf_{\mathbf{x}\sim(\bar{\mathbf{x}},\Sigma_{\mathbf{x}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{x} \geq b\} \geq \alpha, \\ & \inf_{\mathbf{y}\sim(\bar{\mathbf{y}},\Sigma_{\mathbf{y}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{y} \leq b\} \geq \beta, \\ & \beta \geq \beta_0, \end{aligned} \tag{4.2}$$

where α represents the lower bound of the accuracy for the clas-

sification, or the worst-case accuracy of future data points $\mathbf{x}$; likewise β. The parameter β_0 is a pre-specified positive constant, which represents an acceptable accuracy level for the less important class $\mathbf{y}$.

Through this formulation, the BMPM model can handle the biased classification in a direct way. This model provides a different treatment on different classes, i.e., the hyperplane $\mathbf{a}_*^T\mathbf{z} = b_*$ given by the solution of this optimization problem will favor the classification of the important class $\mathbf{x}$ over the less important class $\mathbf{y}$.

4.3.4 Proposed Framework

Here we describe how to formulate the relevance feedback algorithm by employing the BMPM technique and Active Learning. Applying BMPM-based techniques in relevance feedback is similar to the traditional classification task. The challenge lies in how to adopt an appropriate active learning strategy in our framework.

Let I_i, $i = 1, 2, ..., N$ be the images in the database, and $\mathbf{x}, \mathbf{y}$ be the relevant images and irrelevant ones respectively. For each image I_i, we define P_i to be the probability that this image belongs to a particular class $\mathbf{x}$ or $\mathbf{y}$. More specifically we define P_{ix}=1 if the image I_i has been labelled to class $\mathbf{x}$, and P_{ix}=0 if it has been classified to class $\mathbf{y}$. P_{iy} is defined similarly. If the image has not been labelled, P_i is estimated by its k-Nearest Neighborhood as:

$$P_{ix} = \frac{n_x}{k} \tag{4.3}$$

$$p_{iy} = \frac{n_y}{k} \tag{4.4}$$

$$n_x + n_y = k \tag{4.5}$$

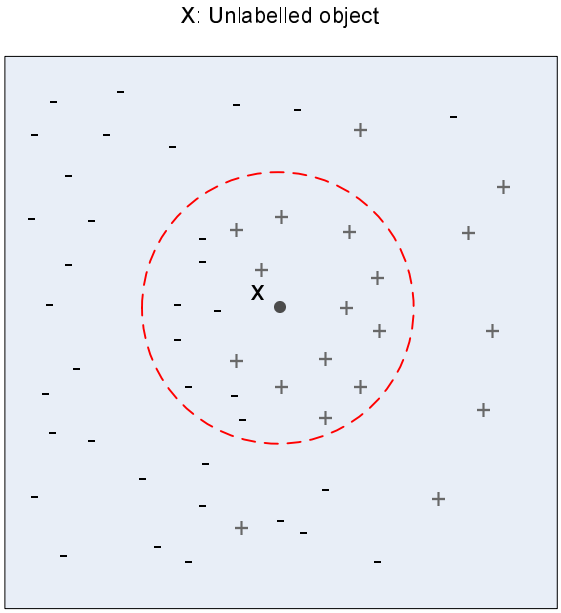


Figure 4.2: Illustration of the active learning strategy in our framework. We define $k = 20$ for kNN algorithm. In this example, $P_{ix} = 60\%$, $P_{iy} = 40\%$, so $G_i = 0.9710$. Attention: the base of log is 2.

where n_x is the number of images which belongs to the class $\mathbf{x}$ by kNN; likewise n_y. In order to derive the expected information gain when we label a certain image, we define the uncertainty measurement as follows:

$$G_i = \Phi(P_{ix}, P_{iy}), \qquad i = 1, 2, ..., N \tag{4.6}$$

where G_i is the information measurement and $\Phi(\cdot)$ is a function on the class probabilities of image I_i. Moreover, we employ the entropy to define the information measurement as

$$G_i = \Phi(P_{ix}, P_{iy}) = -P_{ix} \log P_{ix} - P_{iy} \log P_{iy} \tag{4.7}$$

Figure 4.3 is our proposed methodology for image retrieval tasks in CBIR.

During the iterations of this strategy, we need specify the value of k in kNN algorithm. It could be learned, or we can just assign a reasonable number to it. After the iterations of

Algorithm *$BMPM_{active}$ Loop Summary*
1. Randomly pick n_0 images from the pool and check their labels
2. Loop:
3. Learn a BMPM on the current images whose labels are known
4. Select m images from the dataset based on the criterion of Eq. (4.7) with the *Top-m* highest values, and label them
5. Loop till local optimal boundary achieved or get to maximum number of iterations

Figure 4.3: $BMPM_{active}$ Loop Summary.

relevance feedback have been performed, $BMPM_{active}$ returns the $Top-n$ most relevant images and learn a final BMPM based on the label known images.

Algorithm *$BMPM_{active}$ Final Output*
1. Learn a final BMPM from the labelled images
2. This decision line maybe the a local optimal one for the whole image dataset
3. The final BMPM boundary separates relevant images from irrelevant ones

Figure 4.4: $BMPM_{active}$ Final Output.

When we train and engage BMPM active learning framework in CBIR task, the choice of parameters is very direct, for example a typical settings could be n_0=10, m=10, k=100 and n=50. Users can also set them empirically by experiences.

4.4 Experimental Results

In this section, we show the experimental results. The performance of our proposed approach (we name it $BMPM_{active}$) are compared with three different models for relevance feedback: SVM, MPM and BMPM-based framework without active learning. All of them are based on Radial Basis Function Kernel. The experiments are evaluated on two real-world image datasets: a two-category and a ten-category image datasets. These image datasets were collected from COREL Image CDs. All our works

Figure 4.5: Example Images from COREL Image Database.

are done on a 3.2GHz machine with Intel Pentium 4 processor and 1Gb RAM.

4.4.1 Experiment Setup

COREL Image Datasets

The real-world images are chosen from the COREL image CDs. We organize the datasets which contain various images with different semantic meanings, such as *bird*, *pyramid*, *model*, *autumn*, *dog* and *glacier*, etc.

(A) *Two-Bird set*. The 180 images in this dataset belong to two groups - *bird* which contains 80 images, and *pyramid* which consists of 100 images. And we assume the category of *bird* is the relevant class.

(B) *Ten-Dog set*. The 980 images in this dataset fall into ten categories - *dog*, *autumn*, *bird*, *pyramid*, *Berlin*, *model*, *church*, *wave*, *tiger*, *Kenya*. In this dataset we assign the class of *dog* to be the user desired group and it contains 80 images. The other categories with each having 100 images belong to the irrelevant class.

Image Representation

For the real-world image retrieval, the image representation is an important step for evaluating the relevance feedback algorithms. We extract three different features to represent the images: *color*, *shape* and *texture*.

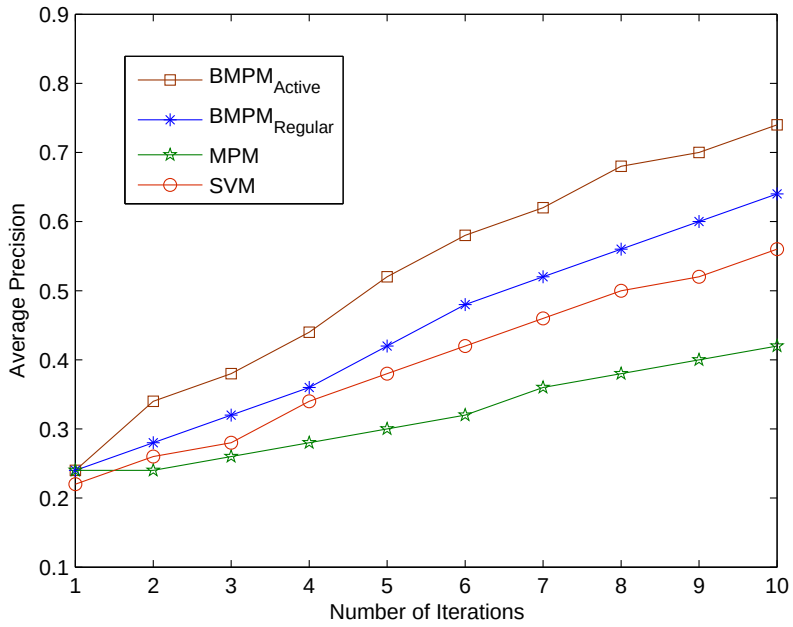


Figure 4.6: Evaluation on Two-Bird Dataset: Top-50 returned images are evaluated.

The color feature employed is the color histograms since it is closer to human natural perception and widely used in image retrieval. We quantized the number of pixels into 10 bins for each color channel (Hue, Saturation, and Value) respectively. Thus we could get a 30-dimensional color histogram.

We use edge direction histogram as shape feature [42]. We first calculate the edge images by Canny edge detector and obtain the edge direction histogram by quantize it into 15 bins of 24 degrees. Therefore a 15-dimensional edge direction histogram is generated as the edge feature.

Texture is an important cue for image feature extraction. We apply the wavelet-based texture in our experiments [91]. Gabor Wavelet Decomposition [20] is first performed and we compute the features for each Gabor filter output afterwards. Following this approach we use a 16-dimensional vector to describe the texture information for each image.

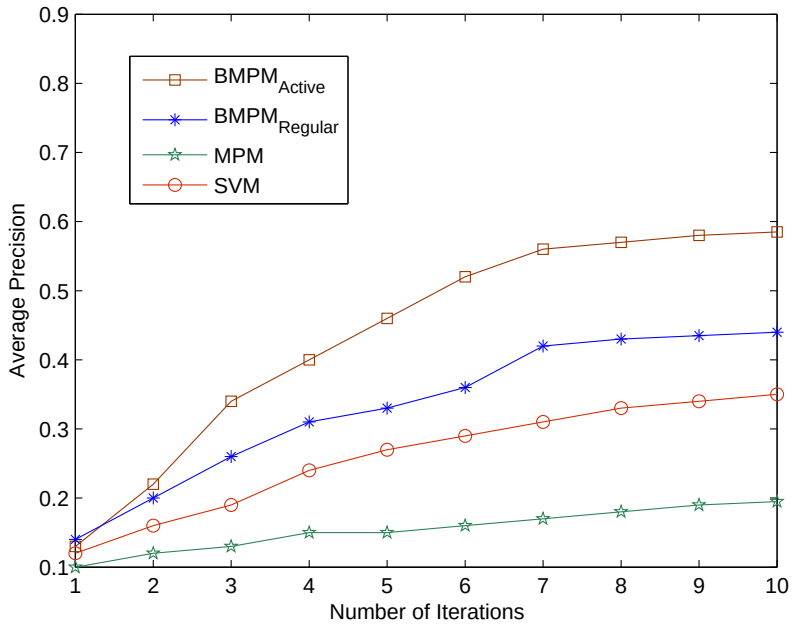


Figure 4.7: Evaluation on Ten-Dog Dataset: Top-50 returned images are evaluated.

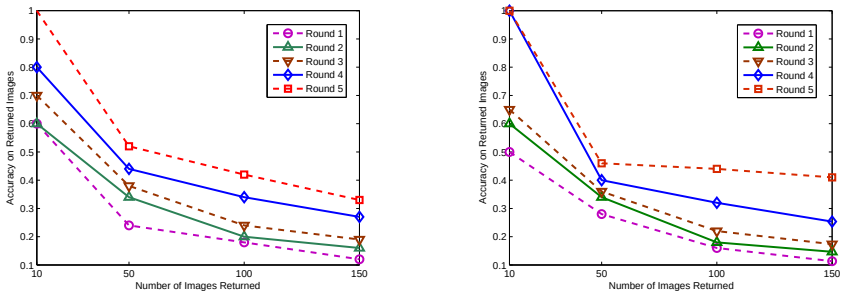
4.4.2 Performance Evaluation

In the following, we present the experimental results by this algorithm on real-world images. The metric of evaluation is *Average Precision* which is defined as the average ratio of the number of relevant images in the returned images over the total number of the returned images.

Precision and Recall are two very important performance measurements, which are defined as the following,

$$\begin{aligned} \mathbf{Pre} &= \frac{Image_{Relevant}}{\mathbf{Image_{Returned}}}, \\ \mathbf{Rec} &= \frac{Image_{Relevant}}{\mathbf{Image_{Relevant\_in\_total}}}. \end{aligned} \tag{4.8}$$

where $\mathbf{Image_{Returned}}$ is the number of images returned in each iteration, $Image_{Relevant}$ is the number of relevant images retrieved, and $\mathbf{Image_{Relevant\_in\_total}}$ is the total number of relevant images in the pool. In general, recall increases as more images



(a) (b)

Figure 4.8: (a) Average Top-n accuracy over the Two-Bird dataset. (b) Average Top-n accuracy over the Ten-Dog dataset.

are retrieved while precision decreases.

Since we define n_0=10, m=10, and n=50 in the experiments, two positive examples and eight negative examples are randomly picked from the dataset for the first iteration, then $BMPM_{Active}$, $BMPM_{Regular}$, MPM, and SVM are applied with the same start point. For the iterations afterward, all the methods select 10 image based on their own strategies. For SVM-based method in our evaluation we select images closest to the boundary from the dataset. In the iterative procedures, the number of returned relevant images is recorded, and the maximum loop used to obtain the average precision is set to be 10 times for all the methods.

Fig. 4.6 and 4.7 show the evaluation results on the Two-Bird dataset and Ten-Dog dataset. From the results on the real-world image datasets, we can see that our proposed framework outperforms the other approaches, especially in the Ten-Dog dataset. The reason is that $BMPM_{Active}$ based framework can reach the optimal solution even when the imbalanced dataset has a large size. Fig. 4.8 shows the average top-n accuracy for the two different sizes of datasets. We considered the performance of $BMPM_{Active}$ after each round of relevance feedback. The graphs indicate that the performance noticeably increases after each round. Furthermore, the performance of $BMPM_{Active}$

Learning Models	Top50@2-Bird	Top30@2-Bird	Top20@2-Bird
$BMPM_{Active}$	**0.73** ↑	**0.79** ↑	**0.88** ↑
$BMPM_{Regular}$	0.63	0.70	0.74
MPM	0.41	0.54	0.59
SVM	0.52	0.65	0.72
Learning Models	Top50@10-Dog	Top30@10-Dog	Top20@10-Dog
$BMPM_{Active}$	**0.58** ↑	**0.77** ↑	**0.83** ↑
$BMPM_{Regular}$	0.41	0.59	0.65
MPM	0.20	0.37	0.40
SVM	0.33	0.46	0.49

Table 4.1: Average Precision after 10 Iterations

framework degrades when the size and complexity of the dataset are increased, which could be observed from the Figures.

In order to observe the detailed comparison of the four methods after 10 iterations, we list the retrieval results in table 4.1. From the results, we can also see the similar results matching the above comparisons. In the tables we notice that when $BMPM_{Active}$ return most of the relevant images from the pool within 10 iterations while for other approaches they take more than 10 iterations. From this point we could say $BMPM_{active}$ based method achieves the optimal decision line much earlier than other algorithms. Readers can also observe that the performance of all these models on 2-Bird dataset is much better than the one on 10-Dog dataset. That's because of the different size and complexity of the datasets.

4.5 Summary

In this chapter, we address the problem of biased classification needed by the relevance feedback in CBIR and present a novel learning tool, BMPM Active Learning, to treat this problem more precisely and efficiently. In contrast to the traditional

methods, the BMPM provides a more elegant way to handle biased classification tasks. We evaluate the performance of the BMPM based algorithm on the COREL image dataset and obtain promising retrieval results.

□ **End of chapter.**

Chapter 5

Large Scale Learning with BMPM

The Biased Minimax Probability Machine (BMPM) constructs a classifier which deals with 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 chapter, 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 Program (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 derivations of the biased classification model, and reformulate it into an 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.

5.1 Introduction

Biased classifiers have many applications [34]. 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.

BMPM has emerged as a good classification technique, especially in imbalanced classification problem, and has achieved excellent generalization performance in a wide variety of applications. 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 points.

5.1.1 Motivation

BMPM has been extensively studied as a state-of-the-art learning techniques in various areas, such as bioinformatics [37, 38], information retrieval [69, 70] and statistical learning [35]. Most of recent studies on BMPM are generally based on the Fractional Program problem (we name it $BMPM_{FP}$) which could be solved by Rosen Gradient method. However the problem formulation has some crucial assumptions which would lead to failure of the model. Another issue is that when applying the Fractional Program (FP)-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 reformulate the model into an 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.

5.1.2 Contribution

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 an SOCP problem instead of an FP one. Under this new proposed framework, the large scale imbalanced classification problem could be modelled and solved efficiently.

5.2 Background Review

5.2.1 Second Order Cone Program

In Second Order Cone Programs a linear function is minimized over the intersection of an affine set and the product of second order (quadratic) cones. SOCPs are nonlinear convex problems that include linear and quadratic (convex) programs as special cases, but are less general than Semi-definite programs (SDPs). Several efficient primal-dual interior-point methods for SOCP have been developed in the last few years [4].

A Second Order Cone Programming problem is as the following form:

$$\begin{aligned} \min \quad & f^T x \\ \text{s.t.} \quad & \| A_i x + b_i \| \leq c_i^T x + d_i \quad i = 1, \ldots, N, \end{aligned} \tag{5.1}$$

where $x \in \mathbb{R}^n$ is the optimization variable, and the problem parameters are $f \in \mathbb{R}^n$, $A_i \in \mathbb{R}^{(n_i-1)\times n}$, $b_i \in \mathbb{R}^{n_i-1}$, $c_i \in \mathbb{R}^n$, and $d_i \in \mathbb{R}$. The norm appearing in the constraints is the standard Euclidean norm, i.e., $\| u \| = (u^T u)^{\frac{1}{2}}$. And we call the constraint

$$\| A_i x + b_i \| \leq c_i^T x + d_i \tag{5.2}$$

a second order cone constraint of dimension n_i. The SOCP problem in Eq. (5.1) is a convex programming problem since the objective function and the constraints define a convex set.

Second order cone constraints can be used to represent several common convex constraints [58]. For example, when $n_i = 1$ for $i = 1, \ldots, N$, the SOCP reduces to the Linear Program (LP)

$$\begin{array}{ll} \min & f^T x \\ \text{s.t.} & 0 \leq c_i^T x + d_i \quad i = 1, \ldots, N. \end{array} \tag{5.3}$$

Another interesting special case arises when $c_i = 0$, so the ith second order cone constraint reduces to $\| A_i x + b_i \| \leq d_i$, which is equivalent (assuming $d_i \geq 0$) to the (convex) quadratic constraint $\| A_i x + b_i \|^2 \leq d_i^2$. Thus, when all c_i vanish, the SOCP reduces to a quadratically constrained linear program (QCLP). Furthermore, (convex) quadratic programs (QPs), quadratically-constrained quadratic programs (QCQPs), and many other non-linear convex optimization problems can be reformulated as SOCPs as well [50].

5.2.2 General Methods for Large Scale Problems

Learning from the data is one of the basic ways that human perceive the world and acquire the knowledge. Nowadays, there are massive amounts of data available at an astonishingly increasing pace on the Internet and in industrial applications. There are classification tasks with a large number of classes such as the retrieval from image database with more than 1,000 classes. In a problem of categorization of Web documents, gigabytes of data with high dimension are processed. How to learn the patterns from a huge volume of dataset is a crucial problem.

Although many methods for solving the optimization problem of machine learning over large scale dataset are available [76, 77], we give a brief introduction on two prominent and popular strategies which can be used to train learning models, such as SVM, MPM etc, on a large dataset. They are *decomposition approach* and *analytical approach*.

The core of the decomposition algorithm is the divide and conquer strategy which is a general principle for solving complex problems [14]. In Computer Science, divide and conquer is an important algorithm design paradigm. It works by recursively breaking down a problem into two or more sub-problems of the same (or related) type, until these become simple enough to be solved directly. The solutions to the sub-problems are then combined to give a solution to the original problem. A divide and conquer algorithm is closely tied to a type of recurrence relation between functions of the data in question; data is divided into smaller portions and the result calculated thence. This technique is the basis of efficient algorithms for all kinds of problems, such as sorting (quicksort, merge sort) and the discrete Fourier transform (FFT).

For instance, training a Support Vector Machine requires the solution of a very large quadratic programming (QP) optimization problem. Sequential Minimal Optimization (SMO) breaks this large QP problem into a series of smallest possible QP problems. These small QP problems are solved analytically, which avoids using a time-consuming numerical QP optimization as an inner loop. The amount of memory required for SMO is linear in the training set size, which allows SMO to handle very large training sets [77].

Another direction for fast training algorithm in data mining is applying the optimization procedure during the model formulation, and then solve it analytically.

Lots of machine learning algorithms employ the technique which combines decomposition and analytical approach together for the learning task on large scale datasets.

5.2.3 Biased Minimax Probability Machine

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

Model 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 $\Sigma_{\mathbf{x}}$, $\Sigma_{\mathbf{y}} \in \mathbb{R}^{n\times n}$. Assuming $\{\overline{\mathbf{x}}, \Sigma_{\mathbf{x}}\}$, $\{\overline{\mathbf{y}}, \Sigma_{\mathbf{y}}\}$ for two classes of data are reliable, BMPM attempts to determine the hyperplane $\mathbf{a}^T\mathbf{z} = b$ 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. It is formulated as follows:

$$\begin{aligned} \max_{\alpha,\beta,b,\mathbf{a}\neq\mathbf{0}} \quad & \alpha \\ \text{s.t.} \quad & \inf_{\mathbf{x}\sim(\overline{\mathbf{x}},\Sigma_{\mathbf{x}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{x} \geq b\} \geq \alpha, \\ & \inf_{\mathbf{y}\sim(\overline{\mathbf{y}},\Sigma_{\mathbf{y}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{y} \leq b\} \geq \beta, \\ & \beta \geq \beta_0, \end{aligned} \tag{5.4}$$

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 in Eq. (5.4). 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.

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 in the following. According to the research effort by Huang *et al.*[38], we first borrow Lemma 2 from [54].

Lemma 5.1 *Given* $\mathbf{a} \neq \mathbf{0}$*,* b *such that* $\mathbf{a}^T\mathbf{y} \leq b$ *and* $\beta \in [0, 1)$*, the condition*

$$\inf \boldsymbol{Pr}\{\mathbf{a}^T\mathbf{y} \leq b\} \geq \beta$$

holds if and only if $b - \mathbf{a}^T\overline{\mathbf{y}} \geq \kappa(\beta)\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}$ *with* $\kappa(\beta) = \sqrt{\frac{\beta}{1-\beta}}$ *.*

This lemma can be proved by using the Lagrangian multiplier method and the work presented in [54]. Interested readers could refer to [54] for more detailed description.

By using Lemma 5.1, we obtain the following transformed optimization problem:

$$\max_{\alpha,\beta,b,\mathbf{a}\neq\mathbf{0}} \quad \alpha \tag{5.5}$$

$$\text{s.t.} \quad -b + \mathbf{a}^T\overline{\mathbf{x}} \geq \kappa(\alpha)\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}\,, \tag{5.6}$$

$$b - \mathbf{a}^T\overline{\mathbf{y}} \geq \kappa(\beta)\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}\,, \tag{5.7}$$

$$\beta \geq \beta_0\,, \tag{5.8}$$

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

From constraints in Eq. (5.6) and Eq. (5.7), 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 further modified to

$$\max_{\alpha,\beta,\mathbf{a}\neq\mathbf{0}} \quad \kappa(\alpha) \tag{5.9}$$

$$\text{s.t.} \quad 1 \geq \kappa(\alpha)\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}} + \kappa(\beta)\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}\ , \tag{5.10}$$

$$\mathbf{a}^T(\overline{\mathbf{x}} - \overline{\mathbf{y}}) = 1\ , \tag{5.11}$$

$$\kappa(\beta) \geq \kappa(\beta_0)\ , \tag{5.12}$$

where Eq. (5.12) is equivalent to Eq. (5.8) due to the monotonic property of the function $\kappa(\cdot)$.

The maximum value of $\kappa(\alpha)$ under the constraints of Eqs (5.10-5.12) is achieved when the right hand side of Eq. (5.10) is strictly equal to 1. Otherwise we could always get a new solution constructed by increasing $\kappa(\alpha)$ with a small positive amount while maintaining $\kappa(\beta)$ and $\mathbf{a}$ unchanged, which will satisfy the constraints, and will be a better solution.

Considering $\mathbf{\Sigma_x}$ and $\mathbf{\Sigma_y}$ can be regarded as positive definite matrices, we obtain $\kappa(\alpha) = \frac{1-\kappa(\beta)\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}}{\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}}$ from Eq. (5.10). The objective function is transformed into the following:

$$\max_{\kappa(\beta),\mathbf{a}\neq\mathbf{0}} \quad \frac{1-\kappa(\beta)\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}}{\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}}\ , \tag{5.13}$$

which is a linear function with respect to $\kappa(\beta)$, and $\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}$ which is a positive term. Therefore, this optimization function is maximized when $\kappa(\beta)$ is set to its lower bound $\kappa(\beta_0)$. Thus, the BMPM optimization problem is changed to:

$$\begin{aligned} \max_{\mathbf{a}\neq\mathbf{0}} \quad & \frac{1-\kappa(\beta_0)\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}}{\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}} \\ \text{s.t.} \quad & \mathbf{a}^T(\overline{\mathbf{x}} - \overline{\mathbf{y}}) = 1, \end{aligned} \tag{5.14}$$

which is an FP problem in the following form,

$$\begin{array}{ll} \max\limits_{\mathbf{a}\neq\mathbf{0}} & \frac{f(\mathbf{a})}{g(\mathbf{a})} \\ \text{s.t.} & \mathbf{a} \in A = \{\mathbf{a} | \mathbf{a}^T(\overline{\mathbf{x}} - \overline{\mathbf{y}}) = 1\}, \end{array} \tag{5.15}$$

where $f(\mathbf{a}) = 1 - \kappa(\beta_0)\sqrt{\mathbf{a}^T\Sigma_\mathbf{y}\mathbf{a}}$, $\quad g(\mathbf{a}) = \sqrt{\mathbf{a}^T\Sigma_\mathbf{x}\mathbf{a}}$. Furthermore, it is easy to see that the domain A is a convex set on $\mathbb{R}^n$, $f(\mathbf{a})$ and $g(\mathbf{a})$ are differentiable on A. Moreover, $f(\mathbf{a})$ is a concave function on A and $g(\mathbf{a})$ is a convex function on A. Then $\frac{f(\mathbf{a})}{g(\mathbf{a})}$ is a concave-convex FP problem. Hence it is strictly quasiconcave on A according to [37], and is solvable.

In its original work of this model, Rosen Gradient Projection method [3] is employed to find the solution of this concave-convex FP problem. It is observed that the inequalities in Eq. (5.6) and Eq. (5.7) will become equalities at the optimal point. The optimal b^* will thus be obtained by

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

5.3 Efficient BMPM Training

In this section, we present our research effort on the efficient training issue on BMPM model. We state the main result, and then introduce the kernelization procedure of BMPM model.

5.3.1 Proposed Strategy

Our main result is stated below.

Theorem 5.2 *If $\overline{\mathbf{x}} = \overline{\mathbf{y}}$, then the minimax probability decision problem of Eq. (5.4) does not have a meaningful solution: the optimal worst-case misclassification probability that we obtain is 1 - a^* =1. Otherwise, an optimal hyperplane $H(\mathbf{a}^*, b^*)$ exists,*

and can be determined by solving the convex optimization problem:

$$\begin{array}{ll} \min\limits_{t,\mathbf{a}\neq\boldsymbol{0}} & t-\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}}) \\ s.t. & \parallel \Sigma_x^{\frac{1}{2}}\mathbf{a} \parallel \leq 1, \\ & \parallel \Sigma_y^{\frac{1}{2}}\mathbf{a} \parallel \leq \sqrt{\frac{1-\beta_0}{\beta_0}}t, \end{array} \tag{5.16}$$

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 an optimal solution of Eq. (5.16), and $t \in \mathbb{R}$ is a new optimization variable. The optimal worst-case misclassification probability for class $\mathbf{x}$ and $\mathbf{y}$ is

$$\mathbf{Pr}(Misclassification_{\mathbf{x}}) = 1-\alpha^*, \tag{5.17}$$

$$\mathbf{Pr}(Misclassification_{\mathbf{y}}) = 1-\beta_0, \tag{5.18}$$

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

Proof 1 *It is observed that the optimization problem of Eq. (5.4) could be transformed to the following format:*

$$\begin{array}{ll} \max\limits_{\alpha,b,\mathbf{a}\neq\mathbf{0}} & \alpha \\ s.t. & \inf\limits_{\mathbf{x}\sim(\bar{\mathbf{x}},\Sigma_{\mathbf{x}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{x} \geq b\} \geq \alpha, \\ & \inf\limits_{\mathbf{y}\sim(\bar{\mathbf{y}},\Sigma_{\mathbf{y}})} \mathbf{Pr}\{\mathbf{a}^T\mathbf{y} \leq b\} \geq \beta_0. \end{array} \tag{5.19}$$

By using Lemma 5.1, the above optimization becomes:

$$\begin{array}{ll} \max\limits_{\alpha,\mathbf{a}\neq\mathbf{0}} & \alpha \\ s.t. & \sqrt{\frac{\alpha}{1-\alpha}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}} + \sqrt{\frac{\beta_0}{1-\beta_0}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}} \leq \mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}}). \end{array} \tag{5.20}$$

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

$$\begin{array}{ll} \max\limits_{\alpha,\mathbf{a}\neq\mathbf{0}} & \sqrt{\frac{\alpha}{1-\alpha}} \\ s.t. & \sqrt{\frac{\alpha}{1-\alpha}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}} + \sqrt{\frac{\beta_0}{1-\beta_0}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}} \leq \mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}}). \end{array} \tag{5.21}$$

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

$$\begin{array}{ll} \max\limits_{\alpha,\mathbf{a}\neq\mathbf{0}} & \sqrt{\frac{\alpha}{1-\alpha}} \\ s.t. & \sqrt{\frac{\alpha}{1-\alpha}} \leq \frac{\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}})-\sqrt{\frac{\beta_0}{1-\beta_0}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}}{\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}}, \end{array} \tag{5.22}$$

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

$$\max_{\mathbf{a}\neq\mathbf{0}} \ \frac{\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}})-\sqrt{\frac{\beta_0}{1-\beta_0}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}}}{\sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}}}. \tag{5.23}$$

It is observed that optimization problem of Eq. (5.23) 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[1]*. Furthermore maximization of an item is equivalent to minimize its opponent. Hence, we could obtain the transformed problem as*

$$\begin{array}{ll} \min\limits_{\mathbf{a}\neq\mathbf{0}} & -\mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}}) + \sqrt{\frac{\beta_0}{1-\beta_0}}\sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}} \\ s.t. & \sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}} \leq 1. \end{array} \tag{5.24}$$

And it could be further transformed to

$$\begin{array}{ll} \min\limits_{t,\mathbf{a}\neq\mathbf{0}} & t - \mathbf{a}^T(\overline{\mathbf{x}}-\overline{\mathbf{y}}) \\ s.t. & \sqrt{\mathbf{a}^T\Sigma_{\mathbf{x}}\mathbf{a}} \leq 1, \\ & \sqrt{\mathbf{a}^T\Sigma_{\mathbf{y}}\mathbf{a}} \leq \sqrt{\frac{1-\beta_0}{\beta_0}}t. \end{array} \tag{5.25}$$

[1]This is a common technique to tackle optimization problems.

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

$$\begin{aligned} \min_{t,\mathbf{a}\neq\mathbf{0}} \quad & t - \mathbf{a}^T(\overline{\mathbf{x}} - \overline{\mathbf{y}}) \\ s.t. \quad & \| \Sigma_{\mathbf{x}}^{\frac{1}{2}}\mathbf{a} \| \leq 1, \\ & \| \Sigma_{\mathbf{y}}^{\frac{1}{2}}\mathbf{a} \| \leq \sqrt{\tfrac{1-\beta_0}{\beta_0}}t. \end{aligned} \tag{5.26}$$

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 Theorem 5.2.

Lemma 5.3 *The Second Order Cone Programming problem with linear objective function and norm constraints is a convex optimization problem and thus can be solved efficiently.*

Proof 2 *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 [98].

5.3.2 Kernelized BMPM 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 [65].

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:

$$\begin{aligned} \mathbf{x} \to \varphi(\mathbf{x}) &\sim (\overline{\varphi(\mathbf{x})}, \mathbf{\Sigma}_{\varphi(\mathbf{x})}) \ , \\ \mathbf{y} \to \varphi(\mathbf{y}) &\sim (\overline{\varphi(\mathbf{y})}, \mathbf{\Sigma}_{\varphi(\mathbf{y})}) \ , \end{aligned}$$

where $\varphi : \mathbb{R}^n \to \mathbb{R}^f$ 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:

$$\begin{aligned} \min_{t,\mathbf{a}\neq\mathbf{0}} \quad & t - \mathbf{a}^T(\overline{\varphi(\mathbf{x})} - \overline{\varphi(\mathbf{y})}) \\ \text{s.t.} \quad & \| \mathbf{\Sigma}^{\frac{1}{2}}_{\varphi(\mathbf{x})}\mathbf{a} \| \leq 1, \\ & \| \mathbf{\Sigma}^{\frac{1}{2}}_{\varphi(\mathbf{y})}\mathbf{a} \| \leq \sqrt{\tfrac{1-\beta_0}{\beta_0}}t. \end{aligned} \tag{5.27}$$

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. We are not going to present a detailed kernelization procedure here. It's a similar way as described in [39]. Readers interested in the details can refer to [39].

We give out the kernelized optimization function for BMPM as follows:

$$\begin{aligned} \min_{t,\mathbf{a}\neq\mathbf{0}} \quad & t - \mathbf{w}^T(\tilde{\mathbf{k}}_\mathbf{x} - \tilde{\mathbf{k}}_\mathbf{y}) \\ \text{s.t.} \quad & \sqrt{\tfrac{1}{N_\mathbf{x}}\mathbf{w}^T\tilde{\mathbf{K}}^T_\mathbf{x}\tilde{\mathbf{K}}_\mathbf{x}\mathbf{w}} \leq 1, \\ & \sqrt{\tfrac{1}{N_\mathbf{y}}\mathbf{w}^T\tilde{\mathbf{K}}^T_\mathbf{y}\tilde{\mathbf{K}}_\mathbf{y}\mathbf{w}} \leq \sqrt{\tfrac{1-\beta_0}{\beta_0}}t, \end{aligned} \tag{5.28}$$

which is also a SOCP problem that has the similar form as the one in Eq. (5.16) and can thus be solved in a similar way. The notations in the above are defined similar to [37]. For an easy reference, we summarize them in Tables **??-??**.

5.4 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

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

Table 5.1: An overview of Reuters-21578 dataset with 10 major classes

empirical comparisons on three standard datasets for text document classification: *Reuters-21578 dataset*, *20-Newsgroup data collection* and *Enron Corpus dataset*. For all three datasets, the same 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 [122]. In particular, 500 of the most informative features are selected for each document in the three datasets which is similar to the technique presented in [30].

5.4.1 Experimental Testbeds

The first dataset is the Reuters-21578 Corpus dataset [2], which has been broadly used as a benchmark dataset for evaluating algorithms for text classification. 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 5.1 shows a

[2]http://www.daviddlewis.com/resources/testcollections/reuters21578/

class	number of messages
user 1	982
user 2	301
user 3	1306
user 4	2747
user 5	493
user 6	948
user 7	1493
user 8	264
user 9	1367
user 10	751

Table 5.2: A list of 10 selected users from the Enron Corpus dataset in our experiments

list of the 10 most frequent topics contained in the dataset [30]. 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 class without consideration of the document being 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.

The other two datasets are 20-Newsgroup data collection [3] and the Enron Corpus dataset [4]. 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 belongs 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 issue, we select 3 out of 20 newsgroups with re-

[3] http://people.csail.mit.edu/jrennie/20Newsgroups/

[4] http://www.cs.cmu.edu/∼enron/

lated topics and define them as the interested class in our study, which are *talk.politics.misc*, *talk.politics.guns* and *talk.politics.mideast*. Apart from that, the others are regarded as uninterested class. In the Enron Corpus, there are a total of 200,399 messages belonging to 158 users with an average of 757 messages per user. In our study, we random select 10 out of 158 users for evaluation. Each users contains approximately 180 messages. Table 5.2 shows the details of this data collection used in our experiments. In this selected dataset, we random define one particular user as the interested category in our evaluation while others are regarded as uninterested one. During this process, we conduct our evaluations on 5 different selected users against the others, and get the average performance.

5.4.2 Experimental Settings

Applying the 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 assume 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 [118]. It has been shown to be a more reliable metric than other metrics when conducted on imbalanced classification problems. 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 curve, the higher the sensitivity for a given specificity, and hence the better the method's performance [38].

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 sub-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 proposed learning model and solving strategy, three reference models are used in our experiments. The first reference model is SVM which is a state-of-the-art text classification technique. The second reference model is based on kNN algorithm which is a traditional classification model. We also include MPM for performance comparison intention. Finally, BMPM has been conducted based on both Fractional Programming and Second Order Cone Programming strategies. By comparing with these three models, we are able to determine the BMPM model is more reliable to handle the imbalanced text classification problem, and the advantages of our proposed training approach.

To deploy efficient implementation of our scheme toward imbalanced text classification tasks, both the $BMPM_{SOCP}$ and the $BMPM_{FP}$ frameworks used in this study are programmed in the Matlab language while SVM and kNN are in C language. The testing hardware environment is on a Windows XP system with 3.2GHz CPU and 1GB physical memory. To implement the SOCP-based BMPM algorithm for our text classification tasks, we adopt the standard optimization package, i.e., SeDuMe [98] and YALMIP [59], to solve the Second Order Cone Programming problem in our algorithm efficiently. The FP-based BMPM framework is based on the Rosen Gradient Projection method described in [37]. For the base model of MPM, we adopt the code shared by Lanckriet[5]. The SVM^{light} package is used in our experiments for the implementation of

[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 5.3: Lower Bound α and Test-Set Accuracy on the Reuter-21578 dataset (%)

SVM, which has been considered as the state-of-the-art tool for text classification[6]. Since SVM is parameter sensitive, we conduct evaluations on a separate validation data collection to determine the optimal parameters for deployment. Furthermore, we adopt the kNN package implemented by Mount and Arya [7].

5.4.3 Performance Evaluation

In this section, we will first describe the results for the Test-Set Accuracy performance on three datasets respectively. They all have been extensively studied for text classification problems. We will then provide the empirical results for the other two measurements in these data collections: ROC analysis and training time comparison.

Test-Set Accuracy Comparison

Table 5.3 shows the experimental results of 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 columns of Table 5.3, we observe that the performance of the two classifiers, $BMPM_{SOCP}$ and $BMPM_{FP}$, outperform the other three

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

[7]http://www.cs.umd.edu/∼mount/ANN/

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. Given the motivation that we want to make the accuracy of the more important class as high as possible, this demonstrates the efficiencies of $BMPM$ for biased classification problems though the parameter β in both $BMPMs$ are worse than the one in MPM. 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 5.3. We find that the average TSA performance, which is indicated as TSA in the table, of these two learning methods becomes close with the BMPM models. But for the TSA of the more important class indicated as $TSA_{\mathbf{x}}$ is much lower than BMPM models. For example, the $TSA_{\mathbf{x}}$ 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 Program based algorithm, i.e., $BMPM_{SOCP}$, to the Fractional Program based methodology $BMPM_{FP}$. It is evident that our proposed learning algorithm outperforms its original approach. For both α and $TSA_{\mathbf{x}}$, which denote the classification accuracy of the more important pattern, the proposed algorithm $BMPM_{SOCP}$ is able to outperform the FP-based learning algorithm noticeably.

In order to evaluate the performance substantially, the classification results of the 20-Newsgroup dataset and the Enron Corpus dataset are listed in Table 5.4 and Table 5.5, respectively. 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. This result also indicates that the proposed learning algorithm is robust when there is a global

	$BMPM_{SOCP}$	$BMPM_{FP}$	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 5.4: Lower Bound α and Test-Set Accuracy on the 20-Newsgroup dataset (%)

	$BMPM_{SOCP}$	$BMPM_{FP}$	MPM	SVM	kNN
α	**76.20 ± 0.22** ↑	74.31 ± 0.14 ↑	69.82 ± 0.28	-	-
β	**70.00 ± 0.00**	70.00 ± 0.00	69.82 ± 0.28	-	-
TSA_x	**72.81 ± 0.26** ↑	71.42 ± 0.28 ↑	71.21 ± 0.34	54.60 ± 0.17	51.52 ± 0.32
TSA_y	**70.62 ± 0.61**	70.20 ± 0.57	67.28 ± 0.24	83.41 ± 0.57	79.30 ± 0.79
TSA	**71.70 ± 1.24**	70.81 ± 1.20	69.23 ± 1.43	67.45 ± 0.29	66.40 ± 0.81

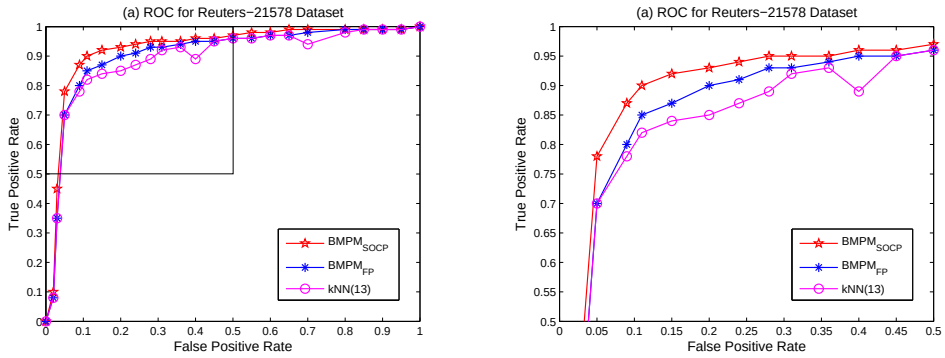
Table 5.5: Lower Bound α and Test-Set Accuracy on the Enron Corpus dataset (%)

optima needed while the FP-based method may suffer critically with the assumptions during the learning model formulation.

ROC Curve Analysis

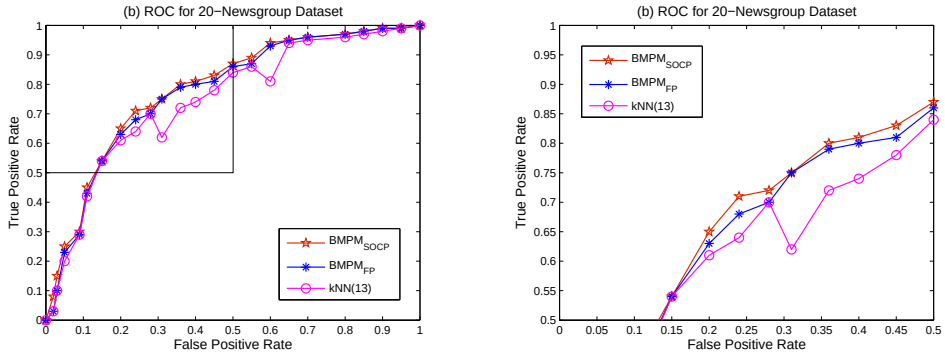
We now compare our BMPM models with kNN in terms of the ROC curve analysis. We generate the ROC curves as illustrated in left parts of Figs (5.1 – 5.3). Note that we do not involve MPM and SVM for comparison here, since it is not easy to generate the ROC curves for SVM and MPM due to their model settings.

It is observed that the $BMPM_{SOCP}$ and $BMPM_{FP}$ perform better than the kNN classifier for all three data collections, since the BMPM curves are above of the one for 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



(a) Full Range (b) Crucial Part

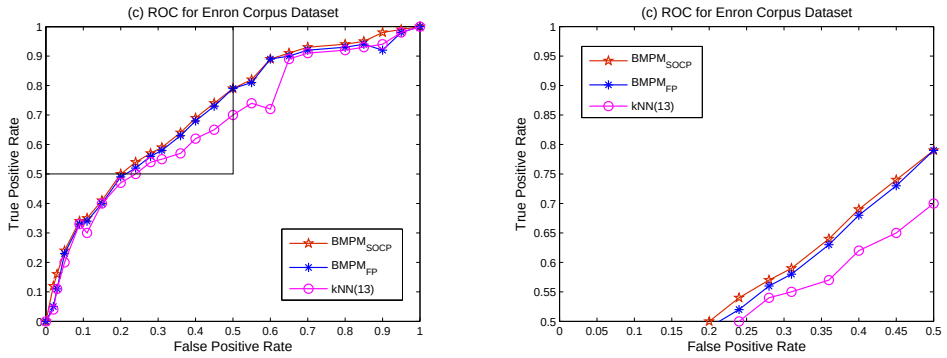
Figure 5.1: ROC Curve Performance Evaluation on Reuters-21578 Dataset.



(a) Full Range (b) Crucial Part

Figure 5.2: ROC Curve Performance Evaluation on 20-Newsgroup Dataset.

this, we show the critical portions in the right parts of Figs (5.1 – 5.3) 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 these critical regions, most parts of the ROC curves of BMPMs are above the corresponding ones of kNN model in all datasets along with the $BMPM_{SOCP}$ curves are above the ones of $BMPM_{FP}$, which again demonstrates the superiority of the BMPM models and our proposed $BMPM_{SOCP}$ algorithm.



(a) Full Range (b) Crucial Part

Figure 5.3: ROC Curve Performance Evaluation on Enron Corpus Dataset.

Training Time Comparison

We record the runtime when conducting experiments on the Reuters-21578 data collection. We divide 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 average performance.

Figure (5.4) compares the CPU-time of two $BMPMs$ and MPM on the task described above. It could be observed that $BMPM_{SOCP}$ is substantially faster than the other two models on all cases. From the experimental result, 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 find that the improvement of our algorithm is more evident compared with the other two approaches when the size of training instances is larger. This is because the larger the size of the problem, the better the performance we could expect. When more samples are conducted, the gap

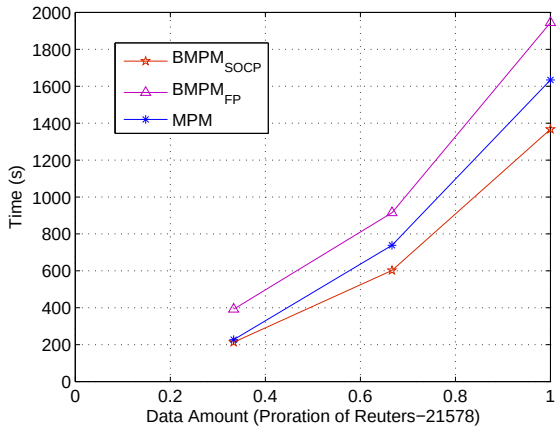


Figure 5.4: Training time performance of different models based on Matlab for Three-Phase Reuters-21578 dataset (*sec.*GHz*)

for performance 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.

5.5 Summary

The computational complexity of our method for BMPM is comparable to the quadratic program that one has to solve for SVM and MPM. While we have presented 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 multi-class classification.

□ **End of chapter.**

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Image retrieval is getting more and more popular now than ever before, due to the rapid growth of the Internet and the growing use of image information in government and commercial organizations. Many organizations produce huge volume of image data everyday. Facing the massive data volume, end users find that it is inefficient to browse a favorite image from Internet, and the content providers have to face the tedious work of managing the ever growing multimedia database. The urgent problem brings a lot of attention to Content-based Image Retrieval (CBIR), which is a state-of-the-art technology intending to solve the problem by providing the people expected images based on their content.

In conclusion, we have proposed an imbalanced learning based relevance feedback framework for CBIR, an active learning strategy for relevance feedback, and an efficient training algorithm for the Biased Minimax Probability Machine (BMPM) model. Our research work has the following contributions:

1. We propose a BMPM-based methodology to capture the user's preference in the relevance feedback process in which BMPM addresses the imbalanced dataset problem. The experimental results on both synthetic dataset and real-

world image collection demonstrate the effectiveness of our proposed approach.

2. We present an active learning framework with imbalanced learning theory to tackle the relevance feedback problem in CBIR. Performance evaluation has shown that our framework improves the performance compared to the traditional methods for relevance feedback problem in CBIR.

3. We propose a Second Order Cone Program based algorithm to solve BMPM model for large scale dataset learning tasks. Our analysis and evaluation of the proposed algorithm show that it is more efficient and accurate than its original solution which is a Fractional Program based method.

6.2 Future Work

In this thesis we describe the work we have done on relevance feedback problem in CBIR and efficient training algorithm on BMPM learning model.

A BMPM-based framework and an active learning framework have been proposed. The frameworks themselves are quite flexible; many other features and constraints can be added into both frameworks as their extensions. In the future, we may enhance the frameworks by incorporating better feature extraction methods and similarity measurement algorithms into our frameworks.

The problem of imbalanced classification has a long and distinguished history. Many results on misclassification rates have been obtained by making distributional assumptions. BMPM makes use of the moment-based inequalities of Marshall and Olkin to obtain distribution-free results for linear discriminants. By converting this model into a Second Order Cone Programming problem, the decision hyperplane is then determined more accurately and efficiently. The computational complexity of our

method is comparable to the quadratic program that one has to solve for SVM and MPM. While we have viewed this model from the angle 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, in particular procedures that break the data into subsets. Another direction that we are currently investigating is the extension of our model to multiway classification.

□ **End of chapter.**

Appendix A

List of Publications

Here is a list of publications during my master study:

1. **Xiang Peng** and Irwin King. Efficient Training on Biased Minimax Probability Machine for Imbalanced Text Classification. In *Poster Proceedings of the 16th International World Wide Web Conference (WWW2007)*, Banff, Alberta, Canada, May 8 – 12, 2007.

2. **Xiang Peng** and Irwin King. Large Scale Imbalanced Classification with Biased Minimax Probability Machine. In *Proceedings of the 20th International Joint Conference on Neural Networks (IJCNN2007)*, Orlando, Florida, USA, August 12 – 17, 2007.

3. **Xiang Peng** and Irwin King. Biased Minimax Probability Machine Active Learning for Relevance Feedback in Content-based Image Retrieval. In *Proceedings of the 7th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL2006)*, Pages 953 – 960, Burgos, Spain, September 20 – 23, 2006.

4. Ka Kan Lo, **Xiang Peng** and Irwin King. A User Profile-based Approach for Personal Information Access: Shaping Your Information Portfolio. In *Poster Proceedings of the 15th International Conference on World Wide Web*

(WWW2006), Pages 921 – 922, Edinburgh, Scotland, May 22 – 26, 2006.

5. **Xiang Peng** and Irwin King. Imbalanced Learning in Relevance Feedback with Biased Minimax Probability Machine for Image Retrieval Tasks. In *Proceedings of 13th International Conference on Neural Information Processing (ICONIP2006)*, Pages 342 – 351, Hong Kong, October 3 – 6, 2006.

□ **End of chapter.**

Bibliography

[1] N. Abe. Sampling approaches to learning from imbalanced datasets: active learning, cost sensitive learning and beyond. Invited Talk in ICML Workshop on Learning from Imbalanced Data Sets, Washington, DC, USA, 2003.

[2] M. Ankerst, B. Braunmuller, H. P. Kriegel, and T. Seidl. Improving adaptable similarity query processing by using approximations. In *Proceedings of 24rd International Conference on Very Large Data Bases*, pages 206–217, New York City, USA, 1998.

[3] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, second edition, 1999.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2003.

[5] L. Breiman, J. Friedman, C. Stone, and R. Olshen. *Classification and Regression Trees*. Chapman & Hall/CRC, 1984.

[6] K. Brinker. Incorporating diversity in active learning with support vector machines. In *Proceedings of 20th International Conference on Machine Learning*, pages 59–66, Washington, DC, USA, August 2003.

[7] S. Chandrasekaran, B. Manjunath, Y. Wang, J.Winkeler, and H. Zhang. An eigenspace update algorithm for image

analysis. *Graphical Models and Image Processing: GMIP*, 59(5):321–332, 1997.

[8] N. S. Chang and K. S. Fu. Query-by-pictorial-example. *IEEE Transactions on Software Engineering*, 6(6):519–524, November 1980.

[9] S. Chang, C. Yan, D. Dimitroff, and T. Arndt. An intelligent image database system. *IEEE Transactions on Software Engineering*, 14(5):681–688, May 1988.

[10] S. K. Chang and A. Hsu. Image information systems: Where do we go from here? *IEEE Transactions on Knowledge and Data Engineering*, 4(5):431–442, October 1992.

[11] Y. Chen, X. S. Zhou, and T. S. Huang. One-class svm for learning in image retrieval. In *Proceedings of International Conference on Image Processing*, pages 34–37, Thessalonica, Greece, 2001.

[12] I. J. Cox, M. L. Miller, T. P. Minka, T. V. Papathomas, and P. N. Yianilos. The bayesian image retrieval system, PicHunter: Theory, implementation and psychophysical experiments. *IEEE Transactions on Image Processing*, 9(1), January 2000.

[13] I. J. Cox, M. L. Miller, T. P. Minka, and P. N. Yianilos. An optimized interaction strategy for bayesian relevance feedback. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 553–558, Santa Barbara, USA, 1998.

[14] J. X. Dong, C. Y. Suen, and A. Krzyzak. Fast svm training algorithm with decomposition on very large data sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):603 – 618, 2005.

[15] H. Drucker, B. Shahrary, and D. Gibbon. Relevance feedback using support vector machines. In *Proceedings of 18th International Conference on Machine Learning*, pages 122–129, Williams College, Williamstown, MA, USA, 2001.

[16] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons Inc, 1973.

[17] W. Equitz and W. Niblack. Retrieving images from a database using texture – alogrithms from the QBIC system. Technical report, IBM Research, 1994.

[18] A. Estabrooks, T. Jo, and N. Japkowicz. A multiple resampling method for learning from imbalanced data sets. *Computational Intelligence*, 20(1):18–36, February 2004.

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

[20] J. Fiser and I. King. Gabor-wavelet decomposition based filtering of gray-level images for object recognition experiments. *Spatial Vision*, 11(1):117–119, 1998.

[21] G. Giacinto and F. Roli. Bayesian relevance feedback for content-based image retrieval. *Pattern Recognition*, 37(7):1499–1508, 2004.

[22] P. H. Gosselin and M. Cord. Active learning techniques for user interactive systems: Application to image retrieval. In *Proceedings of International Workshop on Machine Learning Techniques for Processing Multimedia Content*, pages 85–89, Bonn, Germany, 2005.

[23] C. C. Gotlieb and H. E. Kreyszig. Texture descriptors based on co-occurrence matrices. *Computer Vision, Graphics, and Image Processing*, 51:70–86, 1990.

[24] D. Greene. An implementation and performance analysis of spatial data access methods. In *Proceedings of International Conference on Data Engineering*, pages 606–615, Los Angeles, CA, USA, 1989.

[25] A. Guttman. R-tree: A dynamic index structure for spatial searching. In *Proceedings of Annual Meeting of ACM Special Interest Group on Management Of Data Conference*, pages 47–57, Boston, Massachusetts, USA, 1984.

[26] R. M. Haralick, K. Shanmugam, and I. Dinstein. Texture features for image classification. *IEEE Transactions on System Man and Cybernetics*, 3(6):103–112, 1973.

[27] C. H. Hoi, C. H. Chan, K. Huang, M. R. Lyu, and I. King. Biased support vector machine for relevance feedback in image retrieval. In *Proceedings of International Joint Conference on Neural Networks*, pages 3189–3194, Budapest, Hungary, July 2004.

[28] C. H. Hoi and M. R. Lyu. Group-based relevance feedback with support vector machine ensembles. In *Proceedings of the 17th International Conference on Pattern Recognition*, pages 874–877, Cambridge, UK, 2004.

[29] S. C. Hoi, M. R. Lyu, and R. Jin. A unified log-based relevance feedback scheme for image retrieval. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):509–524, April 2006.

[30] S. C. H. Hoi, R. Jin, and M. R. Lyu. Large-scale text categorization by batch mode active learning. In *Proceedings of 15th International Conference on World Wide Web*, pages 633–642, Edinburgh, Scotland, UK, 2006.

[31] S. C. H. Hoi, R. Jin, J. Zhu, and M. R. Lyu. Batch mode active learning and its application to medical image clas-

sification. In *Proceedings of 23th International Conference on Machine Learning*, pages 417–424, Pittsburgh, Pennsylvania, USA, 2006.

[32] P. Hong, Q. Tian, and T. S. Huang. Incorporate support vector machines to content-based image retrieval with relevance feedback. In *Proceedings of IEEE International Conference on Image Processing*, pages 750–753, Vancouver, BC, Canada, October 2000.

[33] M. K. Hu. Visual pattern recognition by moment invariants. *IEEE Transactions on Information Theory*, 8:179–187, 1962.

[34] K. Huang, H. Yang, I. King, and M. Lyu. Learning classifiers from imbalanced data based on biased minimax probability machine. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 558–563, Washington, DC, USA, July 2004.

[35] K. Huang, H. Yang, I. King, and M. Lyu. Learning large margin classifiers locally and globally. In *Proceedings of 21th International Conference on Machine Learning*, pages 51–59, Banff, Alberta, Canada, 2004.

[36] K. Huang, H. Yang, I. King, and M. Lyu. Imbalanced learning with a biased minimax probability machine. *IEEE Transactions on Systems, Man, and Cybernetics (Part B)*, 36(4):913–923, Auguest 2006.

[37] K. Huang, H. Yang, I. King, and M. Lyu. Maximizing sensitivity in medical diagnosis using biased minimax probability machine. *IEEE Transactions on Biomedical Engineering*, 53(5):821–831, May 2006.

[38] K. Huang, H. Yang, I. King, M. Lyu, and L. Chan. Biased minimax probability machine for medical diagnosis. In *Proceedings of 8th International Symposium on Artificial Intelligence and Mathematics*, pages 1103–1110, Fort Lauderdale, Florida, USA, 2004.

[39] 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.

[40] T. S. Huang, S. Mehrotra, and K. Ramachandran. Multimedia analysis and retrieval system (MARS) project. In *Proceedings of 33rd Annual Clinic on Library Application of Data Processing-Digital Image Access and Retrieval*, pages 100–117, New York, NY, USA, 1996.

[41] M. Ioka. A method of defining the similarity of images on the basis of color information. Technical report, IBM Research, Tokyo Research Laboratory, 1989.

[42] A. K. Jain and A. Vailaya. Shape-based retrieval: a case study with trademark image database. *Pattern Recognition*, 9:1369–1390, 1998.

[43] N. Japkowicz. The class imbalance problem: Significance and strategies. In *Proceedings of International Conference on Artificial Intelligence*, pages 111–117, Las Vegas, USA, 2000.

[44] N. Japkowicz. Class imbalances: Are we focusing on the right issue? In *Proceedings of ICML Workshop on Learning from Imbalanced Data Sets*, pages 123–128, Washington, DC, USA, 2003.

[45] N. Japkowicz and S. Stephen. The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6(5):429–449, 2002.

[46] D. Kapur, Y. N. Lakshman, and T. Saxena. Computing invariants using elimination methods. In *Proceedings of IEEE International Symposium on Computer Vision*, pages 97–102, Coral Gables, FL, USA, 1995.

[47] T. Kohonen. *Self-Organizing Maps*. Springer-Verlag, 2001.

[48] P. Koikkalainen. Progress with the tree-structured self-organizing map. In *Proceedings of 11th European Conference on Artificial Intelligence*, pages 211–215, Amsterdam, The Netherlands, 1994.

[49] P. Koikkalainen and E. Oja. Self-organizing hierarchical feature maps. In *Proceedings of International Joint Conference on Neural Networks*, pages 279–284, San Diego, CA, 1990.

[50] Y. J. Kuo and H. D. Mittelmann. Interior point methods for second-order cone programming and or applications. *Computational Optimization and Applications*, 28(3):255–285, 2004.

[51] J. Laaksonen, M. Koskela, S. Laakso, and E. Oja. Self-organising maps as a relevance feedback technique in content-based image retrieval. *Pattern Analysis and Applications*, 4:140–152, 2001.

[52] J. Laaksonen, M. Koskela, and E. Oja. PicSOM: Self-organizing maps for content-based image retrieval. In *Proceedings of International Joint Conference on Neural Networks*, pages 2470–2473, Washington, DC, USA, 1999.

[53] G. Lanckriet, L. E. Ghaoui, and C. Bhattacharyya. Minimax probability machine. In *Advances in Neural Information Processing Systems 14*, pages 801–807, Vancouver, British Columbia, Canada, 2001.

[54] 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.

[55] D. Lewis and J. Catlett. Heterogenous uncertainty sampling for supervised learning. In *Proceedings of 11th International Conference on Machine Learning*, pages 148–156, New Brunswick, USA, 1994.

[56] D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. In *Proceedings of 17th ACM International Conference on Research and Development in Information Retrieval*, pages 3–12, Dublin, IE, 1994.

[57] M. Lindenbaum, S. Markovitch, and D. Rusakov. Selective sampling for nearest neighbor classifiers. *Machine Learning*, 54(2):125–152, 2004.

[58] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Applications of second-order cone programming. *Linear Algebra and Applications*, 284:193–228, 1998.

[59] J. Lofberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *Proceedings of 13th IEEE International Symposium on Computer Aided Control System Design*, pages 284–289, Taipei, Taiwan, 2004.

[60] S. MacArthur, C. Brodley, and C. Shyu. Relevance feedback decision trees in content-based image retrieval. In *Proceedings of IEEE Workshop on Content-based Access of Image and Video Libraries*, pages 68–72, Washington, DC, USA, 2000.

[61] B. MacNamee, P. Cunningham, S. Byrne, and O. Corrigan. The problem of bias in training data in regression problems in medical decision support. *Artificial Intelligence in Medicine*, 24:51–70, 2002.

[62] C. S. McCamy, H. Marcus, and J. G. Davidson. A color-rendition chart. *Journal of Applied Photographic Engineering*, 2(3):95–99, 1976.

[63] T. Mitchell. *Machine Learning*. McGraw-Hill, New York, NY, 1997.

[64] M. Miyahara. Mathematical transform of (r,g,b) color data to munsell (h,s,v) color data. *Visual Communication and Image Process*, 1001:650–657, 1988.

[65] K. R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf. An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–201, 2001.

[66] W. Niblack, R. Barber, and et al. The QBIC project: Querying images by content using color, texture and shape. In *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, pages 173–187, San Jose, CA, USA, 1994.

[67] M. Ortega, Y. Rui, K. Chakrabarti, S. Mehrotra, and T. S. Huang. Supporting similarity queries in mars. In *Proceedings of the 5th ACM International Conference on Multimedia*, pages 403 – 413, Seattle, Washington, USA, 1997.

[68] J. M. Park. On-line learning by active sampling using orthogonal decision support vectors. *Computer Vision and Image Understanding*, 77(3):263–283, 2000.

[69] X. Peng and I. King. Biased minimax probability machine active learning for relevance feedback in content-based image retrieval. In *Proceedings of 7th International Conference on Intelligent Date Engineering and Automated Learning*, pages 953–960, Burgos, Spain, 2006.

[70] X. Peng and I. King. Imbalanced learning in relevance feedback with biased minimax probability machine for image retrieval tasks. In *Proceedings of the 13th International Conference on Nueral Information Processing*, pages 342–351, Hong Kong, 2006.

[71] X. Peng and I. King. Efficient training on biased minimax probability machine for imbalanced text classification. In *Proceedings of 16th International World Wide Web Conference*, Banff, Alberta, Canada, 2007.

[72] X. Peng and I. King. Large scale imbalanced classification with biased minimax probability machine. In *Proceedings of 20th International Joint Conference on Neural Networks*, Orlando, Florida, 2007.

[73] A. Pentland, R. W. Picard, and S. Sclaroff. Photobook: Content-based manipulation of image databases. *International Journal of Computer Vision*, 18(33):233 – 254, 1996.

[74] E. Persoon and K. S. Fu. Shape discrimination using fourier descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(3):388 – 397, 1986.

[75] R. W. Picard and T. P. Minka. Vision texture for annotation. *Multimedia Systems: Special Issue on Content-based Retrieval*, 3(1):3–14, 1995.

[76] J. C. Platt. Using analytic QP and sparseness to speed training of support vector machines. In *Advances in Neural Information Processing Systems 11*, pages 557–563, Denver, Colorado, USA, 1998.

[77] J. C. Platt. *Fast Training of Support Vector Machines using Sequential Minimal Optimization*, chapter 12, Ad-

vances in Kernel Methods – Support Vector Learning, pages 185 – 208. MIT Press, 1999.

[78] F. Provost. Machine learning from imbalanced data sets 101. In *Proceedings of AAAI Workshop on Learning from Imbalanced Data Sets*, pages 94 – 100, Menlo Park, California, USA, 2000.

[79] J. Quinlan. *C4.5: Programs for machine learning*. Morgan Kaufmann, San Mateo, CA, 1993.

[80] B. Raskutti and A. Kowalczyk. Extreme re-balancing for svms: A case study. *ACM SIGKDD Explorations Newsletter*, 6(1):60–69, 2004.

[81] N. Roy and A. McCallum. Toward optimal active learning through sampling estimation of error reduction. In *Proceedings of the 18th International Conference on Machine Learning*, pages 441–448, Williams College, Williamstown, MA, USA, June 2001.

[82] Y. Rui, T. S. Huang, and S. F. Chang. Image retrieval: Current techniques, promising directions and open issues. *Journal of Visual Communication and Image Representation*, 10(4):39–62, April 1999.

[83] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Content-based image retrieval with relevance feedback in mars. In *Proceedings of IEEE International Conference on Image Processing*, pages 815–818, Washington, DC, USA, October 1997.

[84] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Relevance feedback: A power tool in interactive content-based image retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(5):644–655, September 1998.

[85] Y. Rui, A. C. She, and T. S. Huang. Modified fourier descriptors for shape representation – a practical approach. In *Proceedings of 1st International Workshop on Image Databases and Multi Media Search*, pages 213–232, Amsterdam, The Netherlands, 1996.

[86] Y. Sakurai, M. Yoshikawa, R. Kataoka, and S. Uemura. Similarity search for adaptive ellipsoid queries using spatial transformation. *Very Large Data Base Journal*, pages 231–240, 2001.

[87] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, USA, 1983.

[88] T. Sellis, N. Roussopoulos, and C. Faloutsos. The R^+-tree: A dynamic index for multi-dimensional objects,. In *Proceedings of the 12th Very Large Data Base Conference*, pages 507–518, Brighton, England, 1987.

[89] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 1948.

[90] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, December 2000.

[91] J. R. Smith and S. F. Chang. Automated image retrieval using color and texture. Technical report, Columbia University, 1995.

[92] J. R. Smith and S. F. Chang. Single color extraction and image query. In *Proceedings IEEE International Conference on Image Processing*, pages 528 – 531, Washington, DC, USA, 1995.

[93] J. R. Smith and S. F. Chang. Tools and techniques for color image retrieval. In *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, pages 426 – 437, Philadephia, PA, USA, 1995.

[94] J. R. Smith and S. F. Chang. Automated binary texture feature sets for image retrieval. In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, pages 2239 – 2242, Atlanta, GA, USA, 1996.

[95] J. R. Smith and S. F. Chang. Querying by color regions using the visualseek content-based visual query system. *Intelligent Multimedia Information Retrieval*, pages 23 – 41, 1997.

[96] J. R. Smith and S. F. Chang. Visually searching the web for content. *IEEE Multimedia Magazine*, 4(3):12 – 20, 1997.

[97] M. Stricker and M. Orengo. Similarity of color images. In *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, pages 381–392, Philadephia, PA, USA, 1995.

[98] J. F. Sturm. Using sedumi 1.02: A matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11:625–653, 1999.

[99] Z. Su, H. Zhang, and S. Ma. Using bayesian classifier in relevant feedback of image retrieval. *Journal of Applied Statistics*, 27(3), 2000.

[100] Z. Su, H. Zhang, and S. Ma. Relevance feedback using a bayesian classifier in content-based image retrieval. In *Proceedings of SPIE Storage and Retrieval for Media Databases*, pages 97–106, San Jose, CA, USA, 2001.

[101] M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[102] H. Tamura, S. Mori, and T. Yamawaki. Texture features corresponding to visual perception. *IEEE Transactions on System Man and Cybernetics*, 8(6):460 – 472, 1978.

[103] H. Tamura and N. Yokoya. Image database systems: A survey. *Pattern Recognition*, 17(1), 1984.

[104] G. Taubin and D. B. Cooper. Recognition and positioning of rigid objects using algebraic moment invariants. *Geometric Methods in Computer Vision*, pages 175–186, 1991.

[105] J. Tesic and B. S. Manjunath. Nearest neighbor search for relevance feedback. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 643 – 648, Madison, WI, USA, 2003.

[106] S. Tong. *Active Learning: Theory and Applications*. PhD thesis, Stanford University, 2001.

[107] S. Tong and E. Chang. Support vector machine active learning for image retrieval. In *Proceedings of the 9th ACM International Conference on Multimedia*, pages 107–118, Ottawa, Canada, 2001.

[108] S. Tong and D. Koller. Active learning for parameter estimation in bayesian networks. In *Advances in Neural Information Processing Systems 13*, pages 647–653, Denver, CO, USA, 2000.

[109] S. Tong and D. Koller. Active learning for structure in bayesian networks. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 863–869, Seattle, Washington, USA, 2001.

[110] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2:45–66, March 2002.

[111] K. Tzeras and S. Hartmann. Automatic indexing based on bayesian inference networks. In *Proceedings of 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 22–34, Toronto, Canada, 1993.

[112] V. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag, New York, 1995.

[113] I. Wallace and P. Wintz. An efficient three-dimensional aircraft recognition algorithm using normalized fourier descriptors. *Computer Graphics and Image Processing*, pages 99 – 126, 1980.

[114] J. Wang, W. J. Yang, and R. Acharya. Color clustering techniques for color-content-based image retrieval from image databases. In *Proceedings of IEEE Conference on Multimedia Computing and Systems*, pages 442–449, Ottawa, Ontario, Canada, 1997.

[115] R. Weber, H. J. Schek, and S. Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In *Proceedings of the 24th International Conference on Very Large Data Bases*, pages 194–205, New York City, New York, USA, 1998.

[116] G. Weiss and F. Provost. The effect of class distribution on classifier learning. Technical report, Department of Computer Science, Rutgers University, 2001.

[117] G. Wu. Class-boundary alignment for imbalanced dataset learning. In *Proceedings of ICML Workshop on Learning from Imbalanced Data Sets*, Washington, DC, USA, 2003.

[118] G. Wu and E. Y. Chang. KBA: Kernel boundary alignment considering imbalanced data distribution. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):786–795, June 2005.

[119] P. Wu and B. S. Manjunath. Adaptive nearest neighbor search for relevance feedback in large image databases. In *Proceedings of the 9th ACM International Conference on Multimedia*, pages 89–97, Ottawa, Canada, 2001.

[120] R. Yan, A. Hauptmann, and R. Jin. Negative pseudo-relevance feedback in content-based video retrieval. In *Proceedings of the 11th ACM International Conference on Multimedia*, pages 343 – 346, Berkeley, CA, USA, 2003.

[121] L. Yang and F. Algregtsen. Fast computation of invariant geometric moments: A new method giving correct results. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, pages 201 – 204, Jerusalem, Israel, 1994.

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

[123] C. Zahn and R. Roskies. Fourier descriptors for plane closed curves. *IEEE Transactions on Computers*, 1972.

[124] C. Zhang and T. Chen. An active learning framework for content-based information retrieval. *IEEE Transactions on Multimedia*, 4(2):260–268, June 2002.