Scalable Machine Learning for Visual Data

Xinnan (Felix) Yu

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2015

ABSTRACT

Scalable Machine Learning for Visual Data

Xinnan (Felix) Yu

Recent years have seen a rapid growth of visual data produced by social media, large-scale surveillance cameras, biometrics sensors, and mass media content providers. The unprecedented availability of visual data calls for machine learning methods that are effective and efficient for such large-scale settings.

The input of any machine learning algorithm consists of data and supervision. In a large-scale setting, on the one hand, the data often comes with a large number of samples, each with high dimensionality. On the other hand, the unconstrained visual data requires a large amount of supervision to make machine learning methods effective. However, the supervised information is often limited and expensive to acquire. The above hinder the applicability of machine learning methods for large-scale visual data. In the thesis, we propose innovative approaches to scale up machine learning to address challenges arising from both the **scale of the data** and the **limitation of the supervision**. The methods are developed with a special focus on visual data, yet they are also widely applicable to other domains that require scalable machine learning methods.

Learning with high-dimensionality. The "large-scale" of visual data comes not only from the number of samples but also from the dimensionality of the features. While a considerable amount of effort has been spent on making machine learning scalable for more samples, few approaches are addressing learning with high-dimensional data. In Part I, we propose an innovative solution for learning with very high-dimensional data. Specifically, we use a special structure, the *circulant structure*, to speed up linear projection, the most widely used operation in machine learning. The special structure dramatically improves the space complexity from quadratic to linear, and the computational complexity from

quadratic to linearithmic in terms of the feature dimension. The proposed approach is successfully applied in various frameworks of large-scale visual data analysis, including binary embedding, deep neural networks, and kernel approximation. The significantly improved efficiency is achieved with minimal loss of the performance. For all the applications, we further propose to optimize the projection parameters with training data to further improve the performance.

The scalability of learning algorithms is often fundamentally limited by the amount of supervision available. The massive visual data comes unstructured, with diverse distribution and high-dimensionality — it is required to have a large amount of supervised information for the learning methods to work. Unfortunately, it is difficult, and sometimes even impossible to collect a sufficient amount of high-quality supervision, such as instance-by-instance labels, or frame-by-frame annotations of the videos.

Learning from label proportions. To address the challenge, we need to design algorithms utilizing new types of supervision, often presented in weak forms, such as relatedness between classes, and label statistics over the groups. In Part II, we study a learning setting called Learning from Label Proportions (LLP), where the training data is provided in groups, and only the proportion of each class in each group is known. The task is to learn a model to predict the class labels of the individuals. Besides computer vision, this learning setting has broad applications in social science, marketing, and healthcare, where individual-level labels cannot be obtained due to privacy concerns. We provide theoretical analysis under an intuitive framework called Empirical Proportion Risk Minimization (EPRM), which learns an instance level classifier to match the given label proportions on the training data. The analysis answers the fundamental question, *when and why LLP is possible*. Under EPRM, we propose the proportion-SVM ($\propto$SVM) algorithm, which jointly optimizes the latent instance labels and the classification model in a large-margin framework. The approach avoids making restrictive assumptions on the data, leading to the state-of-the-art results. We have successfully applied the developed tools to challenging problems in computer vision including instance-based event recognition, and attribute modeling.

Scaling up mid-level visual attributes. Besides learning with weak supervision,

the limitation on the supervision can also be alleviated by leveraging the knowledge from different, yet related tasks. Specifically, "visual attributes" have been extensively studied in computer vision. The idea is that the attributes, which can be understood as models trained to recognize visual properties can be leveraged in recognizing novel categories (being able to recognize green and orange is helpful for recognizing apple). In a large-scale setting, the unconstrained visual data requires a high-dimensional attribute space that is sufficiently expressive for the visual world. Ironically, though designed to improve the scalability of visual recognition, conventional attribute modeling requires expensive human efforts for labeling the detailed attributes and is inadequate for designing and learning a large set of attributes. To address such challenges, in Part III, we propose methods that can be used to automatically design a large set of attribute models, without user labeling burdens. We propose *weak attribute*, which combines various types of existing recognition models to form an expressive space for visual recognition and retrieval. In addition, we develop *category-level attribute* to characterize distinct properties separating multiple categories. The attributes are optimized to be discriminative to the visual recognition task over known categories, providing both better efficiency and higher recognition rate over novel categories with a limited number of training samples.

Table of Contents

List of Figures

List of Tables

Glossary

$\propto$SVM – proportion-SVM

ANP – Adjective-Noun Pairs

API – Application Program Interface

AQBC – Angular Quantization-based Binary Codes

AUC – Area Under (ROC) Curve

CBE – Circulant Binary Embedding

CCV – Columbia Consumer Video benchmark

CNM – Compact Nonlinear Map

CNN – Convolutional Neural Network

CPU – Central Processing Unit

DFT – Discrete Fourier Transformation

ECOC – Error-Correcting Output Code

EPRM – Empirical Proportion Risk Minimization

ERM – Empirical Risk Minimization

FFT – Fast Fourier Transformation

FPGA – Field-Programmable Gate Array

GPU – Graphics Processing Unit

IDFT – Inverse DFT

IFFT – Inverse FFT

ITQ – Iterative Quantization

LLP – Learning from Label Proportions

LSH – Locality Sensitive Hashing

MARR – Multi-Attribute based Ranking and Retrieval

MED – Multimedia Event Detection

MIL – Multiple Instance Learning

NDCG – Normalized Discounted Cumulative Gain

NLP – Natural Language Processing

ReLU – Rectified Linear Unit

RFF, RFFM – Random Fourier Feature Map

ROC – Receiver Operating Characteristic

RMLL – Reverse Multi-Label Learning

SH – Spectral Hashing

SKLSH – Shift-invariant Kernel LSH

SVM – Support Vector Machine

TRECVID – Text Retrieval Conference Video Retrieval Evaluation

T-SVM – Transductive SVM

VLAD – Vector of Locally Aggregated Descriptors

Acknowledgments

I would like to express my deep gratitude to my advisor Shih-Fu Chang, who guided me into the vibrating fields of machine learning, computer vision, and multimedia. During the past 5 years, I benefited tremendously from his profound insights in research, rigorous academic attitude, and perseverance in solving challenging problems. The things I learned from Shih-Fu are certainly not limited to the academic aspect. I especially enjoyed his way of summarizing and explaining sophisticated problems and approaches with plain language, precisely to the point. Despite the occasional difference in opinions, Shih-Fu tirelessly developed my skills in research, writing, and communication. I feel greatly fortunate to be his student, and there are just so many more things to learn from Shih-Fu in the years to come.

I spent two memorable summers at the IBM T.J. Watson Research Center in 2012, and 2013. The time at IBM was my first exposure to industrial research. Liangliang Cao always gave me strong encouragement and support when I felt lost. Rogerio Feris always asked the key questions and provided throughout understandings of the related efforts in the computer vision community. I value the countless rides and discussions with Michele Merler. I would like to thank other colleges at IBM who made my internship a rewarding and joyful experience: John R. Smith, Noel Codella, Matt Hill, Bao Nguyen, John Kender, Gang Hua, and Leiguang Gong. Thanks to also the generous financial support from the IBM Ph.D. fellowship award.

A large portion of my thesis work owns to the fruitful collaborations with Sanjiv Kumar, who introduced me to the exciting and emerging topic of large-scale machine learning (or big data with the buzz word). There is always a "field" of curiosity and enthusiasm surrounding him, and I must say that I was highly influenced. Perhaps the most valuable thing I learned from Sanjiv is to keep curious, positive, productive and pushing the boundaries

when working in unknown territories. My summer internship with Sanjiv at Google in 2014 was an exceptional experience, during which I was exposed to both more theoretical and more engineering sides of machine learning. I would like to express my gratitude to other colleagues at Google: thanks to Aditya Bhaskara and Krzysztof Choromanski for their help on the theoretical side on circulant binary embedding and learning from label proportions and to David Simcha and Henry Rowley for developing my skills of writing production quality code.

I would like to thank the committee members Tony Jebara, John Wright and John Paisley for providing constructive suggestions on improving the thesis. I had many discussions with Tony in the last couple of years, and I was constantly thrilled by his deep insights and knowledge in machine learning. The works in learning from label proportions are not possible without his help, and I wish I could have learned and explored more on the ideas we discussed.

Despite continuously trying to learn and explore new areas, I increasingly feel that my knowledge is very limited. I feel truly honored collaborating with so many talented researchers who greatly extended my capability: Subh Bhattacharya, Tao Chen, Yu Cheng, Yunchao Gong, Kuan-Ting Lai, Dong Liu, Ruiqi Guo, Rongrong Ji, Ming-Hen Tsai, Guangnan Ye, and Xu Zhang. None of my thesis works is remotely possible without them. Thanks to also the department staff Elsa Sanchez, former and current DVMM lab members who made my Ph.D. journey smooth and colorful.

Special thanks to my parents Nancy Liu and Haijun Yu, who have been unconditionally supporting my career. Finally, I would like to thank my wife Delia Zhang — I doubt I could have even survived the past five years without her love, understanding, and encouragement.

To Huai Yu

Chapter 1

Introduction

1.1 Motivation

Recent years have seen a rapid growth of all types of visual data produced by social media, large-scale surveillance cameras, biometrics sensors, and mass media content providers. The unprecedented availability of visual data calls for machine learning methods that are effective and efficient for such large-scale settings. The input of any machine learning algorithm consists of data and supervision. We, therefore, motivate the thesis by identifying the challenges in both the data and the supervision.

1.1.1 Large-Scale of Data

It is obvious that the "large-scale" of visual data comes from the number of samples. As of early 2015, 300 hours of videos are uploaded to Youtube every minute[1], billions of photos are uploaded to major social media websites everyday[2]. Even the standard computer vision dataset for benchmarking vision recognition algorithms grew from tens of images to millions of images. Moreover, this is just the beginning – we are experiencing an exponential growth of the visual data.

The "large-scale" of visual data also comes from the dimensionality of the features. The

[1] `https://www.youtube.com/yt/press/statistics.html`

[2] `http://www.kpcb.com/internet-trends`

widely used image representation such as the bag-of-words models and Fisher vectors often consist of tens of thousands of dimensions [183]. The raw image now consists of millions or tens of millions of pixels. In fact, the resolution of image sensors follows the Moore's law: the number of pixels is growing at an exponential rate. While a considerable amount of effort has been devoted to making the machine learning methods scalable to the number of samples [97, 58, 187, 110], few works are addressing making learning methods more applicable for very high-dimensional data. Therefore, with the scalability in the number of samples in mind, one focus of the thesis is to study the scalability in terms of the feature dimension.

1.1.2 Limited Supervision

The scalability of learning algorithms is often fundamentally limited by the amount of supervision available. The massive visual data comes unstructured, with diverse distributions and high-dimensionality — it is required to have a large amount of supervised information to train reliable machine learning models. Unfortunately, it is difficult, and sometimes even impossible to collect sufficient amount of high-quality supervised information, such as instance labels, and detailed annotations on the videos.

One observation is that the massive visual data often comes with some weak forms of supervision, such as the label statistics on the groups. The natural question to ask is whether one can design learning methods to utilize such weak supervision. For example, in recognition of video events, only the event labels on the video level (a group of frames) is given – can we learn a model to pinpoint the frames in which the event actually happens? In modeling attributes, only some semantic similarities between a set of known categories and a set of new attributes are provided – can we leverage such information to model the attributes? Conventional learning methods are not designed to incorporate such forms of supervision.

From another point of view, the limitation of supervision can also be alleviated by leveraging the knowledge learned from different, yet related tasks. In specific, "visual attributes" have been extensively studied in computer vision. The idea is that models trained by other tasks can be leveraged in new tasks (being able to recognize green and orange

should be helpful in recognizing apple). In a large-scale setting, the unconstrained visual data requires a high-dimensional attribute space that is sufficiently expressive for the visual world. Ironically, though designed to improve the scalability of visual recognition, conventional attribute modeling requires expensive human labeling efforts, and it is inadequate for designing and learning a large set of attributes.

1.2 The Thesis Overview

In the thesis, we propose innovative approaches to scale up machine learning considering both the scale of the data and the limitation of the supervision. Part I addresses the problem of learning with high-dimensional data. Part II and Part III address scalable learning with limited supervision. The methods are developed with a special focus on visual data, yet they are also widely applicable to other domains where scalable machine learning methods are required. We provide an overview of the proposed approaches in this section.

1.2.1 Scalable Learning for High-Dimensional Data (Part I)

The first part of the thesis is dedicated to improving the scalability of machine learning on high-dimensional data. We tackle this problem by studying and improving the most widely used operation in machine learning — linear projection. In a large number of application domains in computer vision, data is typically high-dimensional, and the output of linear projection is required to be comparable to the input dimension to preserve the discriminative power of the input space. In such cases, the linear projection becomes a bottleneck: both the computational complexity and the space complexity are $\mathcal{O}(d^2)$, where d is the dimensionality of the data. Such a high cost makes linear projection prohibitive for very high-dimensional data. For example, when applying the method to data with 1 million dimensions, it is required to use terabytes to store the projection matrix, making the method impractical for real-world use. To address this problem, we use the circulant projection to improve the space and computation complexities of linear projections. It operates by imposing a special structure called the circulant structure on the projection matrix. The circulant structure enables the use of Fast Fourier Transformation (FFT) to speed up

the computation. Compared with methods that use unstructured matrices, the proposed method improves the time complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d \log d)$, and the space complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d)$.

We provide an introduction to the problem and the proposed approach in Chapter 2. We apply the circulant projection to three machine learning frameworks widely used in computer vision: binary embedding [234] (Chapter 3), neural networks [37] (Chapter 4), and kernel approximation/ nonlinear map [235] (Chapter 5). For all three learning settings, we study both randomized circulant projections, and optimization methods to improve the quality of the projection. The proposed approaches can dramatically improve the speed and space efficiency, yet without hurting the performance.

1.2.2 Learning from Label Proportions (Part II)

The second part of the thesis addresses the problem of learning with weak supervision. In specific, we consider supervision provided on the group level. To incorporate such information, we study a learning setting called Learning from Label Proportions (LLP), where the training data is provided in groups, and only the proportion of each class in each group is known. The task is to learn a model to predict the class labels of the individuals. This learning setting can be applied in solving various computer vision problems. It is also useful in a broader spectrum of applications of social science, marketing, and healthcare, where individual labels are often confidential due to privacy concerns.

We provide an introduction of the problem and the proposed approach in Chapter 6. To understand when and why LLP is possible, we provide theoretical analysis under an intuitive framework called Empirical Proportion Risk Minimization (EPRM), which learns an instance label classifier to match the given label proportions on the training data [233] (Chapter 7). Under EPRM, we propose the $\propto$SVM algorithm [231] (Chapter 8), which jointly optimizes the latent instance labels and the classification model in a large-margin framework. The approach avoids making restrictive assumptions on the data, leading to state-of-the-art results. We have applied the developed tools to solving challenging problems in computer vision including pinpointing discriminative video shots for video events [118] (Chapter 9) and attribute modeling [232] (Chapter 10).

1.2.3 Scalable Design and Learning of Mid-Level Attributes (Part III)

The third part of the thesis studies improving the scalability of mid-level representation of visual data. The goal of the study is to provide methods that can be used to design and model a large set of useful visual attributes without human efforts. The "usefulness" of the attributes calls for methods that can take the objective of the task such as the quality of retrieval and classification into consideration. The goal of "no human efforts" requires innovative solutions that can leverage data distribution and related tasks.

We provide an introduction to the problem and the proposed approaches in Chapter 11. We first introduce *weak attributes* [229] (Chapter 12), a collection of mid-level representations which can be easily acquired without additional labeling process. It leverages various types of existing recognition tasks to form an expressive visual attribute space. To make the mid-level representations more discriminative to the task, we have further developed the *category-level attributes* [230] (Chapter 13), which can be understood as key properties separating multiple categories, *e.g.*, unique attributes shared by "whale" and "sea lion" in contrast from "dog" and "cat". The proposed framework consists of attribute-encoding and category-decoding steps for recognition, generalizing the classic Error Correcting Output Code (ECOC). The category-level attributes provide both efficiency and performance improvements in recognition with few or zero examples.

Part I

Scalable Learning for High-dimensional Data

Chapter 2

Fast Linear Projections with Circulant Matrices

2.1 Introduction

In numbers of application domains, especially computer vision, data is typically very high-dimensional. For example, the raw images often consist of millions or tens of millions of pixels, and the widely used image representation such as the bag-of-words models, and the Fisher vectors often consist of tens of thousands of dimensions [183]. While a considerable amount of efforts have been devoted to making machine learning method scalable to the number of samples [97, 58, 187, 110], relatively fewer works have been proposed to make learning methods more applicable for very high-dimensional data. The first part of the thesis studies improving the scalability of machine learning for high-dimensional data. In specific, our goal is to improve the space and computational complexities of linear projection, the most widely used operation, and often the bottleneck, in machine learning.

Given a vector $\mathbf{x} \in \mathbb{R}^d$, and a projection matrix $\mathbf{R} \in \mathbb{R}^{k \times d}$, the linear projection computes $h(\mathbf{x}) \in \mathbb{R}^k$:

$$h(\mathbf{x}) = \mathbf{R}\mathbf{x}. \tag{2.1}$$

The linear projections play a key role in learning visual representations. For example, in the task of *dimensionality reduction* (2.1) maps a d-dimensional vector to a k-dimensional

vector ($k < d$). The projection matrix can either be randomized (where it can be shown that the distance can be approximately preserved in the mapped space) [99], or optimized based on certain objectives on the training data. Some examples are the Linear Discriminative Analysis (LDA) [64, 173], and metric learning [220].

As another example, in the task of *binary embedding*, the real-valued data is transformed into binary code for faster processing. A widely used paradigm, linear projection based binary embedding, computes

$$h(\mathbf{x}) = \text{sign}(\mathbf{R}\mathbf{x}). \tag{2.2}$$

The elements of the projection matrix $\mathbf{R}$ can be randomly generated from a probability distribution, in which case the resulting binary code can preserve the angle of the original space [32]. The projection matrix can also be optimized in terms of certain objectives, such as distance preservation or low reconstruction error [112, 152, 218, 153, 72, 216].

The third example is *deep neural networks*, which recently received renewed attention in the computer vision community. In the neural network architecture, a fully connected layer can be seen as performing an operation

$$h(\mathbf{x}) = \phi(\mathbf{R}\mathbf{x}), \tag{2.3}$$

where $\phi(\cdot)$ is a nonlinear activation function, such as the sigmoid or rectified linear unit (ReLU). The projection matrix $\mathbf{R}$ is optimized in terms of certain objective function on the training data, such as the multi-class recognition rate. The fully connected layer captures global information of the image, and it is important in neural network architectures for visual recognition [110].

Another example comes from *kernel approximation*. One widely used method, Random Fourier Features [171], maps the input feature $\mathbf{x}$ by

$$h(\mathbf{x}) = \cos(\mathbf{R}\mathbf{x}), \tag{2.4}$$

where elements of the projection matrix $\mathbf{R}$ is generated *iid* from a probability distribution whose probability density function corresponds to the Fourier transformation of a kernel function. It can be shown that such a nonlinear map can be used to approximate a positive-definite shift-invariance kernel, such as the Gaussian kernel.

In a large number of application domains, data is typically high-dimensional, and the required output is also required to be high-dimensional to preserve the discriminative power of the input space. In fact, the required dimensionality k often needs to be $\mathcal{O}(d)$, where d is the input dimensionality. In such case, both the computational and the space complexities are $\mathcal{O}(d^2)$. This makes the linear projection operation very expensive in both space and computation. For example, when applying the method to data with 1 million dimensions, it requires terabytes to store the projection matrix, making the method impractical for real-world use.

To address this problem, we use the circulant projection to improve the space and computation complexities of linear projections. It operates by imposing a special structure called the circulant structure on the projection matrix. The circulant structure enables the use of Fast Fourier Transformation (FFT) to speed up the computation. Compared with methods that use unstructured matrices, the proposed method improves the time complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d \log d)$, and the space complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d)$ where d is the input dimensionality.

In the thesis, we apply the circulant structure to applications including the aforementioned binary embedding, deep neural networks, and kernel approximation. In all the application scenarios, we show that the randomized circulant project leads to competitive performance compared with the unstructured fully randomized projections, yet with dramatically improved space and computation cost. To further improve the performance, we also propose to optimize the circulant projection based on the training data.

2.2 Related Works

Using structured matrices to speed up linear projections, especially for dimensionality reduction, is not a new topic. We begin by reviewing the related works in this section.

2.2.1 Randomized Matrices for Dimensionality Reduction

The celebrated Johnson-Lindenstrauss lemma states that [99]:

Lemma 2.1 (Johnson-Lindenstrauss)**.** *Let* $\mathbf{x}_1, ..., \mathbf{x}_N \in \mathbb{R}^d$ *be* N *points. For a given* $\epsilon \in$

$(0, 1/2)$ and a natural number $k = \mathcal{O}(\epsilon^{-2} \log N)$, there exists a linear map $f : \mathbb{R}^d \to \mathbb{R}^k$, such that

$$(1 - \epsilon)||\mathbf{x}_i||_2^2 \leq ||f(\mathbf{x}_i)||_2^2 \leq (1 + \epsilon)||\mathbf{x}_i||_2^2,$$

for any $i \in \{1, ..., N\}$.

The Johnson-Lindenstruss lemma is a remarkable result as it states that the dimensionality for any N samples can be dramatically reduced to $\mathcal{O}(\log N/\epsilon^2)$, while approximately preserving the pair-wise ℓ_2 distances within a factor of $(1 \pm \epsilon)$.

The proof of the Johnson-Lindenstruss lemma (see, for example, [99, 45]) is by using randomized matrices generated from certain distribution as the linear map $f : \mathbb{R}^d \to \mathbb{R}^k$. It then shows that with a certain probability the Johnson-Lindenstruss lemma is true with standard concentration inequalities. For example, the elements of the projection matrix can be generated *iid* from a standard Gaussian distribution. Due to the simplicity and theoretical support, random projection based dimensionality reduction has been applied in broad applications including approximate nearest neighbor research [89], dimensionality reduction in databases [1], and bi-Lipschitz embeddings of graphs into normed spaces [65].

When using an unstructured randomized matrix to compute the random projection, both the space and computational complexities are $\mathcal{O}(kd)$, making the method prohibitively expensive for very high-dimensional datasets. Therefore, approaches have been proposed to show the Johnson-Lindenstruss-type results with structured randomized matrices, including Hadamard matrices with a sparse random Gaussian matrix [3], sparse matrices [142], and Lean Walsh Transformations [131]. The advantage of using a structured matrix is that the space and computation cost can be dramatically reduced, yet the distance preserving property remains to be competitive.

2.2.2 Circulant Projection for Dimensionality Reduction

Recently, the randomized circulant matrices were used to achieve the Johnson-Lindenstruss results. The projection comprises of a randomized circulant matrix and a random sign flipping operation on the input features. For d-dimensional input, and k-dimensionally output ($k < d$), the method has computation complexity $\mathcal{O}(d \log d)$ and space complexity $\mathcal{O}(d)$.

Showing the distance preserving property for the randomized circulant projections is not a trivial task. Since the projections are highly dependent, the classic concentration inequalities does not hold. The initial result suggested an embedding dimension $\mathcal{O}(\epsilon^{-2}\log^3 N)$ (compared with $\mathcal{O}(\epsilon^{-2}\log N)$ in the original Johnson-Lindenstauss) [82], with the decoupling lemma. This was subsequently improved to $\mathcal{O}(\epsilon^{-2}\log^2 N)$ [212] by considering the projection in the Fourier domain. Recently, the result was further improved to $\mathcal{O}(\epsilon^{-2}\log^{(1+\delta)} N)$ [239] using matrix-valued Bernstein inequalities.

2.2.3 Randomized Structured Matrices in Other Applications

There have been methods proposed to apply structured randomized matrices in locality sensitive hashing for approximate nearest neighbor search [44], and kernel approximation [123]. Different from the above works, we study using the circulant matrices in applications including binary embedding, neural network, and kernel approximation. Although circulant matrices have been demonstrated to achieve satisfactory properties for dimensionality reduction, whether such structure is helpful in other applications has not been previously studied. More importantly, there are two advantages of the circulant projection compared with other choices. First, both the space and computational complexities are superior to the alternatives [44, 123]. Second, the circulant projection is equivalent to an element-wise multiplication in the Fourier domain. This makes optimizing the parameters simple and efficient. As shown in our experiments, very efficient optimization procedures exist for all the three applications, and the optimization can significantly improve the performance.

2.3 Fast Linear Projections with Circulant Matrices

We present the framework of using circulant matrices to perform linear projections in this section. We first show how to compute circulant projection when the input and the output dimension are the same, *i.e.*, $k = d$ (Section 2.3). We then show how to deal with the situation where $k \neq d$ (Section 2.3.2). Table 2.2.3 summarizes the key notations of Part I.

$\mathbf{r}$	A column vector
r_i	The i-th element of $\mathbf{r}$
$\mathbf{R}$	A matrix
$R_{i,j}$	Element of i-th row, and j-th column of $\mathbf{R}$
$\mathbf{R}^T$, $\mathbf{r}^T$	Transpose of real-valued matrix or vector
$\mathbf{R}^H$, $\mathbf{r}^H, \bar{r}$	Conjugate transpose of $\mathbf{R}$, $\mathbf{r}$, r, respectively
$\mathbf{R}_{i\cdot}$	i-th row vector of $\mathbf{R}$
$\mathbf{R}_{\cdot j}$	j-th column vector of $\mathbf{R}$
$\text{circ}(\mathbf{r})$	The circulant matrix formed by $\mathbf{r}$
$\text{Tr}(\cdot)$	Trace of a matrix
$\|\|\cdot\|\|_F$	Frobenius norm of a matrix
$\|\|\cdot\|\|_p$	ℓ_p Norm
$\circ$	Hadamard product
$\circledast$	Circular convolution
$\mathcal{F}(\cdot)$	Discrete Fourier Transform (DFT)
$\mathcal{F}^{-1}(\cdot)$	Inverse DFT (IDFT)
$\Re(\cdot)$	The real part of a scalar, vector, or matrix
$\Im(\cdot)$	The imaginary part of a scalar, vector, or matrix
$\text{diag}(\mathbf{r})$	Diagonal matrix with $\mathbf{r}$ as the diagonal vector
d	Dimensionality of the feature
k	Dimensionality after the projection
N	Number of samples in training
$\text{sign}(\cdot)$	Element-wise binarization
$\mathbb{P}(\cdot)$	Probability of an event
$\mathbb{E}(\cdot)$	Expectation
$var(\cdot)$	Variance
$s_{\rightarrow j}(\mathbf{r})$	Vector formed by downwards circularly shifting $\mathbf{r}$ by j elements
$\text{span}(\mathbf{x}, \mathbf{y})$	The span of $\mathbf{x}$ and $\mathbf{y}$

Table 2.1: Key Notations of Part I.

2.3.1 The Framework

A circulant matrix $\mathbf{R} \in \mathbb{R}^{d \times d}$ is a matrix defined by a vector $\mathbf{r} = (r_0, r_1, \cdots, r_{d-1})^T$ [78].

$$\mathbf{R} = \text{circ}(\mathbf{r}) := \begin{bmatrix} r_0 & r_{d-1} & \cdots & r_2 & r_1 \\ r_1 & r_0 & r_{d-1} & & r_2 \\ \vdots & r_1 & r_0 & \ddots & \vdots \\ r_{d-2} & & \ddots & \ddots & r_{d-1} \\ r_{d-1} & r_{d-2} & \cdots & r_1 & r_0 \end{bmatrix}. \tag{2.5}$$

Let $\mathbf{D}$ be a diagonal matrix with each diagonal entry being a Rademacher variable (± 1 with probability 1/2). For $\mathbf{x} \in \mathbb{R}^d$, its d-dimensional circulant projection with $\mathbf{r} \in \mathbb{R}^d$ is defined as:

$$h(\mathbf{x}) = \mathbf{RDx}, \tag{2.6}$$

where $\mathbf{R} = \text{circ}(\mathbf{r})$.

The above framework follows from using circulant matrices in dimensionality reduction [212, 82]. The sign flipping matrix $\mathbf{D}$ is required to show the Johnson-Lindenstrauss type results. In other words, $\mathbf{D}$ is helpful in order for the circulant projection to "simulate" a fully randomized projection. Empirically, we found that omitting such an operation will lead to inferior performance in all applications and most datasets. Note that applying $\mathbf{D}$ to $\mathbf{x}$ is equivalent to applying random sign flipping to each dimension of $\mathbf{x}$. Since sign flipping can be carried out as a preprocessing step for each input $\mathbf{x}$, here onwards for simplicity in presenting the algorithms, we will drop explicit mention of $\mathbf{D}$. Hence the circulant projection is given as $h(\mathbf{x}) = \mathbf{Rx}$.

The main advantage of circulant binary embedding is its ability to use Fast Fourier Transformation (FFT) to speed up the computation.

Proposition 2.1. *For d-dimensional data, circulant projection has space complexity $\mathcal{O}(d)$, and time complexity $\mathcal{O}(d \log d)$.*

Since a circulant matrix is defined by a single column/row, clearly the storage needed is $\mathcal{O}(d)$. Given a data point $\mathbf{x}$, the d-dimensional circulant projection can be efficiently

computed as follows. Denote $\circledast$ as operator of the circulant convolution. Based on the definition of circulant matrix,

$$\mathbf{R}\mathbf{x} = \mathbf{r} \circledast \mathbf{x}. \tag{2.7}$$

The above can be computed based on Discrete Fourier Transformation (DFT), for which fast algorithm (FFT) is available.

The DFT of a vector $\mathbf{t} \in \mathbb{C}^d$ is a d-dimensional vector with each element defined as

$$\mathcal{F}(\mathbf{t})_l = \sum_{m=0}^{d-1} t_n \cdot e^{-i2\pi lm/d}, l = 0, \cdots, d-1. \tag{2.8}$$

The above can be expressed equivalently in a matrix form as

$$\mathcal{F}(\mathbf{t}) = \mathbf{F}_d \mathbf{t}, \tag{2.9}$$

where $\mathbf{F}_d$ is the d-dimensional DFT matrix. Let $\mathbf{F}_d^H$ be the conjugate transpose of $\mathbf{F}_d$. It is easy to show that $\mathbf{F}_d^{-1} = (1/d)\mathbf{F}_d^H$. Similarly, for any $\mathbf{t} \in \mathbb{C}^d$, the Inverse Discrete Fourier Transformation (IDFT) is defined as

$$\mathcal{F}^{-1}(\mathbf{t}) = (1/d)\mathbf{F}_d^H \mathbf{t}. \tag{2.10}$$

The introduced matrix notation is useful in the Circulant Binary Embedding (CBE) (Chapter 3).

Since the convolution of two signals in their original domain is equivalent to the Hadamard product in their frequency domain [154],

$$\mathcal{F}(\mathbf{R}\mathbf{x}) = \mathcal{F}(\mathbf{r}) \circ \mathcal{F}(\mathbf{x}). \tag{2.11}$$

Therefore,

$$h(\mathbf{x}) = \mathcal{F}^{-1}(\mathcal{F}(\mathbf{r}) \circ \mathcal{F}(\mathbf{x})). \tag{2.12}$$

As both DFT and IDFT can be efficiently computed in $\mathcal{O}(d \log d)$ with FFT [154], circulant projection has time complexity $\mathcal{O}(d \log d)$.

2.3.2 When $k \neq d$

We have proposed the framework of circulant projection when the output dimensionality k is equivalent to the input dimensionality d. Such a setting is commonly used in all the applications considered in the thesis including binary embedding, neural network, and kernel approximation. In this section, we provide extensions to handle the case where $k \neq d$.

When $k < d$, a "compression", or dimensionality reduction is performed. In such a case, we still use the circulant matrix $\mathbf{R} \in \mathbb{R}^{d \times d}$ with d parameters, and the output is simply set to be the first k elements in (2.6). One may notice that the circulant projection is not computationally more efficient in this situation compared with $k = d$. But when k is larger than $O(\log d)$, circulant projection still has better computational complexity ($\mathcal{O}(d \log d)$ *vs.* $\mathcal{O}(dk)$) and space complexity ($\mathcal{O}(d)$ *vs.* $\mathcal{O}(dk)$).

When $k > d$, an "expansion" is performed. In such a case, the simplest solution is to use multiple circulant projections, and concatenate the output of them. This gives the computational complexity $\mathcal{O}(k \log d)$, and space complexity $\mathcal{O}(k)$. Note that the DFT of the feature vector can be reused in this case. An alternative approach is to extend every feature vector to a k-dimensional vector, by padding $(k - d)$ zeros at the end. Then the problem becomes the conventional setting described in Section 2.3. This gives space complexity $\mathcal{O}(k)$, and computational complexity $\mathcal{O}(k \log k)$. In practice, k is usually at most a few times larger than d. Empirically the two approaches give similar computational time and performance in all the applications.

2.4 Overview of the Proposed Approaches

We have applied the circulant projection to three applications: binary embedding for high-dimensional data [234] (Chapter 3), fully connected layers in neural networks [37] (Chapter 4), and kernel approximation/ nonlinear map for high-dimensional data [235] (Chapter 5). For all the applications, we show that the randomized circulant matrices lead to faster computation and much lower memory cost without hurting the performance compared with randomized unstructured projections. We also propose to optimize the parameters of the circulant matrices based on the training data, leading to significantly

improved performance, *i.e.*, better recall using binary embedding, better recognition using neural network and nonlinear maps.

2.4.1 Circulant Binary Embedding

Binary embedding of high-dimensional data requires long codes to preserve the discriminative power of the input space. Traditional binary coding methods often suffer from very high computation and storage costs in such a scenario. To address this problem, we propose Circulant Binary Embedding (CBE) which generates binary codes by projecting the data with a circulant matrix. The circulant structure enables the use of Fast Fourier Transformation to speed up the computation. Compared with methods that use unstructured matrices, the proposed method improves the time complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d \log d)$, and the space complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d)$ where d is the input dimensionality. We first analyze the angle preserving properties of CBE with the randomized projection matrix. We then propose a novel time-frequency alternating optimization to learn data-dependent circulant projections, which alternatively minimizes the objective in original and Fourier domains. We show by extensive experiments that the proposed approach gives much better performance than the state-of-the-art approaches for fixed time and provides much faster computation with no performance degradation for fixed number of bits. The work was originally presented in [234].

2.4.2 Circulant Neural Networks

The basic computation of a fully-connected neural network layer is a linear projection of the input signal followed by a non-linear transformation. The linear projection step consumes the bulk of the processing time and memory footprint. In this work, we propose to replace the conventional linear projection with the circulant projection. The circulant structure enables the use of the Fast Fourier Transform to speed up the computation. Considering a neural network layer with d input nodes, and d output nodes, this method improves the time complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d \log d)$ and space complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d)$. We further show that the gradient computation and optimization of the circulant projections can be performed very efficiently. Our experiments on three standard datasets

show that the proposed approach achieves this significant gain in efficiency and storage with minimal loss of accuracy compared with neural networks with unstructured projections. The work was originally presented in [37].

2.4.3 Compact Nonlinear Map with Circulant Extensions

Kernel approximation via nonlinear random feature maps is widely used in speeding up kernel machines. There are two main challenges for the conventional kernel approximation methods. First, before performing kernel approximation, a good kernel has to be chosen. Picking a good kernel is a very challenging problem in itself. Second, high-dimensional maps are often required in order to achieve good performance. This leads to high computational cost in both generating the nonlinear maps, and in the subsequent learning and prediction process. In this work, we propose to optimize the nonlinear maps directly with respect to the classification objective in a data-dependent fashion. This achieves kernel approximation and kernel learning in a joint framework, leading to much more compact maps without hurting the performance. As a by-product, the same framework can also be used to achieve more compact kernel maps to approximate a known kernel. Under the above CNM framework, we introduce Circulant Nonlinear Maps, which uses a circulant-structured projection matrix to speed up the nonlinear maps for high-dimensional data. This leads to better computational complexity and space complexity without hurting the performance. The work was originally presented in [235].

Chapter 3

Circulant Binary Embedding

3.1 Introduction

Embedding input data in binary spaces is becoming popular for efficient retrieval and learning on massive data sets [127, 73, 170, 71, 134]. Moreover, in a large number of application domains such as computer vision, biology, and finance, data is typically high-dimensional. When representing such high dimensional data by binary codes, it has been shown that long codes are required in order to achieve good performance. In fact, the required number of bits is $\mathcal{O}(d)$, where d is the input dimensionality [127, 73, 183]. The goal of binary embedding is to well approximate the input distance as Hamming distance so that efficient learning and retrieval can happen directly in the binary space. It is important to note that another related area called *hashing* is a special case with slightly different goal: creating hash tables such that points that are similar fall in the same (or nearby) bucket with high probability. In fact, even in hashing, if high accuracy is desired, one typically needs to use hundreds of hash tables involving tens of thousands of bits.

Most of the existing linear binary coding approaches generate the binary code by applying a projection matrix, followed by a binarization step. Formally, given a data point, $\mathbf{x} \in \mathbb{R}^d$, the k-bit binary code, $h(\mathbf{x}) \in \{+1, -1\}^k$ is generated as

$$h(\mathbf{x}) = \text{sign}(\mathbf{R}\mathbf{x}), \tag{3.1}$$

where $\mathbf{R} \in \mathbb{R}^{k \times d}$, and sign$(\cdot)$ is a binary map which returns element-wise sign[1]. Several techniques have been proposed to generate the projection matrix randomly without taking into account the input data [32, 170]. These methods are very popular due to their simplicity but often fail to give the best performance due to their inability to adapt the codes with respect to the input data. Thus, a number of data-dependent techniques have been proposed with different optimization criteria such as reconstruction error [112], data dissimilarity [152, 218], ranking loss [153], quantization error after PCA [72], and pairwise misclassification [216]. These methods are shown to be effective for learning compact codes for relatively low-dimensional data. However, the $\mathcal{O}(d^2)$ computational and space costs prohibit them from being applied to learning long codes for high-dimensional data. For instance, to generate $\mathcal{O}(d)$-bit binary codes for data with $d \sim$1M, a huge projection matrix will be required needing TBs of memory, which is not practical[2].

In order to overcome these computational challenges, [73] proposed a bilinear projection based coding method for high-dimensional data. It reshapes the input vector $\mathbf{x}$ into a matrix $\mathbf{Z}$, and applies a bilinear projection to get the binary code:

$$h(\mathbf{x}) = \text{sign}(\mathbf{R}_1^T \mathbf{Z} \mathbf{R}_2). \tag{3.2}$$

When the shapes of $\mathbf{Z}, \mathbf{R}_1, \mathbf{R}_2$ are chosen appropriately, the method has time and space complexity of $\mathcal{O}(d^{1.5})$ and $\mathcal{O}(d)$, respectively. Bilinear codes make it feasible to work with datasets with very high dimensionality and have shown good results in a variety of tasks.

In this work, we propose a novel Circulant Binary Embedding (CBE) technique which is even faster than the bilinear coding. It is achieved by imposing a circulant structure on the projection matrix $\mathbf{R}$ in (3.1). This special structure allows us to use Fast Fourier Transformation (FFT) based techniques, which have been extensively used in signal processing. The proposed method further reduces the time complexity to $\mathcal{O}(d \log d)$, enabling efficient binary embedding for very high-dimensional data[3]. Table 3.1 compares the time and space complexity for different methods. This work makes the following contributions:

[1] A few methods transform the linear projection via a nonlinear map before taking the sign [218, 170].

[2] In principle, one can generate the random entries of the matrix on-the-fly (with fixed seeds) without needing to store the matrix. But this will increase the computational time even further.

[3] One could in principal use other structured matrices like Hadamard matrix along with a sparse random

Method	Time	Space	Time (Learning)
Full projection	$\mathcal{O}(d^2)$	$\mathcal{O}(d^2)$	$\mathcal{O}(Nd^3)$
Bilinear projection	$\mathcal{O}(d^{1.5})$	$\mathcal{O}(d)$	$\mathcal{O}(Nd^{1.5})$
Circulant projection	$\mathcal{O}(d \log d)$	$\mathcal{O}(d)$	$\mathcal{O}(Nd \log d)$

Table 3.1: Comparison of the proposed method (Circulant projection) with other methods for generating long codes (code dimension D comparable to input dimension d). N is the number of instances used for learning data-dependent projection matrices.

- We propose the circulant binary embedding method, which has space complexity $\mathcal{O}(d)$ and time complexity $\mathcal{O}(d \log d)$ (Section 3.2, 3.3).
- We analyze the angle preserving properties of CBE with randomized circulant matrix. We show that the quality of randomized CBE is almost identical to LSH with mild assumptions.
- We propose to learn the data-dependent circulant projection matrix by a novel and efficient time-frequency alternating optimization, which alternatively optimizes the objective in the original and frequency domains (Section 3.4).
- Extensive experiments show that, compared with the state-of-the-art, the proposed method improves the result dramatically for a fixed time cost, and provides much faster computation with no performance degradation for a fixed number of bits (Section 3.5).

3.2 Circulant Binary Embedding

For $\mathbf{x} \in \mathbb{R}^d$, its d-dimensional Circulant Binary Embedding with $\mathbf{r} \in \mathbb{R}^d$ is defined as:

$$h(\mathbf{x}) = \text{sign}(\mathbf{RDx}), \tag{3.3}$$

where $\mathbf{R} = \text{circ}(\mathbf{r})$, and $\mathbf{D}$ is the random sign flipping matrix as in (2.6). When $k < d$, we simply use the first k bits as the result. Following the analysis in Section 2.3, CBE has computational complexity $\mathcal{O}(d \log d)$ and space complexity $\mathcal{O}(d)$.

Gaussian matrix to achieve fast projection as was done in fast Johnson-Lindenstrauss transform[3, 44], but it is still slower than circulant projection and needs more space.

3.3 Randomized Circulant Binary Embedding

A simple way to obtain CBE is by generating the elements of $\mathbf{r}$ in (2.6) independently from the standard normal distribution $\mathcal{N}(0,1)$. We call this method randomized CBE (CBE-rand). In this section, we show that the angle preserving property of randomized CBE is almost as good as Locality Sensitive Hashing (LSH) under mild assumptions for high-dimensional data.

The analysis is based on two fixed data points, $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$. As $\text{sign}(\mathbf{r}^T\mathbf{x}) = \text{sign}(\mathbf{r}^T\mathbf{x}/||\mathbf{x}||)$, without loss of generality, we assume $||\mathbf{x}|| = 1$, $||\mathbf{y}|| = 1$. Let the angle between $\mathbf{x}$ and $\mathbf{y}$ be θ. Given a projection matrix $\mathbf{R}$, and a random sign flipping matrix $\mathbf{D}$,

$$X_i = \frac{1 - \text{sign}(\mathbf{R}_{i\cdot}\mathbf{D}\mathbf{x})\,\text{sign}(\mathbf{R}_{i\cdot}\mathbf{D}\mathbf{y})}{2} - \frac{\theta}{\pi}. \tag{3.4}$$

Therefore $\frac{1}{k}\sum_{i=1}^k X_i + \frac{\theta}{\pi}$ is the averaged hamming distance for k-bit code. We are interested in the expectation and the variance of $\frac{1}{k}\sum_{i=1}^k X_i$.

3.3.1 Properties of Locality Sensitive Hashing (LSH).

We begin by reviewing the properties of the LSH (simhash) [32]. In LSH, all the elements of the projection matrix $\mathbf{R} \in \mathbb{R}^{k\times d}$ are drawn *iid* from $\mathcal{N}(0,1)$. For all $i = 0, \cdots, k-1$, we have[4]

$$\mathbb{P}(\text{sign}(\mathbf{R}_{i\cdot}\mathbf{D}\mathbf{x}) = \text{sign}(\mathbf{R}_{i\cdot}\mathbf{D}\mathbf{y})) = 1 - \frac{\theta}{\pi}, \quad \mathbb{P}(\text{sign}(\mathbf{R}_{i\cdot}\mathbf{D}\mathbf{x}) \neq \text{sign}(\mathbf{R}_{i\cdot}\mathbf{D}\mathbf{y})) = \frac{\theta}{\pi}. \tag{3.5}$$

Therefore

$$\mathbb{E}\left[\frac{1}{k}\sum_{i=1}^k X_i\right] = 0. \tag{3.6}$$

In other words, the expectation of the average hamming distance is θ/π. As the averaged hamming distance is defined as the mean over k independent variables, the variance can be obtained based on that of the binomial distribution:

$$var\left[\frac{1}{k}\sum_{i=1}^k X_i\right] = \frac{1}{k}\frac{\theta}{\pi}\left(1 - \frac{\theta}{\pi}\right). \tag{3.7}$$

[4]Note that the matrix $\mathbf{D}$ does not influence the properties of the projection, as $\mathbf{RD}$ is also a matrix with every elements drawn *iid* from $\mathcal{N}(0,1)$. We put $\mathbf{D}$ here to make analysis of CBE more consistent.

The above shows that longer code will lead to better estimation of the angle. Note that the variance is a function of k, the number of bits. It is independent on d, the feature dimension. This makes one wonder why long code is required for high-dimensional data. The reason is that there is the well-known "curse of dimensionality" for real-world data: in high-dimensional space, all the angles of the data points tend to be close to $\pi/2$. In order to discriminate the subtle difference of the angles, lower angle estimation error, therefore, longer code, is required.

3.3.2 Properties of Randomized CBE

For randomized CBE, the elements of the first column of $\mathbf{R}$ is drawn *iid.* from $\mathcal{N}(0,1)$. First, we also have

$$\mathbb{E}\left[\frac{1}{k}\sum_{i=1}^{k} X_i\right] = 0. \tag{3.8}$$

This is based on the fact that $\mathbb{E}X_i = 0$, for $i = 0, \cdots, k-1$. Following the analysis of LSH, we would hope that $var(\frac{1}{k}\sum_{i=1}^{k} X_i)$ decreases as a function of k. Unfortunately, the variance is no longer a straightforward computation, as the codes are dependent. We show in Theorem 3.1 that, under mild assumptions, the variance is almost identical to that of LSH for high-dimensional data.

Theorem 3.1. *For any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, such that $\|\mathbf{x}\| = 1$, $\|\mathbf{y}\| = 1$, $\max\{\|\mathbf{x}\|_\infty, \|\mathbf{y}\|_\infty\} \leq \rho$, for some parameter $\rho < 1$. Then with probability at least $1 - \delta$ over the choice of $\mathbf{D}$, we have*

$$var\left[\frac{1}{k}\sum_{i=0}^{k-1} X_i\right] \leq \frac{1}{k}\frac{\theta}{\pi}\left(1 - \frac{\theta}{\pi}\right) + 8\rho\sqrt{\ln\frac{4k^2}{\delta}}.$$

The variance above is over the choice of $\mathbf{r}$.

Remarks. The small infinity norm constraint means that the data should be sufficiently "spread-out". A stronger constraint will lead to a stronger result. For example let $\rho = \mathcal{O}(\frac{\log d}{\sqrt{d}})$, then $var\left[\frac{1}{k}\sum_{i=0}^{k-1} X_i\right] = \frac{1}{k}\frac{\theta}{\pi}\left(1 - \frac{\theta}{\pi}\right) + \mathcal{O}\left(\frac{\log^{1.5} d}{\sqrt{d}}\right)$.

Proof. To show the theorem:

$$\begin{aligned} & var\left[\frac{1}{k}\sum_{i=0}^{k-1}X_i\right] = \mathbb{E}\left[\frac{1}{k^2}\left(\sum_{i=0}^{k-1}X_i\right)^2\right] - \mathbb{E}\left[\frac{1}{k}\sum_{i=0}^{k-1}X_i\right]^2 \\ = & \mathbb{E}\left[\frac{1}{k^2}\left(\sum_{i=0}^{k-1}X_i\right)^2\right] = \mathbb{E}\left[\frac{\sum_{i=0}^{k-1}X_i^2 + \sum_{i\neq j}X_iX_j}{k^2}\right] \\ = & \frac{1}{k^2}\left(k\mathbb{E}X_1^2 + \sum_{i\neq j}\mathbb{E}(X_iX_j)\right) = \frac{1}{k}\frac{\theta}{\pi}\left(1-\frac{\theta}{\pi}\right) + \frac{1}{k^2}\sum_{i\neq j}\mathbb{E}(X_iX_j). \end{aligned}$$

Thus it is sufficient to show that $\mathbb{E}(X_iX_j) \leq 8\rho\sqrt{\ln\frac{4k^2}{\delta}}$, for all $i \neq j$, with probability at least $1-\delta$. Note that a more careful consideration of all the terms may lead to a tighter bound (since there are both positive and negative terms). We leave this for our future works.

Without loss of generality, we assume $i = 0$. With a slight abuse of notation, we assume that the first row (instead of the column in Section 2.3) of the circulant projection matrix $\mathbf{R}$ is $\mathbf{r}^T$. Let $s_{\rightarrow j}(\mathbf{r})$ be the vector formed by downwards circular shifting $\mathbf{r}$ by j element, and let $t = d - i$. Note that $s_{\rightarrow j}(\mathbf{r})^T\mathbf{x} = \mathbf{r}^T s_{\rightarrow(d-j)}(\mathbf{x})$. We have,

$$\begin{aligned} & \mathbb{E}(X_0X_j) \qquad (3.9) \\ = & \mathbb{E}\left[\left(\frac{1-\text{sign}(\mathbf{r}^T\mathbf{D}\mathbf{x})\,\text{sign}(\mathbf{r}^T\mathbf{D}\mathbf{y})}{2} - \frac{\theta}{\pi}\right)\left(\frac{1-\text{sign}(\mathbf{r}^T s_{\rightarrow t}(\mathbf{D}\mathbf{x}))\,\text{sign}(\mathbf{r}^T s_{\rightarrow t}(\mathbf{D}\mathbf{y}))}{2} - \frac{\theta}{\pi}\right)\right]. \end{aligned}$$

For a moment, suppose $s_{\rightarrow t}(\mathbf{D}\mathbf{x})$ and $s_{\rightarrow t}(\mathbf{D}\mathbf{y})$ are orthogonal to the span of $\mathbf{D}\mathbf{x}$ and $\mathbf{D}\mathbf{y}$, the above is 0, based on the following lemma.

Lemma 3.1. *Let $\mathbf{a}, \mathbf{b}, \mathbf{u}, \mathbf{v}$ be unit vectors in $\mathbb{R}^d$ such that $\mathbf{u}$ and $\mathbf{v}$ are orthogonal to the subspace spanned by $\mathbf{a}$ and $\mathbf{b}$. Let $\mathbf{r}$ be a random Gaussian vector. Then we have*

$$\mathbb{E}\left[\left(\frac{1-\text{sign}(\mathbf{r}^T\mathbf{a})\,\text{sign}(\mathbf{r}^T\mathbf{b})}{2} - \frac{\angle(\mathbf{a},\mathbf{b})}{\pi}\right)\left(\frac{1-\text{sign}(\mathbf{r}^T\mathbf{u})\,\text{sign}(\mathbf{r}^T\mathbf{v})}{2} - \frac{\angle(\mathbf{u},\mathbf{v})}{\pi}\right)\right] = 0.$$

Proof. The two dimensional subspace spanned by $\mathbf{u}$ and $\mathbf{v}$ is orthogonal to the two dimensional subspace spanned by $\mathbf{a}$ and $\mathbf{b}$. Then the hyperplanes defined by $\mathbf{r}$ in the two subspaces are independent. Therefore the two terms in the expectation are independent. The expectation of the multiplication of them is equal to the multiplication of the expectations of each of them, which is 0. □

The following lemma extends Lemma 3.1, by showing that with near orthogonality, the expectation is close to 0.

Lemma 3.2. *Let* $\mathbf{a}, \mathbf{b}, \mathbf{u}, \mathbf{v}$ *be unit vectors in* $\mathbb{R}^d$ *such that* $\angle(\mathbf{a}, \mathbf{b}) = \angle(\mathbf{u}, \mathbf{v}) = \theta$*, and let* Π *be the projector onto the span of* $\mathbf{a}$*,* $\mathbf{b}$*. Suppose* $\max\{\|\Pi\mathbf{u}\|, \|\Pi\mathbf{v}\|\} \leq \delta < 1$*. Let* $\mathbf{r}$ *be a random Gaussian vector. Then we have*

$$\mathbb{E}\left[\left(\frac{1 - \text{sign}(\mathbf{r}^T\mathbf{a})\,\text{sign}(\mathbf{r}^T\mathbf{b})}{2} - \frac{\theta}{\pi}\right)\left(\frac{1 - \text{sign}(\mathbf{r}^T\mathbf{u})\,\text{sign}(\mathbf{r}^T\mathbf{v})}{2} - \frac{\theta}{\pi}\right)\right] \leq 2\delta.$$

Here again, the expectation is over the choice of $\mathbf{r}$*.*

The proof is shown in the Appendix. Next, we show that $s_{\to t}(\mathbf{Dx})$ and $s_{\to t}(\mathbf{Dy})$ are actually close to orthogonal to the span of $\mathbf{Dx}$ and $\mathbf{Dy}$, by a general lemma.

Lemma 3.3. *Let* $\mathbf{x}, \mathbf{z} \in \mathbb{R}^d$ *be unit vectors, and* $||\mathbf{x}||_\infty \leq \rho$*, for some parameter* ρ*. Suppose* $\sigma_0, \sigma_1, \cdots, \sigma_{d-1}$ *are random signs taking* ± 1 *with equal probabilities. Let* $\gamma > 0$ *be some fixed offset. Then*

$$\mathbb{P}[s_{\to t}(\mathbf{Dx})^T\mathbf{Dz} \geq \gamma] \leq e^{-\gamma^2/8\rho^2}.$$

The proof of the above lemma is shown in the Appendix. We can now complete the proof using the lemmas above. First, for any two shifts $i \neq j$, we would like to show that the projection of $s_{\to i}\mathbf{Dx}$ onto $\text{span}(\mathbf{Dx}, \mathbf{Dy})$ is small. In other words, we need to show that the projection is small onto *any* unit vector in the span. Let $\mathbf{z}$ be any given unit vector in $\mathcal{S} := \text{span}(\mathbf{Dx}, \mathbf{Dy})$. We can now apply Lemma 3.3, to conclude that

$$\mathbb{P}[|s_{\to i}(\mathbf{Dx})^T\mathbf{z}| \leq \gamma] \leq 1 - e^{-\gamma^2/8\rho^2}. \tag{3.10}$$

Thus if we consider an orthogonal basis $\mathbf{z}_1, \mathbf{z}_2$ for $\mathcal{S}$, (3.10) is true for both $\mathbf{z}_1$ and $\mathbf{z}_2$. Note that $\mathbf{z}_1$ and $\mathbf{z}_2$ can be picked independently on $\mathbf{D}$, by setting $\mathbf{z}_1 = \mathbf{Dz}_1'$, $\mathbf{z}_2 = \mathbf{Dz}_2'$, such that $\mathbf{z}_1'$ and $\mathbf{z}_2'$ are a set of orthogonal bases in $\text{span}(\mathbf{x}, \mathbf{y})$. Thus by a union bound, we have that

$$\mathbb{P}[\max\{|s_{\to i}(\mathbf{Dx})^T\mathbf{z}_1|, ||s_{\to i}(\mathbf{Dx})^T\mathbf{z}_2|\} \leq \gamma] \geq 1 - 2e^{-\gamma^2/8\rho^2}. \tag{3.11}$$

Now suppose we have $\max\{|s_{\rightarrow i}(\mathbf{Dx})^T\mathbf{z}_1|, ||s_{\rightarrow i}(\mathbf{Dx})^T\mathbf{z}_2|\} \leq \gamma$. For *any* unit vector $\mathbf{z}$, it can be written as $\mathbf{z} = c_1\mathbf{z}_1 + c_2\mathbf{z}_2$, where $c_1^2 + c_2^2 = 1$. Thus by Cauchy-Schwartz, we get

$$\begin{aligned}(s_{\rightarrow i}(\mathbf{Dx})^T\mathbf{z})^2 &= (c_1 s_{\rightarrow i}(\mathbf{Dx})^T\mathbf{z}_1 + c_2 s_{\rightarrow i}(\mathbf{Dx})^T\mathbf{z}_2)^2 \\ &\leq (c_1^2 + c_2^2)((s_{\rightarrow i}(\mathbf{Dx})^T\mathbf{z}_1)^2 + (s_{\rightarrow i}(\mathbf{Dx})^T\mathbf{z}_2)^2) \\ &\leq 2\gamma^2.\end{aligned}$$

This implies that the projection onto $\mathcal{S}$ has length at most $\gamma\sqrt{2}$, whenever the event in (3.11) holds.

We can thus take a union over such events for all $i \neq j$ (there are k^2 such choices), to conclude that,

$$\mathbb{P}[\max_{i,j}\|\text{proj}(\mathbf{Dx},\ \text{span}(s_{\rightarrow i}\mathbf{Dx}, s_{\rightarrow i}\mathbf{Dy}\|) \leq \sqrt{2}\gamma] \geq 1 - 2k^2 e^{-\gamma^2/8\rho^2}. \tag{3.12}$$

The same can be done replacing $\mathbf{x}$ with $\mathbf{y}$, and hence the joint event holds with probability $1 - 4k^2 e^{-\gamma^2/8\rho^2}$.

Finally, we can appeal to Lemma 3.2 to conclude that $\mathbb{E}(X_i X_j) \leq 2\sqrt{2}\gamma$ for all i, j (where expectation is now over the choice of $\mathbf{r}$), with probability at least $1 - 4k^2 e^{-\gamma^2/8\rho^2}$ over the choice of $\mathbf{D}$. Thus, with probability at least $1 - \delta$, $\mathbb{E}(X_i X_j) \leq 8\rho\sqrt{\ln \frac{4k^2}{\delta}}$, for all i, j. This concludes the proof. □

3.4 Learning Circulant Binary Embedding

In the previous section, we showed the randomized CBE has LSH-like angle preserving properties, especially for high-dimensional data. One problem with the randomized CBE method is that it does not utilize the underlying data distribution while generating the matrix $\mathbf{R}$. In the next section, we propose to learn $\mathbf{R}$ in a data-dependent fashion, to minimize the distortions due to circulant projection and binarization.

We propose data-dependent CBE (CBE-opt), by optimizing the projection matrix with a novel time-frequency alternating optimization. We consider the following objective function in learning the d-bit CBE. The extension of learning $k < d$ bits will be shown in Section

3.4.2.

$$\begin{aligned} \underset{\mathbf{B},\mathbf{r}}{\text{argmin}} \quad & ||\mathbf{B} - \mathbf{X}\mathbf{R}^T||_F^2 + \lambda||\mathbf{R}\mathbf{R}^T - \mathbf{I}||_F^2 \\ \text{s.t.} \quad & \mathbf{R} = \text{circ}(\mathbf{r}), \end{aligned} \tag{3.13}$$

where $\mathbf{X} \in \mathbb{R}^{N\times d}$, is the data matrix containing n training points: $\mathbf{X} = [\mathbf{x}_0, \cdots, \mathbf{x}_{N-1}]^T$, and $\mathbf{B} \in \{-1, 1\}^{N\times d}$ is the corresponding binary code matrix.[5]

In the above optimization, the first term minimizes distortion due to binarization. The second term tries to make the projections (rows of $\mathbf{R}$, and hence the corresponding bits) as uncorrelated as possible. In other words, this helps to reduce the redundancy in the learned code. If $\mathbf{R}$ were to be an orthogonal matrix, the second term would vanish, and the optimization would find the best rotation such that the distortion due to binarization is minimized. However, being a circulant matrix, $\mathbf{R}$, in general, will not be orthogonal[6]. The similar objective has been used in previous works including [72, 73] and [216].

3.4.1 The Time-Frequency Alternating Optimization

The above is a difficult non-convex combinatorial optimization problem. In this section, we propose a novel approach to efficiently find a local solution. The idea is to alternatively optimize the objective by fixing $\mathbf{r}$, and $\mathbf{B}$, respectively. For a fixed $\mathbf{r}$, optimizing $\mathbf{B}$ can be easily performed in the input domain ("time" as opposed to "frequency"). For a fixed $\mathbf{B}$, the circulant structure of $\mathbf{R}$ makes it difficult to optimize the objective in the input domain. Hence we propose a novel method, by optimizing $\mathbf{r}$ in the frequency domain based on DFT. This leads to a very efficient procedure.

For a fixed r. The objective is independent on each element of $\mathbf{B}$. Denote B_{ij} as the

[5]If the data is ℓ_2 normalized, we can set $\mathbf{B} \in \{-1/\sqrt{d}, 1/\sqrt{d}\}^{N\times d}$ to make $\mathbf{B}$ and $\mathbf{X}\mathbf{R}^T$ more comparable. This does not empirically influence the performance.

[6]We note that the rank of the circulant matrices can range from 1 (an all-1 matrix) to d (an identity matrix).

element of the i-th row and j-th column of $\mathbf{B}$. It is easy to show that $\mathbf{B}$ can be updated as:

$$B_{ij} = \begin{cases} 1 & \text{if } \mathbf{R}_{j\cdot}\mathbf{x}_i \geq 0 \\ -1 & \text{if } \mathbf{R}_{j\cdot}\mathbf{x}_i < 0 \end{cases}, \tag{3.14}$$

$$i = 0, \cdots, N-1. \quad j = 0, \cdots, d-1.$$

For a fixed B. Define $\tilde{\mathbf{r}}$ as the DFT of the circulant vector $\tilde{\mathbf{r}} := \mathcal{F}(\mathbf{r})$. Instead of solving $\mathbf{r}$ directly, we propose to solve $\tilde{\mathbf{r}}$, from which $\mathbf{r}$ can be recovered by IDFT.

The key to our derivation is the fact that DFT projects the signal to a set of orthogonal basis. Therefore the ℓ_2 norm can be preserved. Formally, according to Parseval's theorem , for any $\mathbf{t} \in \mathbb{C}^d$ [154],

$$||\mathbf{t}||_2^2 = (1/d)||\mathcal{F}(\mathbf{t})||_2^2. \tag{3.15}$$

Denote diag$(\cdot)$ as the diagonal matrix formed by a vector. Denote $\Re(\cdot)$ and $\Im(\cdot)$ as the real and imaginary parts, respectively. We use $\mathbf{B}_{i\cdot}$ to denote the i-th row of $\mathbf{B}$. With complex arithmetic, the first term in (3.13) can be expressed in the frequency domain as:

$$\begin{aligned} &||\mathbf{B} - \mathbf{X}\mathbf{R}^T||_F^2 = \frac{1}{d}\sum_{i=0}^{N-1} ||\mathcal{F}(\mathbf{B}_{i\cdot}^T - \mathbf{R}\mathbf{x}_i)||_2^2 \qquad (3.16)\\ =&\frac{1}{d}\sum_{i=0}^{N-1} ||\mathcal{F}(\mathbf{B}_{i\cdot}^T) - \tilde{\mathbf{r}} \circ \mathcal{F}(\mathbf{x}_i)||_2^2 = \frac{1}{d}\sum_{i=0}^{N-1} ||\mathcal{F}(\mathbf{B}_{i\cdot}^T) - \text{diag}(\mathcal{F}(\mathbf{x}_i))\tilde{\mathbf{r}}||_2^2 \\ =&\frac{1}{d}\sum_{i=0}^{N-1} \left(\mathcal{F}(\mathbf{B}_{i\cdot}^T) - \text{diag}(\mathcal{F}(\mathbf{x}_i))\tilde{\mathbf{r}}\right)^T \left(\mathcal{F}(\mathbf{B}_{i\cdot}^T) - \text{diag}(\mathcal{F}(\mathbf{x}_i))\tilde{\mathbf{r}}\right) \\ =&\frac{1}{d}\Big[\Re(\tilde{\mathbf{r}})^T\mathbf{M}\Re(\tilde{\mathbf{r}}) + \Im(\tilde{\mathbf{r}})^T\mathbf{M}\Im(\tilde{\mathbf{r}}) + \Re(\tilde{\mathbf{r}})^T\mathbf{h} + \Im(\tilde{\mathbf{r}})^T\mathbf{g}\Big] + ||\mathbf{B}||_F^2, \end{aligned}$$

where,

$$\mathbf{M} = \text{diag}\big(\sum_{i=0}^{N-1} \Re(\mathcal{F}(\mathbf{x}_i)) \circ \Re(\mathcal{F}(\mathbf{x}_i)) + \Im(\mathcal{F}(\mathbf{x}_i)) \circ \Im(\mathcal{F}(\mathbf{x}_i))\big), \tag{3.17}$$

$$\mathbf{h} = -2\sum_{i=0}^{N-1} \Re(\mathcal{F}(\mathbf{x}_i)) \circ \Re(\mathcal{F}(\mathbf{B}_{i\cdot}^T)) + \Im(\mathcal{F}(\mathbf{x}_i)) \circ \Im(\mathcal{F}(\mathbf{B}_{i\cdot}^T)), \tag{3.18}$$

$$\mathbf{g} = 2\sum_{i=0}^{N-1} \Im(\mathcal{F}(\mathbf{x}_i)) \circ \Re(\mathcal{F}(\mathbf{B}_{i\cdot}^T)) - \Re(\mathcal{F}(\mathbf{x}_i)) \circ \Im(\mathcal{F}(\mathbf{B}_{i\cdot}^T)). \tag{3.19}$$

The above can be derived based on the following fact. For any $\mathbf{Q} \in \mathbb{C}^{d\times d}$, $\mathbf{s}$, $\mathbf{t} \in \mathbb{C}^d$,

$$
\begin{aligned}
&||\mathbf{s}-\mathbf{Q}\mathbf{t}||_2^2 = (\mathbf{s}-\mathbf{Q}\mathbf{t})^H(\mathbf{s}-\mathbf{Q}\mathbf{t}) \qquad (3.20)\\
=&\mathbf{s}^H\mathbf{s} - \mathbf{s}^H\mathbf{Q}\mathbf{t} - \mathbf{t}^H\mathbf{Q}^H\mathbf{s} + \mathbf{t}^H\mathbf{Q}^HA\mathbf{t}\\
=&\Re(\mathbf{s})^T\Re(\mathbf{s}) + \Im(\mathbf{s})^T\Im(\mathbf{s}) - 2\Re(\mathbf{t})^T(\Re(\mathbf{Q})^T\Re(\mathbf{s}) + \Im(\mathbf{Q})^T\Im(\mathbf{s}))\\
&+ 2\Im(\mathbf{t})^T(\Im(\mathbf{Q})^T\Re(\mathbf{s}) - \Re(\mathbf{Q})^T\Im(\mathbf{s})) + \Re(\mathbf{t})^T(\Re(\mathbf{Q})^T\Re(\mathbf{Q}) + \Im(\mathbf{Q})^T\Im(\mathbf{Q}))\Re(\mathbf{t})\\
&+ \Im(\mathbf{t})^T(\Re(\mathbf{Q})^T\Re(\mathbf{Q}) + \Im(\mathbf{Q})^T\Im(\mathbf{Q}))\Im(\mathbf{t}) + 2\Re(\mathbf{t})^T(\Im(\mathbf{Q})^T\Re(\mathbf{Q}) - \Re(\mathbf{Q})^T\Im(\mathbf{Q}))\Im(\mathbf{t}).
\end{aligned}
$$

For the second term in (3.13), we note that the circulant matrix can be diagonalized by DFT matrix $\mathbf{F}_d$ and its conjugate transpose $\mathbf{F}_d^H$. Formally, for $\mathbf{R} = \text{circ}(\mathbf{r})$, $\mathbf{r} \in \mathbb{R}^d$,

$$
\mathbf{R} = (1/d)\mathbf{F}_d^H \text{diag}(\mathcal{F}(\mathbf{r}))\mathbf{F}_d. \qquad (3.21)
$$

Let $\text{Tr}(\cdot)$ be the trace of a matrix. Therefore,

$$
\begin{aligned}
&||\mathbf{R}\mathbf{R}^T - \mathbf{I}||_F^2 = ||\frac{1}{d}\mathbf{F}_d^H(\text{diag}(\tilde{\mathbf{r}})^H\text{diag}(\tilde{\mathbf{r}}) - \mathbf{I})\mathbf{F}_d||_F^2 \qquad (3.22)\\
=&\,\text{Tr}\left[\frac{1}{d}\mathbf{F}_d^H(\text{diag}(\tilde{\mathbf{r}})^H\text{diag}(\tilde{\mathbf{r}}) - \mathbf{I})^H(\text{diag}(\tilde{\mathbf{r}})^H\text{diag}(\tilde{\mathbf{r}}) - \mathbf{I})\mathbf{F}_d\right]\\
=&\,\text{Tr}\left[(\text{diag}(\tilde{\mathbf{r}})^H\text{diag}(\tilde{\mathbf{r}}) - \mathbf{I})^H(\text{diag}(\tilde{\mathbf{r}})^H\text{diag}(\tilde{\mathbf{r}}) - \mathbf{I})\right]\\
=&||\tilde{\mathbf{r}}^H \circ \tilde{\mathbf{r}} - \mathbf{1}||_2^2 = ||\Re(\tilde{\mathbf{r}})^2 + \Im(\tilde{\mathbf{r}})^2 - \mathbf{1}||_2^2.
\end{aligned}
$$

Furthermore, as $\mathbf{r}$ is real-valued, additional constraints on $\tilde{\mathbf{r}}$ are needed. For any $u \in \mathbb{C}$, denote $\overline{u}$ as its complex conjugate. We have the following result [154]: For any real-valued vector $\mathbf{t} \in \mathbb{C}^d$, $\mathcal{F}(\mathbf{t})_0$ is real-valued, and

$$
\mathcal{F}(\mathbf{t})_{d-i} = \overline{\mathcal{F}(\mathbf{t})_i}, \quad i = 1, \cdots, \lfloor d/2 \rfloor. \qquad (3.23)
$$

From (3.16) − (3.23), the problem of optimizing $\tilde{\mathbf{r}}$ becomes

$$
\begin{aligned}
\underset{\tilde{\mathbf{r}}}{\text{argmin}} \quad & \Re(\tilde{\mathbf{r}})^T\mathbf{M}\Re(\tilde{\mathbf{r}}) + \Im(\tilde{\mathbf{r}})^T\mathbf{M}\Im(\tilde{\mathbf{r}}) + \Re(\tilde{\mathbf{r}})^T\mathbf{h}\\
& + \Im(\tilde{\mathbf{r}})^T\mathbf{g} + \lambda d||\Re(\tilde{\mathbf{r}})^2 + \Im(\tilde{\mathbf{r}})^2 - \mathbf{1}||_2^2 \qquad (3.24)\\
\text{s.t.} \quad & \Im(\tilde{r}_0) = 0\\
& \Re(\tilde{r}_i) = \Re(\tilde{r}_{d-i}), i = 1, \cdots, \lfloor d/2 \rfloor\\
& \Im(\tilde{r}_i) = -\Im(\tilde{r}_{d-i}), i = 1, \cdots, \lfloor d/2 \rfloor.
\end{aligned}
$$

The above is non-convex. Fortunately, the objective function can be decomposed, such that we can solve two variables at a time. Denote the diagonal vector of the diagonal matrix $\mathbf{M}$ as $\mathbf{m}$. The above optimization can then be decomposed to the following sets of optimizations.

$$\begin{aligned}
&\underset{\tilde{r}_0}{\text{argmin}} \quad m_0\tilde{r}_0^2 + h_0\tilde{r}_0 + \lambda d\left(\tilde{r}_0^2 - 1\right)^2, \text{ s.t. } \tilde{r}_0 = \overline{\tilde{r}_0}. \qquad (3.25)\\
&\underset{\tilde{r}_i}{\text{argmin}} \quad (m_i + m_{d-i})(\Re(\tilde{r}_i)^2 + \Im(\tilde{r}_i)^2) + 2\lambda d\left(\Re(\tilde{r}_i)^2 + \Im(\tilde{r}_i)^2 - 1\right)^2\\
&\qquad\qquad + (h_i + h_{d-i})\Re(\tilde{r}_i) + (g_i - g_{d-i})\Im(\tilde{r}_i), \quad i = 1, \cdots, \lfloor d/2 \rfloor.
\end{aligned}$$

In (3.25), we need to minimize a 4^{th} order polynomial with one variable, with the closed form solution readily available. In (3.26), we need to minimize a 4^{th} order polynomial with two variables. Though the closed form solution is hard to find (requiring solution of a cubic bivariate system), a local minima can be found by gradient descent, which in practice has constant running time for such small-scale problems. The overall objective is guaranteed to be non-increasing in each step. In practice, we find that a good solution can be reached within just 5-10 iterations. Therefore in practice, the proposed time-frequency alternating optimization procedure has running time $\mathcal{O}(Nd \log d)$.

3.4.2 Learning with Dimensionality Reduction

In the case of learning $k < d$ bits, we need to solve the following optimization problem:

$$\begin{aligned}
\underset{\mathbf{B},\mathbf{r}}{\text{argmin}} \quad & ||\mathbf{B}\mathbf{P}_k - \mathbf{X}\mathbf{P}_k^T\mathbf{R}^T||_F^2 + \lambda||\mathbf{R}\mathbf{P}_k\mathbf{P}_k^T\mathbf{R}^T - \mathbf{I}||_F^2 \qquad (3.26)\\
\text{s.t.} \quad & \mathbf{R} = \text{circ}(\mathbf{r}),
\end{aligned}$$

in which $\mathbf{P}_k = \begin{bmatrix} \mathbf{I}_k & \mathbf{O} \\ \mathbf{O} & \mathbf{O}_{d-k} \end{bmatrix}$, $\mathbf{I}_k$ is a $k \times k$ identity matrix, and $\mathbf{O}_{d-k}$ is a $(d-k) \times (d-k)$ all-zero matrix.

In fact, the right multiplication of $\mathbf{P}_k$ can be understood as a "temporal cut-off", which is equivalent to a frequency domain convolution. This makes the optimization difficult, as the objective in the frequency domain can no longer be decomposed. To address this issue, we propose a simple solution in which $B_{ij} = 0$, $i = 0, \cdots, N-1, j = k, \cdots, d-1$ in (3.13). Thus, the optimization procedure remains the same, and the cost is also $\mathcal{O}(Nd \log d)$. We will show in experiments that this heuristic provides good performance in practice.

3.5 Experiments

To compare the performance of the circulant binary embedding techniques, we conducte experiments on three real-world high-dimensional datasets used by the current state-of-the-art method for generating long binary codes [73]. The Flickr-25600 dataset contains 100K images sampled from a noisy Internet image collection. Each image is represented by a $25,600$-dimensional vector. The ImageNet-51200 contains 100k images sampled from 100 random classes from ImageNet [47], each represented by a $51,200$ dimensional vector. The third dataset (ImageNet-25600) is another random subset of ImageNet containing 100K images in $25,600$-dimensional space. All the vectors are normalized to be of the unit norm.

We compare the performance of the randomized (CBE-rand) and learned (CBE-opt) versions of our circulant embeddings with the current state-of-the-art for high-dimensional data, *i.e.*, bilinear embeddings. We use both the randomized (bilinear-rand) and learned (bilinear-opt) versions. Bilinear embeddings have been shown to perform similarly or better than another promising technique called Product Quantization [92]. Finally, we also compare against the binary codes produced by the baseline LSH method [32], which is still applicable to 25,600 and 51,200-dimensional feature but with much longer running time and much more space. We also show an experiment with a relatively low-dimensional feature (2048, with Flickr data) to compare against techniques that perform well for low-dimensional data but do not scale to the high-dimensional scenario. Example techniques include ITQ [72], SH [218], SKLSH [170], and AQBC [71].

Following [73, 152, 74], we use 10,000 randomly sampled instances for training. We then randomly sample 500 instances, different from the training set as queries. The performance (recall@1-100) is evaluated by averaging the recalls of the query instances. The ground-truth of each query instance is defined as its 10 nearest neighbors based on ℓ_2 distance. For each dataset, we conduct two sets of experiments: *fixed-time* where code generation time is fixed and *fixed-bits* where the number of bits is fixed across all techniques. We also show an experiment where the binary codes are used for classification.

The proposed CBE method is found robust to the choice of λ in (3.13). For example, in the retrieval experiments, the performance difference for $\lambda = 0.1$, 1, 10, is within 0.5%. Therefore, in all the experiments, we simply fix $\lambda = 1$. For the bilinear method, in order

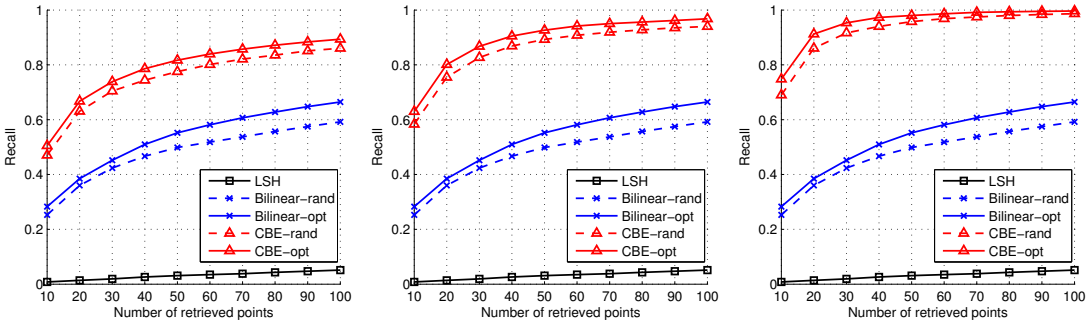
d	Full projection	Bilinear projection	Circulant projection
2^{15}	5.44×10^2	2.85	1.11
2^{17}	-	1.91×10^1	4.23
2^{20} (1M)	-	3.76×10^2	3.77×10^1
2^{24}	-	1.22×10^4	8.10×10^2
2^{27} (100M)	-	2.68×10^5	8.15×10^3

Table 3.2: Computational time (ms) of full projection (LSH, ITQ, SH *etc.*), bilinear projection (Bilinear), and circulant projection (CBE). The time is based on a single 2.9GHz CPU core. The error is within 10%. An empty cell indicates that the memory needed for that method is larger than the machine limit of 24GB.

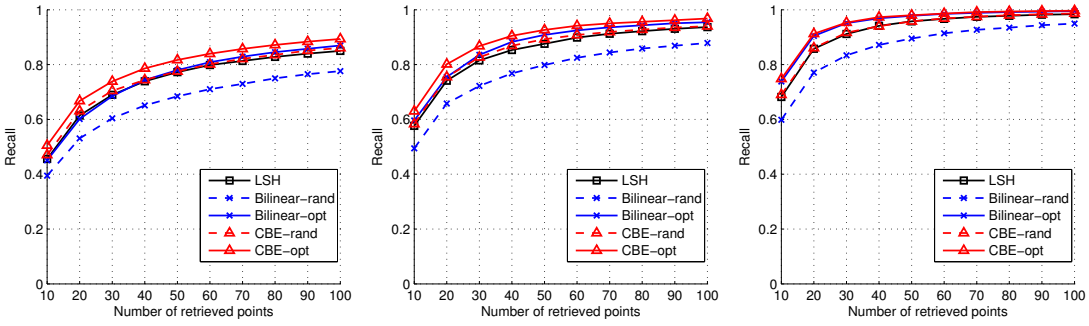
to get fast computation, the feature vector is reshaped to a near-square matrix, and the dimension of the two bilinear projection matrices are also chosen to be close to square. Parameters for other techniques are tuned to give the best results on these datasets.

3.5.1 Computational Time

When generating k-bit code for d-dimensional data, the full projection, bilinear projection, and circulant projection methods have time complexity $O(kd)$, $O(\sqrt{k}d)$, and $O(d \log d)$, respectively. We compare the computational time in Table 3.2 on a fixed hardware. Based on our implementation, the computational time of the above three methods can be roughly characterized as $d^2 : d\sqrt{d} : 5d \log d$. Note that faster implementation of FFT algorithms will lead to better computational time for CBE by further reducing the constant factor. Due to the small storage requirement $\mathcal{O}(d)$, and the wide availability of highly optimized FFT libraries, CBE is also suitable for implementation on GPU. Our preliminary tests based on GPU shows up to 20 times speedup compared with CPU. In this chapter, for a fair comparison, we use same CPU based implementation for all the methods.



(a) #bits(CBE) = 6,400 (b) #bits(CBE) = 12,800 (c) #bits(CBE) = 25,600

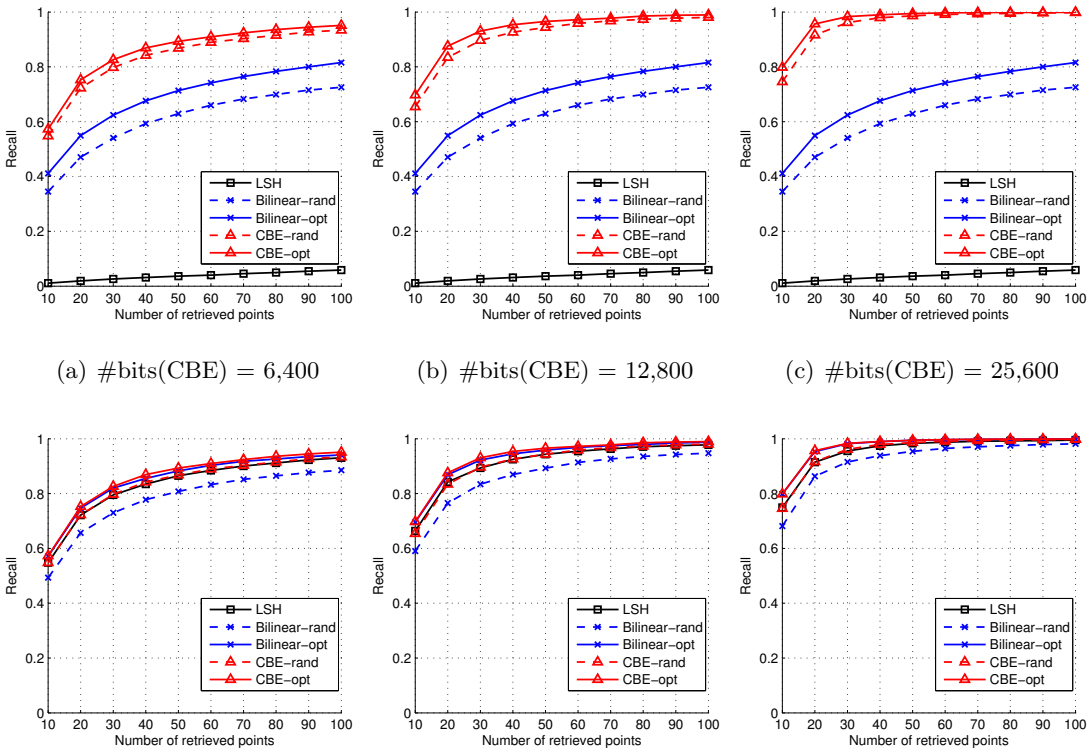


(d) # bits (all) = 6,400 (e) # bits (all) = 12,800 (f) # bits (all) = 25,600

Figure 3.1: Recall on Flickr-25600. The standard deviation is within 1%. **First Row**: Fixed time. "# bits" is the number of bits of CBE. Other methods are using fewer bits to make their computational time identical to CBE. **Second Row**: Fixed number of bits. CBE-opt/CBE-rand are 2-3 times faster than Bilinear-opt/Bilinear-rand, and hundreds of times faster than LSH.

3.5.2 Retrieval

The recalls of different methods are compared on the three datasets, shown in Figure 3.1 – 3.3. The top row in each figure shows the performance of different methods when the code generation time for all the methods is kept the same as that of CBE. For a fixed time, the proposed CBE yields much better recall than other methods. Even CBE-rand outperforms LSH and Bilinear code by a large margin. The second row compares the performance for different techniques with codes of the same length. In this case, the performance of CBE-rand is almost identical to LSH even though it is hundreds of time



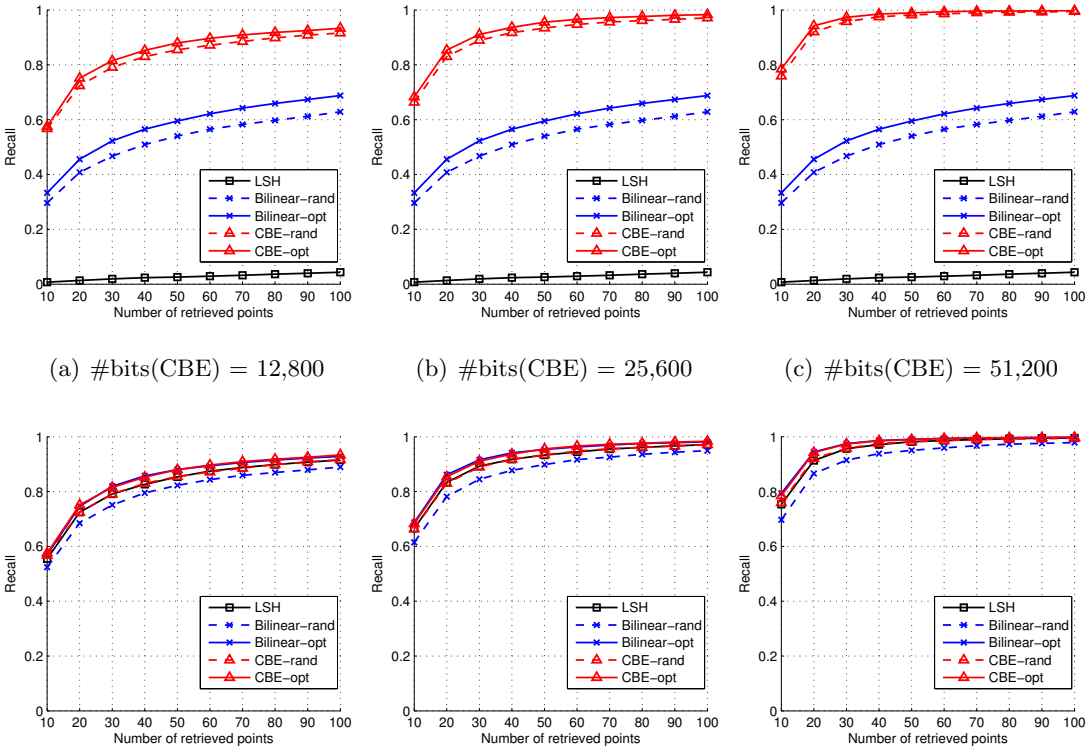
(a) #bits(CBE) = 6,400 (b) #bits(CBE) = 12,800 (c) #bits(CBE) = 25,600

(d) # bits (all) = 64,00 (e) # bits (all) = 12,800 (f) # bits (all) = 25,600

Figure 3.2: Recall on ImageNet-25600. The standard deviation is within 1%. **First Row**: Fixed time. "# bits" is the number of bits of CBE. Other methods are using fewer bits to make their computational time identical to CBE. **Second Row**: Fixed number of bits. CBE-opt/CBE-rand are 2-3 times faster than Bilinear-opt/Bilinear-rand, and hundreds of times faster than LSH.

faster. This is consistent with our analysis in Section 3.3. Moreover, CBE-opt/CBE-rand outperform Bilinear-opt/Bilinear-rand in addition to being 2-3 times faster.

There exist several techniques that do not scale to the high-dimensional case. To compare our method with those, we conduct experiments with fixed number of bits on a relatively low-dimensional dataset (Flickr-2048), constructed by randomly sampling 2,048 dimensions of Flickr-25600. As shown in Figure 3.4, though CBE is not designed for such scenario, the CBE-opt performs better or equivalent to other techniques except ITQ which scales very poorly with d ($\mathcal{O}(d^3)$). Moreover, as the number of bits increases, the gap be-



(a) #bits(CBE) = 12,800 (b) #bits(CBE) = 25,600 (c) #bits(CBE) = 51,200

(d) # bits (all) = 12,800 (e) # bits (all) = 25,600 (f) # bits (all) = 51,200

Figure 3.3: Recall on ImageNet-51200. The standard deviation is within 1%. **First Row**: Fixed time. "# bits" is the number of bits of CBE. Other methods are using fewer bits to make their computational time identical to CBE. **Second Row**: Fixed number of bits. CBE-opt/CBE-rand are 2-3 times faster than Bilinear-opt/Bilinear-rand, and hundreds of times faster than LSH.

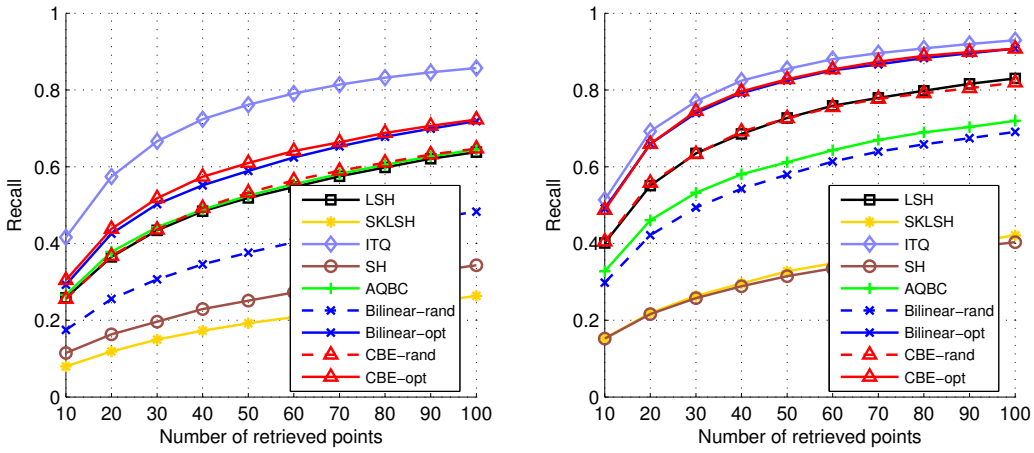
tween ITQ and CBE becomes much smaller suggesting that the performance of ITQ is not expected to be better than CBE even if one could run ITQ on high-dimensional data.

3.5.3 Classification

Besides retrieval, we also test the binary codes for classification. The advantage is to save on storage, allowing even large-scale datasets to fit in memory [127, 183]. We follow the asymmetric setting of [183] by training linear SVM on binary code sign($\mathbf{Rx}$), and testing on the original $\mathbf{Rx}$. Empirically, this has been shown to give better accuracy than the

Original	LSH	Bilinear-opt	CBE-opt
25.59±0.33	23.49±0.24	24.02±0.35	24.55 ±0.30

Table 3.3: Multiclass classification accuracy (%) on binary coded ImageNet-25600. The binary codes of same dimensionality are 32 times more space efficient than the original features (single-float).



(a) # bits = 1,024 (b) # bits = 2,048

Figure 3.4: Performance comparison on relatively low-dimensional data (Flickr-2048) with fixed number of bits. CBE gives comparable performance to the state-of-the-art even on low-dimensional data as the number of bits is increased. However, these other methods do not scale to very high-dimensional data setting which is the main focus of this work.

symmetric procedure. We use ImageNet-25600, with randomly sampled 100 images per category for training, 50 for validation and 50 for testing. The code dimension is set as 25,600. As shown in Table 3.3, CBE, which has much faster computation, does not show any performance degradation compared with LSH or bilinear codes in the classification task.

3.6 Semi-supervised Extension

In some applications, one can have access to a few labeled pairs of similar and dissimilar data points. Here we show how the CBE formulation can be extended to incorporate such

information in learning. This is achieved by adding an additional objective term $J(\mathbf{R})$.

$$\begin{aligned} \underset{\mathbf{B},\mathbf{r}}{\text{argmin}} \quad & ||\mathbf{B} - \mathbf{X}\mathbf{R}^T||_F^2 + \lambda||\mathbf{R}\mathbf{R}^T - \mathbf{I}||_F^2 + \mu J(\mathbf{R}) \\ \text{s.t.} \quad & \mathbf{R} = \text{circ}(\mathbf{r}), \end{aligned} \tag{3.27}$$

$$J(\mathbf{R}) = \sum_{i,j \in \mathcal{M}} ||\mathbf{R}\mathbf{x}_i - \mathbf{R}\mathbf{x}_j||_2^2 - \sum_{i,j \in \mathcal{D}} ||\mathbf{R}\mathbf{x}_i - \mathbf{R}\mathbf{x}_j||_2^2. \tag{3.28}$$

Here $\mathcal{M}$ and $\mathcal{D}$ are the set of "similar" and "dissimilar" instances, respectively. The intuition is to maximize the distances between the dissimilar pairs and minimize the distances between the similar pairs. Such a term is commonly used in semi-supervised binary coding methods [216]. We again use the time-frequency alternating optimization procedure of Section 3.4. For a fixed $\mathbf{r}$, the optimization procedure to update $\mathbf{B}$ is the same. For a fixed $\mathbf{B}$, optimizing $\mathbf{r}$ is done in frequency domain by expanding $J(\mathbf{R})$ as below, with similar techniques used in Section 3.4.

$$||\mathbf{R}\mathbf{x}_i - \mathbf{R}\mathbf{x}_j||_2^2 = (1/d)||\text{diag}(\mathcal{F}(\mathbf{x}_i) - \mathcal{F}(\mathbf{x}_j))\tilde{\mathbf{r}}||_2^2. \tag{3.29}$$

Therefore,

$$J(\mathbf{R}) = (1/d)(\Re(\tilde{\mathbf{r}})^T \mathbf{A} \Re(\tilde{\mathbf{r}}) + \Im(\tilde{\mathbf{r}})^T \mathbf{A} \Im(\tilde{\mathbf{r}})), \tag{3.30}$$

where $\mathbf{A} = \mathbf{A}_1 + \mathbf{A}_2 - \mathbf{A}_3 - \mathbf{A}_4$, and

$$\mathbf{A}_1 = \sum_{(i,j) \in \mathcal{M}} \Re(\text{diag}(\mathcal{F}(\mathbf{x}_i) - \mathcal{F}(\mathbf{x}_j)))^T \Re(\text{diag}(\mathcal{F}(\mathbf{x}_i) - \mathcal{F}(\mathbf{x}_j))), \tag{3.31}$$

$$\mathbf{A}_2 = \sum_{(i,j) \in \mathcal{M}} \Im(\text{diag}(\mathcal{F}(\mathbf{x}_i) - \mathcal{F}(\mathbf{x}_j)))^T \Im(\text{diag}(\mathcal{F}(\mathbf{x}_i) - \mathcal{F}(\mathbf{x}_j))), \tag{3.32}$$

$$\mathbf{A}_3 = \sum_{(i,j) \in \mathcal{D}} \Re(\text{diag}(\mathcal{F}(\mathbf{x}_i) - \mathcal{F}(\mathbf{x}_j)))^T \Re(\text{diag}(\mathcal{F}(\mathbf{x}_i) - \mathcal{F}(\mathbf{x}_j))), \tag{3.33}$$

$$\mathbf{A}_4 = \sum_{(i,j) \in \mathcal{D}} \Im(\text{diag}(\mathcal{F}(\mathbf{x}_i) - \mathcal{F}(\mathbf{x}_j)))^T \Im(\text{diag}(\mathcal{F}(\mathbf{x}_i) - \mathcal{F}(\mathbf{x}_j))). \tag{3.34}$$

Hence, the optimization can be carried out as in Section 3.4, where $\mathbf{M}$ in (3.16) is simply replaced by $\mathbf{M} + \mu\mathbf{A}$. Our experiments show that the semi-supervised extension improves over the non-semi-supervised version by 2% in terms of averaged AUC on the ImageNet-25600 dataset.

3.7 Conclusion and Future Works

We proposed Circulant Binary Embedding (CBE) for generating long codes for very high-dimensional data. We showed that the angle preserving property of randomized CBE can be as good as LSH when applied on high-dimensional data. A novel time-frequency alternating optimization was also introduced to learn the model parameters from the training data. The proposed method has time complexity $\mathcal{O}(d \log d)$ and space complexity $\mathcal{O}(d)$, while showing no performance degradation on real-world data compared with more expensive approaches ($\mathcal{O}(d^2)$ or $\mathcal{O}(d^{1.5})$). On the contrary, for the fixed time, it showed significant accuracy gains. The full potential of the method can be unleashed when applied to ultra-high dimensional data (say $d \sim$100M), for which no other methods are applicable.

For the future works, we are exploring alternative structured projections which can be even more flexibility in terms of the computation and space cost. It is also worthwhile to explore generalized FFT or FFTs defined on other finite groups [108, 140, 175].

Chapter 4

Circulant Neural Network

4.1 Introduction

Deep neural network-based methods have recently achieved dramatic accuracy improvements in many areas of computer vision, including image classification [110, 237, 132], object detection [69, 185], face recognition [200, 198], and text recognition [18, 91]. These high-performing methods rely on deep networks containing millions or even billions of parameters. For example, the work by Krizhevsky et al.[110] achieved breakthrough results on the 2012 ImageNet challenge using a network containing 60 million parameters with five convolutional layers and three fully-connected layers. The "DeepFace" system [200] obtained face verification results close to human performance on the Labeled Faces in the Wild (LFW) dataset with a network containing 120 million parameters and a mix of convolutional, locally-connected, and fully-connected layers. If only fully-connected layers are applied, the number of parameters can grow to billions [46]. During the training stage, in order to avoid overfitting, usually millions of training samples are required to train such high-dimensional models, demanding heavy computational processing.

As larger neural networks are considered, with more layers and also more nodes in each layer, reducing their storage and computational costs become critical to meet the requirements of practical applications. Current efforts towards this goal focus mostly on the optimization of convolutional layers [90, 141, 56], which consume the bulk of computational processing in modern convolutional architectures. We instead explore the redundancy of

parameters in fully-connected layers, which are often the bottleneck in terms of memory consumption. In this chapter, we propose a solution based on *circulant projections* to significantly reduce the storage and computational costs of fully-connected neural network layers while maintaining competitive error rates.

A basic computation in a fully-connected neural network layer is

$$h(\mathbf{x}) = \phi(\mathbf{R}\mathbf{x}), \tag{4.1}$$

where $\mathbf{R} \in \mathbb{R}^{k \times d}$, and $\phi(\cdot)$ is a element-wise nonlinear activation function. The above operation connects a layer with d nodes and a layer with k nodes. In a multi-layer perceptron, for example, all the layers are fully-connected. In convolutional neural networks, the fully connected layers are often used before the final softmax output layer, in order to capture global properties of the image. The computational complexity and space complexity of this linear projection are $\mathcal{O}(dk)$. In practice, k is usually comparable or even larger than d. This leads to computation and space complexity at least $\mathcal{O}(d^2)$, creating a bottleneck for many neural network architectures. In fact, fully-connected layers in modern convolutional architectures typically contain over 90% of the network parameters.

In this work, we propose to impose a *circulant structure* on the projection matrix $\mathbf{R}$ in (4.1). This special structure allows us to use the Fast Fourier Transform (FFT) to speed up the computation. Considering a neural network layer with d input nodes, and d output nodes, the proposed method reduces dramatically the complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d \log d)$, and space complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d)$. Table 4.1 compares the computation and space complexity of the proposed approach with the conventional method.

Although the circulant matrix is highly structured with very small amount of parameters ($\mathcal{O}(d)$), it captures the global information well and does not impact the final performance much. We show by experiments that our method can provide a significant reduction of storage and computational costs while achieving very competitive error rates. Our work makes the following contributions:

- We propose to impose the circulant structure on the linear projection matrix of fully-connected layers of neural networks in order to speed up computations and reduce storage costs. (Section 4.3)

Method	Time	Space	Time (Learning)
Conventional NN	$\mathcal{O}(d^2)$	$\mathcal{O}(d^2)$	$\mathcal{O}(Nd^2)$
Circulant NN	$\mathcal{O}(d \log d)$	$\mathcal{O}(d)$	$\mathcal{O}(Nd \log d)$

Table 4.1: Comparison of the proposed method with neural networks based on unstructured projections. We assume a fully-connected layer, and the number of input nodes and number of output nodes are both d. N is the number of training examples.

- We propose a method which can efficiently optimize the neural network while keeping the circulant structure. (Section 4.5)
- We demonstrate by experiments on visual data that the proposed method can speed up the computation and reduce memory needs while maintaining competitive error rates. In addition, with much fewer parameters, our method is empirically shown to require less training data. (Section 4.6)

4.2 Related Works

Compressing Neural Networks. The work of Collins and Kohli [39] addressed the problem of memory usage in deep networks by applying sparsity-inducing regularizers during training to encourage zero-weight connections in the convolutional and fully-connected layers. Different from the above methods where memory consumption is reduced only at test time, our method cuts down storage costs at both training and testing times. Other approaches exploited low-rank matrix factorization [182, 48] to reduce the number of neural network parameters. In contrast, our approach exploits the redundancy in the parametrization of deep architectures by imposing a circulant structure on the projection matrix, reducing its storage to a single column vector, while allowing the use of FFT for faster computation.

Techniques based on *knowledge distillation* [84] aimed to compress the knowledge of a network with a large set of parameters into a compact and fast-to-execute network model. This can be achieved by training a compact model to imitate the soft outputs of a larger model. Romero et al. [177] further showed that the intermediate representations learned by

the large model serve as hints to improve the training process and final performance of the compact model. In contrast, our work does not require the training of an auxiliary model.

Network in Network [132] was recently proposed as a tool for richer local patch modeling in convolutional networks, where linear convolutions in each layer were replaced by convolving the input with a *micro-network* filter defined, for example, by a multi-layer perceptron. The inception architecture [199] extended this work by using these micro-networks as dimensionality reduction modules to remove computational bottlenecks and reduce storage costs. A key differentiating aspect is that we focus on modeling global dependencies and reducing the cost of fully connected layers, which usually contain the large majority of parameters in standard configurations. Therefore, our work is complementary to these methods. Although Lin et al. [132] suggested that fully-connected layers could be replaced by average pooling without hurting performance for general image classification, other work in computer vision [200] and speech recognition [193] highlighted the importance of these layers to capture global dependencies and achieve state-of-the-art results.

Speeding up Neural Networks. Several recent methods have been proposed to speed-up the computation of neural networks, with the focus on convolutional architectures [90, 141, 56, 49]. Related to our work, Mathieu et al. [141] used the Fast Fourier Transform to accelerate the computation of convolutional layers, through the well-known *Convolution Theorem*. In contrast, our work focuses on the optimization of fully-connected layers by imposing the circulant structure on the weight matrix of each layer to speed up the computation in both training and testing stages. In the context of object detection, many techniques such as detector cascades or segmentation-based selective search [205, 56] have been proposed to reduce the number of candidate object locations in which a deep neural network is applied. Our proposed approach is complementary to these techniques. Other approaches for speeding up neural networks rely on hardware-specific optimizations. For example, fast neural network implementations have been proposed for GPUs [51], CPUs [206], and FPGAs [59]. Our method is also related to the recent efforts of "shallow" neural networks, which showed that sometimes shallow structures can match the performance of deep structures [143, 144, 88, 235].

4.3 Circulant Neural Network

We present the general framework of using circulant projections to speed up fully connected layers of a neural network. For $\mathbf{x} \in \mathbb{R}^d$, defined its d-dimensional output with $\mathbf{r} \in \mathbb{R}^d$:

$$h(\mathbf{x}) = \phi(\mathbf{RDx}), \quad \mathbf{R} = \text{circ}(\mathbf{r}), \tag{4.2}$$

where the circulant matrix $\mathbf{R}$ and the sign flipping matrix $\mathbf{D}$ in defined as in Section 2.3. For d-dimensional data, the 1-layer circulant neural network has space complexity $\mathcal{O}(d)$, and time complexity $\mathcal{O}(d \log d)$. Following the former section, because the multiplication with $\mathbf{D}$ can be seen as an $\mathcal{O}(d)$ pre-processing step, we will omit it for clear presentation. The setting of $k = d$ is commonly used in fully connected layers of recent convolutional neural network architectures. When $k \neq d$, the framework can be adapted based on Section 2.3.2[1].

It is shown in previous works that when the parameters of the circulant projection matrix are generated *iid* from the standard normal distribution, the circulant projection (with the random sign flipping matrix) mimics an unstructured randomized projection [82, 212, 234]. It is then reasonable to conjecture that randomized circulant projections can also achieve good performance in neural networks (compared with using unstructured randomized matrices). This is indeed true as shown in the experiment section. And same as binary embedding and kernel approximation, by optimizing the parameters of the projection matrix, we can significantly improve the performance.

4.4 Randomized Circulant Neural Networks

We first consider the case where the elements of $\mathbf{r}$ in (4.2) are generated independently from a standard normal distribution $\mathcal{N}(0, 1)$. We refer to these models as randomized circulant neural networks. In this case, the parameters of the circulant projections are defined by random weights, without optimization. In other words, in the optimization process, only the parameters of convolutional layers and the softmax classification layer are

[1]For $k > d$, our experimental results are based on the "padding zero" approach.

optimized. This setting is interesting to study as it provides insight on the "capacity" of the model, independent on specific optimization mechanisms.

We will show by experiments that compared with unstructured randomized neural networks, the circulant neural network is faster with the same amount of nodes while keeping similar performance. This surprising result is in line with the recent theoretical/empirical discoveries of using circulant projections on dimensionality reduction [212], and binary embedding [234]. It has been shown that the circulant projection behaves very similarly compared with fully randomized projections in terms of the distance preserving properties. In other words, the randomized circulant projection can be seen as a simulation of the unstructured randomized projection, both of which can capture global information of the data.

In addition, we will show that with the optimizations described in Section 4.5.1, the error rate of the neural networks decreases significantly, meaning that the circulant structure is flexible and powerful enough to be used in a data-dependent fashion.

4.5 Training Circulant Neural Networks

4.5.1 Gradient Computation

The most critical step for optimizing a neural network given a training set is to compute the gradient of the error function with respect to the network weights. Let us consider the conventional neural network with two layers, where the first layer computes the linear projection followed by a nonlinear activation function:

$$h(\mathbf{x}) = \phi(\mathbf{R}\mathbf{x}), \tag{4.3}$$

where $\mathbf{R}$ is an unstructured matrix. We assume the second layer is a linear classifier with weights $\mathbf{w}$. Therefore the output of the two-layer neural network is

$$J(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{R}\mathbf{x}) \tag{4.4}$$

When training the neural network, computing the gradient of the error function involves

computing the gradient of $J(\mathbf{x})$ with respect to each entry of $\mathbf{R}$. It is easy to show that

$$\frac{\partial J(\mathbf{x})}{\partial R_{ij}} = w_i \phi'(\mathbf{R}_{i\cdot}\mathbf{x}) x_j, \quad i = 0, \cdots, d-1, \quad j = 0, \cdots, d-1. \tag{4.5}$$

where $\phi'(\cdot)$ is the derivative of $\phi(\cdot)$.

Note that (4.5) suffices for the gradient-based optimization of neural networks, as the gradient *w.r.t.* networks with more layers can simply be computed with the chain rule, leading to the well-known "back-propagation" scheme.

In the circulant case, we need to compute the gradient of the following objective function:

$$J(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{R}\mathbf{x}) = \sum_{i=0}^{d-1} w_i \phi\left(\mathbf{R}_{i\cdot}\mathbf{x}\right), \quad \mathbf{R} = \mathrm{circ}(\mathbf{r}). \tag{4.6}$$

It is easy to show that

$$\frac{\partial \mathbf{w}^T \phi(\mathbf{R}\mathbf{x})}{\partial r_i} = \mathbf{w}^T \left(\phi'(\mathbf{R}\mathbf{x}) \circ s_{\to i}(\mathbf{x})\right) = s_{\to i}(\mathbf{x})^T (\mathbf{w} \circ \phi'(\mathbf{R}\mathbf{x})) \tag{4.7}$$

$s_{\to i}(\cdot) : \mathbb{R}^d \to \mathbb{R}^d$, right (downwards for a column vector) circularly shifts the vector by one element. Therefore,

$$\begin{aligned} \nabla_{\mathbf{r}} J(\mathbf{x}) =& [s_{\to 0}(\mathbf{x}), s_{\to 1}(\mathbf{x}), \cdots, s_{\to (d-1)}(\mathbf{x})]^T (\mathbf{w} \circ \phi'(\mathbf{R}\mathbf{x})) \\ =& \,\mathrm{circ}(s_{\to 1}(\mathrm{rev}(\mathbf{x})))(\mathbf{w} \circ \phi'(\mathbf{R}\mathbf{x})) \\ =& s_{\to 1}(\mathrm{rev}(\mathbf{x})) \circledast (\mathbf{w} \circ \phi'(\mathbf{r} \circledast \mathbf{x})), \end{aligned} \tag{4.8}$$

where $\mathrm{rev}(\mathbf{x}) = (x_{d-1}, x_{d-2}, \cdots, x_0)$, $s_{\to 1}(\mathrm{rev}(\mathbf{x})) = (x_0, x_{d-1}, x_{d-2}, \cdots, x_1)$. The above uses the same trick of converting the circulant matrix multiplication to circulant convolution. Therefore, computing the gradient takes only $\mathcal{O}(d \log d)$ time with FFT. Training a multi-layer neural network is nothing more than applying (4.7) in each layer with the chain rule.

Note that when $k < d$, we can simply set the last $d - k$ entries of $\mathbf{w}$ in (4.4) to be zero. And when $k > d$, the above derivations can be applied with minimal changes.

4.6 Experiments

We apply our model to three standard datasets in our experiments: MNIST, CIFAR-10, and ImageNet. We note that it is not our goal to obtain state-of-the-art results on these

Method	Train Error	Test Error	Memory (MB)	Testing Time (sec.)
LeNet	0.35%	0.92%	1.56	3.06
Circulant LeNet	0.47%	0.95%	0.27	2.14

Table 4.2: Experimental results on MNIST.

datasets, but rather to provide a fair analysis of the effectiveness of circulant projections in the context of deep neural networks, compared with unstructured projections. Next we describe our implementation and analysis of accuracy and storage costs on these three datasets, followed by an experiment on reduced training set size.

4.6.1 Experiments on MNIST

The MNIST digit dataset contains 60,000 training and 10,000 test images of ten handwritten digits (0 to 9), with 28×28 pixels. We use the LeNet network [124] as our basic CNN model, which is known to work well on digit classification tasks. LeNet consists of a convolutional layer followed by a pooling layer, another convolution layer followed by a pooling layer, and then two fully connected layers similar to the conventional multilayer perceptrons. We used a slightly different version from the original LeNet implementation, where the sigmoid activations are replaced by Rectified Linear Unit (ReLU) activations for the neurons.

Our implementation is extended from Caffe [93], by replacing the weight matrix with the proposed circulant projections with the same dimensionality. The results are compared and shown in Table 4.2. Our fast circulant neural network achieves an error rate of 0.95% on the full MNIST test set, which is very competitive to the 0.92% error rate from the conventional neural network. At the same time, the circulant LeNet is 5.7x more space efficient and 1.43x more time efficient than LeNet.

4.6.2 Experiments on CIFAR

CIFAR-10 is a dataset of natural 32x32 RGB images covering 10-classes with 50,000 images for training and 10,000 for testing. Images in CIFAR-10 vary significantly not only in object position and object scale within each class, but also in object colors and textures.

Method	Train Error	Test Error	Memory (MB)	Testing Time (sec.)
CIFAR-10 CNN	4.45%	15.60%	0.45	4.56
Circulant CIFAR-10 CNN	6.57%	16.71%	0.12	3.92

Table 4.3: Experimental results on CIFAR-10.

Method	Top-5 Error	Top-1 Error	Memory (MB)
Randomized AlexNet	33.5%	61.7%	233
Randomized Circulant CNN 1	35.2%	62.8%	12.5
AlexNet	17.1 %	42.8%	233
Circulant CNN 1	19.4 %	44.1%	12.5
Circulant CNN 2	17.8 %	43.2%	12.7
Reduced-AlexNet	37.2 %	65.3%	12.7

Table 4.4: Classification error rate and memory cost on ILSVRC-2010.

The CIFAR10-CNN network [83] used in our test consists of 3 convolutional layers, 1 fully-connected layer and 1 softmax layer. Rectified linear units (ReLU) are used as the activation units. The circulant CIFAR10-CNN is implemented by adding the circulant weight matrix into the the fully connected layer. Images are cropped to 24x24 and augmented with horizontal flips, rotation, and scaling transformations. We use an initial learning rate of 0.001 and train for 700-300-50 epochs with their default weight decay.

A comparison of the error rates obtained by circulant and unstructured projections is shown in Table 4.3. Our efficient approach based on circulant networks obtains test error of 16.71% on this dataset, compared with 15.60% obtained by the conventional model. At the same time, the circulant networks is 4x more space efficient and 1.2x more time efficient than the conventional CNN.

4.6.3 Experiments on ImageNet (ILSVRC-2010)

ImageNet is a dataset containing over 15 million labeled high-resolution images belonging to roughly 22,000 categories. Starting in 2010, as part of the Pascal Visual Object

Challenge, an annual competition called the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) has been held. A subset of ImageNet with roughly 1000 images in each of 1000 considered categories is used in this challenge. Our experiments were performed on the ILSVRC-2010 dataset.

We use a standard CNN network – "AlexNet" [110] as the building block. The AlexNet consists of 5 convolutional layers, 2 fully-connected layers and 1 final softmax layer. Rectified linear units (ReLU) are used as the activation units. Pooling layers and response normalization layers are also used between convolutional layers. Our circulant network version involves three components: 1) a feature extractor, 2) fully circulant layers, and 3) a softmax classification layer. For 1 and 3 we utilize the Caffe package [93]. For 2, we implement it with Cuda FFT.

All models are trained using mini-batch stochastic gradient descent (SGD) with momentum on batches of 128 images with the momentum parameter fixed at 0.9. We set the initial learning rate to 0.01 and manually decrease the learning rate if the network stops improving as in [110] according to a schedule determined on a validation set. Dataset augmentation is also exploited.

Table 4.4 shows the error rate of various models. We have used two types of structures for the proposed method. Circulant CNN 1 replaces the fully connected layers of AlexNet with circulant layers. Circulant CNN 2 uses "fatter" circulant layers compared with Circulant CNN 1: d of Circulant CNN 2 is set to be 2^{14}. In "Reduced AlexNet", we reduce the parameters size on the fully-connected layer of the original AlexNet to the similar size of our Circulant CNN by cutting d. We have the following observations.

- The performance of Randomized Circulant CNN 1[2] is very competitive to the Randomized AlexNet. This is expected as the circulant projection closely simulates a fully randomized projection (Section 4.3).
- Optimization significantly improves the performance for both unstructured projections and circulant projections. The performance of Circulant CNN 1 is very competitive to AlexNet, yet with a fraction of the space cost.

[2]This is Circulant CNN 1 with randomized circulant projections. In other words, only the convolutional layer is optimized.

d	Full proj.	Circulant proj.	Speedup	Space Saving (in fully connected layer)
2^{10}	2.97	2.52	1.18x	1,000x
2^{12}	3.84	2.79	1.38x	4,000x
2^{14}	19.5	5.43	3.60x	30,000x

Table 4.5: Comparison of training time (ms/per image) and space of full projection and circulant projection. The speedup is defined as the time of circulant projection divided by the time of unstructured projection. Space saving is defined as the space of storing the circulant model by the space of storing the unstructured matrix. The unstructured projection matrix in conventional neural networks takes more than 90% of the space cost. In AlexNet, d is 2^{12}.

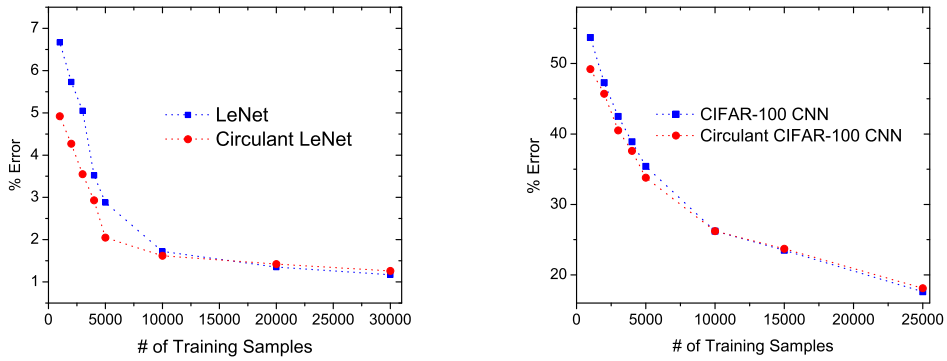
- By tweaking the structure to include more parameters, Circulant CNN 2 further drops the error rate to 17.8%, yet it takes only marginally larger amount of space compared with Circulant CNN 1, an 18x space saving compared with AlexNet.
- With the same memory cost, the Reduced AlexNet performs much worse than Circulant CNN 1.

In addition, one interesting finding is that "dropout", which is widely used in training CNN, does not improve the performance of circulant neural networks. In fact, it increases the error rate from 19.4% (without dropout) to 20.3% (not shown in the figure). This indicates that the proposed method is more immune to over-fitting.

We also show the training time (per image) on standard and the circulant version of AlexNet. We vary the number of hidden nodes d in the fully connected layers and compare the training time until the model converges (ms/per image). Table 4.5 shows the result. Our method provides dramatic space saving, and significant speedup compared with the conventional approach.

4.6.4 Reduced Training Set Size

Compared with the neural network model with unstructured projections, the circulant neural network has fewer parameters. Intuitively, this may bring the benefit of better model



(a) MNIST (b) CIFAR-10

Figure 4.1: Test error when training with reduced dataset sizes of circulant CNN and conventional CNN.

generalization. In other words, the circulant neural network might be less data hungry compared with the conventional neural networks. To verify our assumption, we report the performance of each model under different training sizes on MINST and CIFAR-10 datasets. Figure 4.1 shows test error rate when training on a random selection of 1,000, 2,000, 3,000, 5,000, 10,000, and half of training set. On the MNIST set, to achieve a fixed error rate, the circulant models need less data. On CIFAR-10, this improvement is limited as the circulant layer only occupies a small part of the model.

4.7 Discussions

Fully Connected Layer *vs.* Convolution Layer. The goal of the method developed in this paper is to improve the efficiency of the fully connected layers of neural networks. In convolutional architectures, the fully connected layers are often the bottleneck in terms of the space cost. For example, in "Alexnet", the fully connected layers take 95% of the storage. Remarkably, the proposed method enables dramatic space saving in the fully connected layer (4000x as shown in Table 4.5), making it negligible in memory compared with the convolutional layers. Our discovery resonates with the recent works showing that the fully connected layers can be compressed, or even completely removed [132, 199].

In addition, the fully connected layer costs roughly 20% – 30% of the computation time based on our implementation. The FFT-based implementation can further improve the time cost, though not as impressive as the space saving aspect. Our method is complementary to the works improving the time and space cost of convolutional layers [90, 141, 56, 49].

Circulant Projection *vs.* 2D Convolution. One may notice that although our approach leverages convolutions for speeding up computations, it is fundamentally different from the convolutions performed in CNNs. The convolution filters in CNNs are all small 2D filters aiming at capturing local information of the images, whereas the proposed method is used to replace the fully connected layers, which are often "big" layers capturing global information. The operation involved is large 1D convolutions rather than small 2D convolutions. The circulant projection can be understood as "simulating" an unstructured projection, with much less cost. Note that one can also apply FFT to compute the convolutions on the 2D convolutional layers, but due to the computational overhead, the speed improvement is generally limited on small-scale problems. In contrast, our method can be used to dramatically speed up and scale the processing in fully connected layers. For instance, when the number of input nodes and output nodes are both 1 million, the conventional linear projection is essentially impossible, as it requires TBs of memory. On the other hand, doing a convolution of two 1 million dimensional vector is a light computation task with FFT.

Towards Larger Neural Networks. Currently, deep neural network models usually contain hundreds of millions of parameters. In real world applications, there exist problems which involve an increasing amount of data. We may need larger and deeper networks to learn better representations from large amounts of data. Compared with unstructured projections, the circulant projection significantly reduces the computation and storage cost. Therefore, with the same amount of resources, circulant neural networks can use deeper, as well as larger fully-connected networks. We have conducted preliminary experiments showing that the circulant model can be extended at least 10x deeper than conventional neural networks with the same scale of computational resources.

4.8 Conclusion and Future Works

We proposed to use circulant projections to replace the unstructured projections in order to optimize fully connected layers of neural networks. This dramatically improves the computational complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d \log d)$ and space complexity from $\mathcal{O}(d^2)$ to $\mathcal{O}(d)$. An efficient approach was proposed for optimizing the parameters of the circulant projections. We showed by experiments that this optimization can lead to much faster convergence and training time compared with conventional neural networks. Our experimental analysis was carried out on three standard datasets, showing the effectiveness of the proposed approach. We also reported experiments on randomized circulant projections, achieving performance similar to that of unstructured randomized projections. Our ongoing work includes exploring different matrix structures for circulant neural networks.

Chapter 5

Compact Nonlinear Maps with Circulant Extension

5.1 Introduction

Kernel methods such as the Support Vector Machines (SVMs) [40] are widely used in machine learning to provide nonlinear decision function. The kernel methods use a positive-definite kernel function K to induce an implicit nonlinear map ψ such that $K(\mathbf{x}, \mathbf{y}) = \langle \psi(\mathbf{x}), \psi(\mathbf{y}) \rangle$, $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$. This implicit feature space could potentially be an infinite dimensional space. Fortunately, kernel methods allow one to utilize the power of these rich feature spaces without explicitly working in such high dimensions. Kernel methods are also widely used in solving computer vision problems [120].

Despite their popularity, the kernel machines come with high computational cost due to the fact that at the training time it is necessary to compute a large kernel matrix of size $N \times N$ where N is the number of training points. Hence, the overall training complexity varies from $O(N^2)$ to $O(N^3)$, which is prohibitive when training with millions of samples. Testing also tends to be slow due to the linear growth in the number of support vectors with training data, leading to $O(Nd)$ complexity for d-dimensional vectors.

On the other hand, linear SVMs are appealing for large-scale applications since they can be trained in $O(N)$ time [97, 58, 187] and applied in $O(d)$ time, independent of N. Hence, if the input data can be mapped nonlinearly into a compact feature space explicitly, one

can utilize fast training and testing of linear methods while still preserving the expressive power of kernel methods.

Following this reasoning, kernel approximation via explicit nonlinear maps has become a popular strategy for speeding up kernel machines [171]. Formally, given a kernel $K(\mathbf{x}, \mathbf{y})$, kernel approximation aims at finding a nonlinear map $Z(\cdot)$, such that

$$K(\mathbf{x}, \mathbf{y}) \approx Z(\mathbf{x})^T Z(\mathbf{y}) \tag{5.1}$$

In the computer vision community in specific, different types of nonlinear maps have been developed to approximate intersection kernels [137], additive kernels [208], skewed multiplicative histogram kernels [125] *etc.*

However, there are two main issues with the existing nonlinear mapping methods. Before the kernel approximation, a "good" kernel has to be chosen. Choosing a good kernel is perhaps an even more challenging problem than approximating a known kernel. In addition, the existing methods are designed to approximate the kernel in the whole space independent on the data. As a result, the feature mapping often needs to be high-dimensional in order to achieve low kernel approximation error.

In this chapter, we first propose the Compact Nonlinear Map (CNM), a formulation that optimizes the nonlinear maps directly in a data-dependent fashion. Specifically, we adopt the Random Fourier Feature framework [171] for approximating positive definite shift-invariant kernels. Instead of generating the parameter of the nonlinear map randomly from a distribution, we learn the parameters by minimizing the classification loss based on the training data (Section 5.4). The proposed method can be seen as approximating an "optimal kernel" for the classification task. The method results in significantly more compact maps with very competitive classification performance. As a by-product, the same framework can also be used to achieve compact kernel approximation if the goal is to approximate some predefined kernels (Section 5.4.3). The proposed compact nonlinear maps are fast to learn and compare favorably to the baselines.

To make the method scalable for very high-dimensional data, we then propose to use circulant structured projection matrices under the CNM framework (Section 5.7). This further improves the computational complexity from $\mathcal{O}(kd)$ to $\mathcal{O}(k \log d)$ and the space

complexity from $\mathcal{O}(kd)$ to $\mathcal{O}(k)$, where k is the number of nonlinear maps, and d is the input dimensionality.

5.2 Related Works

Kernel Approximation. Following the seminal work on explicit nonlinear feature maps for approximating positive definite shift-invariant kernels [171], nonlinear mapping techniques have been proposed to approximate other forms of kernels such as the polynomial kernel [103, 168], generalized RBF kernels [195], intersection kernels [137], additive kernels [208], skewed multiplicative histogram kernels [125], and semigroup kernel [224]. Techniques have also been proposed to improve the speed and compactness of kernel approximations by using structured projections [123], better quasi Monte Carlo sampling [223], binary code [239, 147], and dimensionality reduction [80]. Our method in this chapter is built upon the Random Fourier Feature [171] for approximating shift-invariant kernel, a widely used kernel type in machine learning. Besides explicit nonlinear maps, kernel approximation can also be achieved using sampling-based low-rank approximations of the kernel matrices such as the Nystrom method [219, 53, 115]. In order for these approximations to work well, the eigenspectrum of the kernel matrix should have a large gap [225].

Kernel Learning. There have been significant efforts in learning a good kernel for the kernel machines. Works have been proposed to optimize the hyperparameters of a kernel function [30, 105], and finding the best way of combining multiple kernels, *i.e.*, Multiple Kernel Learning (MKL) [10, 7, 66, 41]. A summary of MKL can be found in [70]. Related to our work, methods have been proposed to optimize shift-invariant kernels [13, 67]. Different from the above, the approach in this chapter can be seen as learning an optimal kernel by directly optimizing its nonlinear maps. Therefore, it is a joint kernel approximation and kernel learning.

Fast Nonlinear Models. Besides kernel approximation, there have been other types of works aiming at speeding up kernel machine [24]. Such techniques include decomposition methods [155, 29], sparsifying kernels [2], limiting the number of support vectors [104, 164], and low-rank approximations [63, 11]. Unfortunately, none of the above can be scaled to

truly large-scale data. Another alternative is to consider the local structure of the data to train and apply the kernel machines locally [116, 85, 100, 86]. However, partitioning becomes unreliable in high-dimensional data. Our work is also related to shallow neural networks as we will discuss in a later part of this chapter.

5.3 Random Fourier Features: A Review

We begin by reviewing the Random Fourier Feature method [171], which is widely used in approximating positive-definite shift-invariant kernels. A kernel K is shift-invariant, if $K(\mathbf{x}, \mathbf{y}) = K(\mathbf{z})$ where $\mathbf{z} = \mathbf{x} - \mathbf{y}$. For a function $K(\mathbf{z})$ which is positive definite on $\mathbb{R}^d$, it guarantees that the Fourier transform of $K(\mathbf{z})$,

$$\mathcal{K}(\boldsymbol{\theta}) = \frac{1}{(2\pi)^{d/2}} \int d^d\mathbf{z}\, K(\mathbf{z})\, e^{i\boldsymbol{\theta}^T\mathbf{z}}\,, \tag{5.2}$$

admits an interpretation as a probability distribution. This fact follows from Bochner's celebrated characterization of positive definite functions,

Theorem 5.1. *[22] A function $K \in C(\mathbb{R}^d)$ is positive definite on $\mathbb{R}^d$ if and only if it is the Fourier transform of a finite non-negative Borel measure on $\mathbb{R}^d$.*

A consequence of Bochner's theorem is that the inverse Fourier transform of $\mathcal{K}(\boldsymbol{\theta})$, *i.e.*, $K(\mathbf{z})$, can be interpreted as the computation of an expectation, *i.e.*,

$$\begin{aligned} K(\mathbf{z}) &= \frac{1}{(2\pi)^{d/2}} \int d^d\boldsymbol{\theta}\, \mathcal{K}(\boldsymbol{\theta})\, e^{-i\boldsymbol{\theta}^T\mathbf{z}} \\ &= E_{\boldsymbol{\theta}\sim p(\boldsymbol{\theta})}\, e^{-i\boldsymbol{\theta}^T(\mathbf{x}-\mathbf{y})} \\ &= 2\, E_{\substack{\boldsymbol{\theta}\sim p(\boldsymbol{\theta}) \\ b\sim U(0,2\pi)}} \left[\cos(\boldsymbol{\theta}^T\mathbf{x} + b)\cos(\boldsymbol{\theta}^T\mathbf{y} + b)\right]\,, \end{aligned} \tag{5.3}$$

where $p(\boldsymbol{\theta}) = (2\pi)^{-d/2}\mathcal{K}(\boldsymbol{\theta})$ and $U(0, 2\pi)$ is the uniform distribution on $[0, 2\pi)$. If the above expectation is approximated using Monte Carlo with k random samples $\{\boldsymbol{\theta}_i, b_i\}_{i=1}^k$, then $K(\mathbf{x}, \mathbf{y}) \approx \langle Z(\mathbf{x}), Z(\mathbf{y})\rangle$ with

$$Z(\mathbf{x}) = \sqrt{2/k}\left[\cos(\boldsymbol{\theta}_1^T\mathbf{x} + b_1), ..., \cos(\boldsymbol{\theta}_k^T\mathbf{x} + b_k)\right]^T\,. \tag{5.4}$$

Such Random Fourier Features have been used to approximate different types of positive definite shift-invariant kernels, including the Gaussian kernel, the Laplacian kernel, and the

Cauchy kernel [171]. Despite the popularity and success of Random Fourier Feature, the notable issues for all kernel approximation methods are that:

- Before performing the kernel approximation, a known kernel has to be chosen. This is a very challenging task. As a matter of fact, the classification performance is influenced by both the quality of the kernel, and the error in approximating it. Therefore, better kernel approximation in itself may not lead to better classification performance.
- The Monte-Carlo sampling technique tries to approximate the kernel for *any* pair of points in the entire input space without considering the data distribution. This usually leads to very high-dimensional maps in order to achieve low kernel approximation error everywhere.

In this chapter, we follow the Random Fourier Feature framework. Instead of sampling the kernel approximation parameters $\boldsymbol{\theta}_i$ and b_i from a probability distribution to approximate a known kernel, we propose to optimize them directly with respect to the classification objective. This leads to very compact maps as well as higher classification accuracy.

5.4 The Compact Nonlinear Map (CNM)

5.4.1 The Framework

Consider the following feature maps, and the resulted kernel based on the Random Fourier Features proposed in [171][1]:

$$\hat{K}_{\boldsymbol{\Theta}}(\mathbf{x},\mathbf{y}) = Z(\mathbf{x})^T Z(\mathbf{y}), \quad Z_i(\mathbf{x}) = \sqrt{2/k}\cos(\boldsymbol{\theta}_i^T\mathbf{x}), \quad i = 1, ..., k. \tag{5.5}$$

By representing $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_k]$, we can write $Z(\mathbf{x}) = \cos(\boldsymbol{\Theta}^T\mathbf{x})$, where $\cos(\cdot)$ is the element-wise consine function.

Proposition 5.1. *For any $\boldsymbol{\Theta}$, the kernel function $\hat{K}$, defined as $\hat{K}_{\boldsymbol{\Theta}}(\mathbf{x},\mathbf{y}) = Z(\mathbf{x})^T Z(\mathbf{y})$, is a positive-definite shift-invariant kernel.*

Proof. The shift-invariance follows from the fact that, for any $x, y \in \mathbb{R}$

$$\cos(x)\cos(y) = \frac{\cos(x-y) - \sin(x-y)}{2}, \text{ is a function of } x - y.$$

[1]For simplicity, we do not consider the bias term which can be added implicitly by augmenting the dimension to the feature $\mathbf{x}$.

Algorithm 1 Optimizing $\mathbf{w}$ with fixed $\mathbf{\Theta}$

1: INPUT: initialized $\mathbf{w}$, $||\mathbf{w}|| < 1/\sqrt{\lambda}$.
2: OUTPUT: updated $\mathbf{w}$.
3: **for** $t = 1$ to T_1 **do**
4: Sample M points to get $\mathcal{A}$, and compute the gradient $\nabla_{\mathbf{w}}$.
5: $\mathbf{w} \leftarrow \mathbf{w} - (1/\lambda t)\nabla_{\mathbf{w}}$.
6: $\mathbf{w} \leftarrow \min\{1, 1/\lambda||\mathbf{w}||\}\,\mathbf{w}$.
7: **end for**

The positive definiteness follows from a direct computation and the definition. □

In addition, it has been shown in the Bochner's theorem that such a cosine map can be used to approximate *any* positive shift-invariant kernels. Therefore, if we optimize the "kernel approximation" parameters directly, it can be seen as approximating an optimal positive definite shift-invariant kernel for the task. In this chapter, we consider the task of binary classification using SVM. The proposed approach can be easily extended to other scenarios such as multi-class classification and regression.

Suppose we have N samples with +1/-1 labels as training data $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_N, y_N)$. The Compact Nonlinear Maps (CNM) jointly optimize the nonlinear map parameters $\mathbf{\Theta}$ and the linear classifier $\mathbf{w}$ in a data-dependent fashion.

$$\underset{\mathbf{w},\mathbf{\Theta}}{\operatorname{argmin}} \frac{\lambda}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{N}\sum_{i=1}^{N} L\left(y_i, \mathbf{w}^T Z(\mathbf{x}_i)\right) \tag{5.6}$$

In this chapter, we use the hinge loss as the loss function: $L(y_i, \mathbf{w}^T Z(x_i)) = \max(0, 1 - y_i \mathbf{w}^T Z(x_i))$.

5.4.2 The Alternating Minimization

Optimizing (5.6) is a challenging task. A large number of parameters need to be optimized, and the problem is nonconvex. In this chapter, we propose to find a local solution of the optimization problem with Stochastic Gradient Descent (SGD) in an alternating fashion.

For a fixed $\mathbf{\Theta}$, the optimization of $\mathbf{w}$ is simply the traditional linear SVM learning problem.

$$\underset{\mathbf{w}}{\operatorname{argmin}} \frac{\lambda}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{N}\sum_{i=1}^{N} L\left(y_i, \mathbf{w}^T Z(\mathbf{x}_i)\right). \tag{5.7}$$

We use the Pegasos procedure [187] to perform SGD. In each step, we sample a small set of data points $\mathcal{A}$. The data points with non-zero loss are denoted as $\mathcal{A}_+$. Therefore, the gradient can be written as

$$\nabla_{\mathbf{w}} = \lambda\mathbf{w} - \frac{1}{|\mathcal{A}|}\sum_{(\mathbf{x},y)\in\mathcal{A}_+} y\cos(\mathbf{\Theta}^T\mathbf{x}). \tag{5.8}$$

Each step of the Pegasos procedure consists of gradient descent and a projection step. The process is summarized in Algorithm 1.

For a fixed $\mathbf{w}$, optimizing $\mathbf{\Theta}$ becomes

$$\underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{N}\sum_{i=1}^{N} L\left(y_i, \mathbf{w}^T Z(\mathbf{x}_i)\right). \tag{5.9}$$

We also preform SGD with sampled mini-batches. Let the set of sampled data points be $\mathcal{A}$, the gradient can be written as

$$\nabla_{\boldsymbol{\theta}_i} = \frac{w_i}{|\mathcal{A}|}\sum_{(\mathbf{x},y)\in\mathcal{A}_+} y\sin(\boldsymbol{\theta}_i^T\mathbf{x})\mathbf{x}, \tag{5.10}$$

where $\mathcal{A}_+$ is the set of samples with non-zero loss, and w_i is the i-th element of $\mathbf{w}$. The process is summarized in Algorithm 2.

The overall algorithm is shown in Algorithm 3. The sampled gradient descent steps are repeated to optimize $\mathbf{w}$ and $\mathbf{\Theta}$ alternatively. We use a $\mathbf{\Theta}$ obtained from sampling the Gaussian distribution (same as Random Fourier Feature) as initialization.

5.4.3 CNM for Kernel Approximation

In the previous sections, we presented the Compact Nonlinear Maps (CNM) optimized to achieve low classification error. This framework can also be used to achieve compact kernel approximation. The idea is to optimize with respect to kernel approximation error. For example, given a kernel function K, we can minimize $\mathbf{\Theta}$ in terms of the MSE on the

Algorithm 2 Optimizing $\mathbf{\Theta}$ with fixed $\mathbf{w}$

1: INPUT: initialized $\mathbf{\Theta}$.
2: OUTPUT: updated $\mathbf{\Theta}$.
3: **for** $t = 1$ to T_2 **do**
4: Sample M points to get $\mathcal{A}$, and compute the gradient $\nabla_{\mathbf{\Theta}}$.
5: $\mathbf{\Theta} \leftarrow \mathbf{\Theta} - (1/\lambda t)\nabla_{\mathbf{\Theta}}$.
6: **end for**

Algorithm 3 The Compact Nonlinear Map (CNM)

1: Initialize $\mathbf{\Theta}$ as the Random Fourier Feature.
2: Choose $\mathbf{w}$ such that $||\mathbf{w}|| < 1/\sqrt{\lambda}$.
3: **for** iter= 1 to T **do**
4: Perform T_1 SGD (Pegasos [187]) steps to optimize $\mathbf{w}$, shown in Algorithm 1.
5: Perform T_2 SGD steps with to optimize $\mathbf{\Theta}$, shown in Algorithm 2.
6: **end for**

training data:

$$\underset{\mathbf{\Theta}}{\operatorname{argmin}} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(K(\mathbf{x}_i, \mathbf{x}_j) - Z(\mathbf{x}_i)^T Z(\mathbf{x}_j) \right)^2. \tag{5.11}$$

This can be used to achieve more compact kernel approximation by considering the data under consideration. Note that the ultimate goal of a nonlinear map is to improve the classification performance. Therefore, this section should be viewed only as a by-product of the proposed method.

For the optimization, we can also perform SGD similar to the former section. Let $\mathcal{A}$ be the set of random samples, we only need to compute the gradient in terms of $\mathbf{\Theta}$:

$$\nabla_{\boldsymbol{\theta}_i} = \frac{8}{k} \sum_{\mathbf{x}, \mathbf{x}' \in \mathcal{A}} \left(\mathcal{K}(\mathbf{x}, \mathbf{x}') - \frac{2}{k} \cos(\mathbf{\Theta}^T \mathbf{x})^T \cos(\boldsymbol{\theta}^T \mathbf{x}') \right) \sin(\boldsymbol{\theta}_i^T \mathbf{x}) \cos(\boldsymbol{\theta}_i^T \mathbf{x}') \mathbf{x}_i. \tag{5.12}$$

5.5 Discussions

We presented Compact Nonlinear Maps (CNM) with an alternating optimization algorithm for the task of binary classification. CNM can be easily adapted to other tasks

such as regression and multi-class classification. The only difference is that the gradient computation of the algorithm needs to be changed. We provide a brief discussion regarding adding regularization, and the relationship of CNM to neural networks.

Regularization. One interesting fact is that the cos function has an infinite VC dimension. In the proposed method, with a fixed $\mathbf{w}$, if we only optimize $\mathbf{\Theta}$ with SGD, the magnitude of $\mathbf{\Theta}$ will grow unbounded, and this will lead to near-perfect training accuracy, and obviously, overfitting. Therefore, a regularizer over $\mathbf{\Theta}$ should lead to better performance. We have tested different types of regularizations of $\mathbf{\Theta}$ such as the Frobenius norm, and the ℓ_1 norm. Interestingly, such a regularization could only marginally improve the performance. It appears that early stopping in the alternating minimization framework provides reasonable regularization in practice on the tested datasets.

CNM as Neural Networks. One can view the proposed CNM framework from a different angle. If we ignore the original motivation of the work *i.e.*, kernel approximation via Random Fourier Features, the proposed method can be seen as a *shallow* neural network with one hidden layer, with $\cos(\cdot)$ as the activation function, and the SVM objective. It is interesting to note that such a "two-layer neural network", which simulates certain shift-invariant kernels, leads to very good classification performance as shown in the experimental section. Under the neural network view, one can also use back-propagation as the optimization method, similar to the proposed alternating SGD, or use other types of activation functions such as the sigmoid, and ReLU functions. However the "network" then will no longer correspond to a shift-invariant kernel.

5.6 Experiments

We conduct experiments using 6 UCI datasets summarized in Table 5.1. Four of them are image datasets (`USPS`, `MNIST`, `CIFAR`, `LETTER`). The size of the mini batches in the optimization is empirically set as 500. The number of SGD steps in optimizing $\mathbf{\Theta}$ and $\mathbf{w}$ is set as 100. We find that satisfactory classification accuracy can be achieved within a few hundred iterations. The bandwidth of the RBF kernel in classification experiments, and the kernel approximation experiments is set to be $\gamma = 2/\sigma^2$, where σ is the average distance to

Table 5.1: 8 UCI datasets used in the experiments

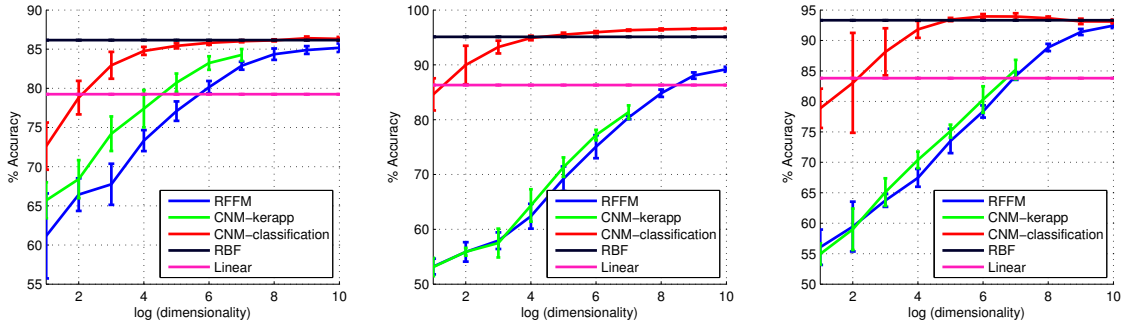
Dataset	# Training Samples	# Testing Samples	Dimensionality
USPS	7,291	2,007	256
MNIST	60,000	10,000	784
CIFAR	50,000	10,000	400
FOREST	522,910	58,102	54
LETTER	12,000	6,000	16
MAGIC04	14,226	4,795	10

the 50th nearest neighbor estimated from 1,000 samples of the dataset. Further fine tuning of γ may lead to even better performance.

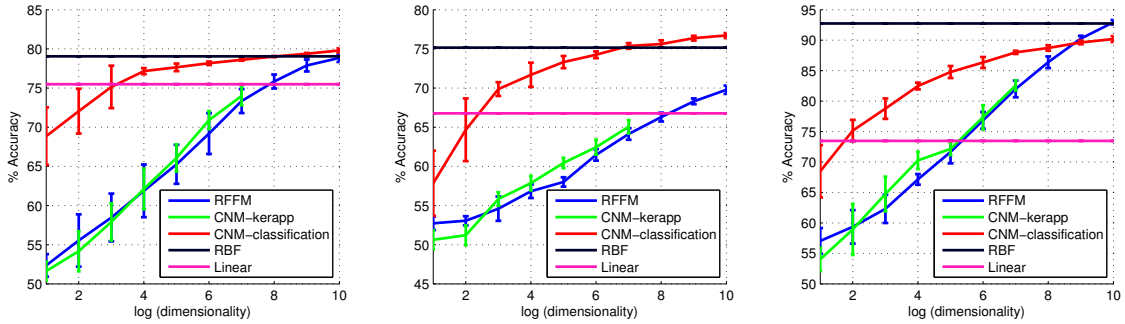
5.6.1 CNM for Classification

Figure 5.1 shows the classification accuracies. CNM-classification is the proposed method. We compare it with three baselines: linear SVM based on the original features (Linear), kernel SVM based on RBF (RBF), and the Random Fourier Feature method (RFFM). As shown in the figures, all the datasets are not linearly separable, as the RBF SVM performance is much better than the linear SVM performance.

- For all the datasets, CNM is much more compact than the Random Fourier Feature to achieve the same classification accuracy. For example, on the USPS dataset, to get 90% accuracy, the dimensionality of CNM is 8, compared with 512 of RFFM, a 60x improvement.
- As the dimensionality k grows, accuracies of both the RFFM and CNM improve, with the RFFM approaching the RBF performance. In a few cases, the CNM performance can be even higher than the RBF performance. This is due to the fact that CNM is "approximating" an optimal kernel, which could be better than the fixed RBF kernel.



(a) MAGIC04 (b) MNIST (c) USPS

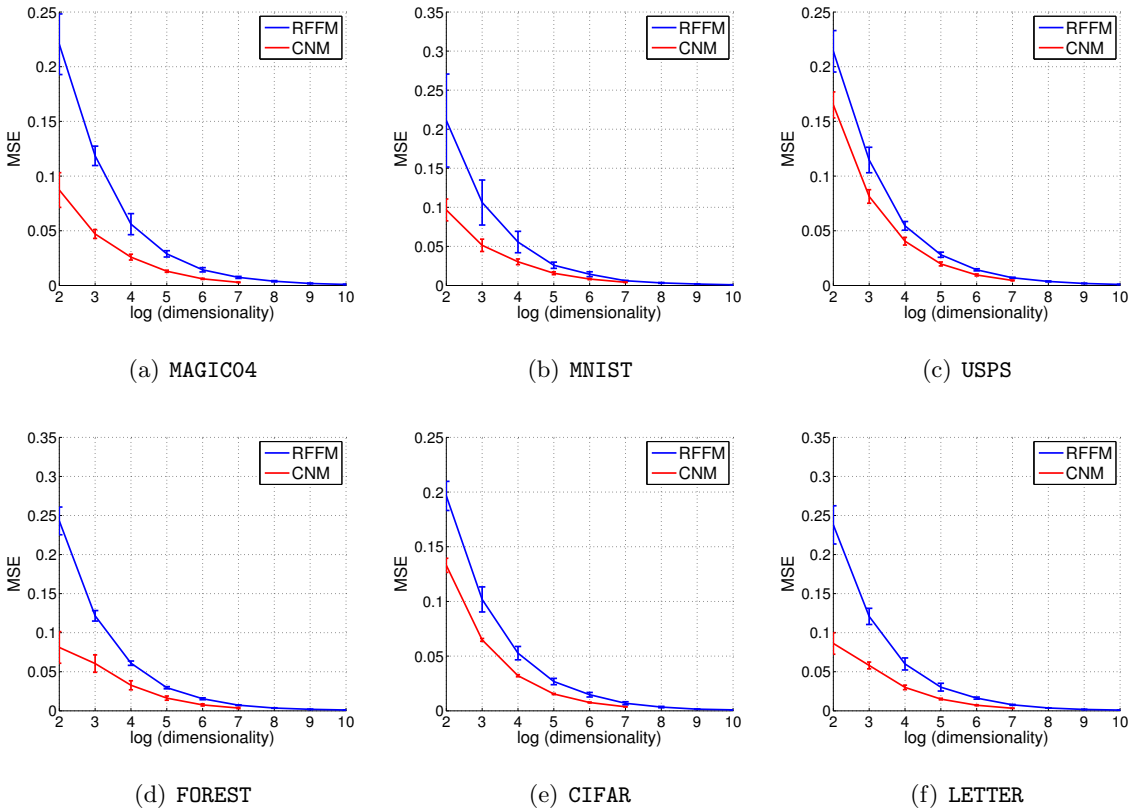


(d) FOREST (e) CIFAR (f) LETTER

Figure 5.1: Compact Nonlinear Map (CNM) for classification. RFFM: Random Fourier Feature Map based on RBF kernel. CNM-kerapp: CNM for kernel approximation (Section 5.4.3). CNM-classification: CNM for classification (Section 5.4). RBF: RBF kernel SVM. Linear: linear SVM based on the original feature.

5.6.2 CNM for Kernel Approximation

We conduct experiments on using the CNM framework to approximate a known kernel (Section 5.4.3). The kernel approximation performance (measured by MSE) is shown in Figure 5.2. CNM is computed with dimensionality up to 128. For all the datasets, CNM achieves more compact kernel approximations compared with the Random Fourier Features. We further use such features in the classification task. The performance is shown as the green curve (CNM-kerapp) in Figure 5.1. Although CNM-kerapp has lower MSE in kernel approximation than RFFM, its accuracy is only comparable or marginally better than RFFM. This verifies the fact that better kernel approximation may not necessarily lead to



(a) MAGIC04 (b) MNIST (c) USPS

(d) FOREST (e) CIFAR (f) LETTER

Figure 5.2: Compact Nonlinear Map (CNM) for kernel approximation. RFFM: Random Fourier Feature based on RBF kernel. CNM-kerapp: CNM for kernel approximation (Section 5.4.3).

better classification.

5.7 Circulant Extension

Kernel approximation with nonlinear maps comes with an advantage that SVM can be trained in $\mathcal{O}(N)$, and evaluated in $\mathcal{O}(k)$ time, leading to scalable learning and inference in terms of the number of samples. In this chapter, we have presented CNM where the projection matrix of the Random Fourier Features is optimized to achieve high classification performance. For d-dimensional inputs and k-dimensional nonlinear maps, the computational and space complexities of both CNM and RFFM are $\mathcal{O}(kd)$. CNM comes with the advantage that k can be much smaller than that for RFFM to achieve a similar

performance. One observation from Section 5.6 is that though CNM can lead to much more compact maps, it still has better performance when higher-dimensional maps are used. In many situations, it is required that the number of nonlinear map k is comparable to the feature dimension d. This will lead to both space and computational complexities $\mathcal{O}(d^2)$, which is not suitable for high-dimensional datasets. One natural question to ask is whether it is possible to improve further the scalability in terms of the input dimension d.

In this section, we show that by imposing the *circulant structure* on the projection matrix, one can achieve similar kernel approximation performance compared with the fully randomized matrix. The proposed approach reduces the computational complexity to $\mathcal{O}(k \log d)$, and the space complexity to $\mathcal{O}(k)$, when $k \geq d$.

5.7.1 Circulant Nonlinear Maps

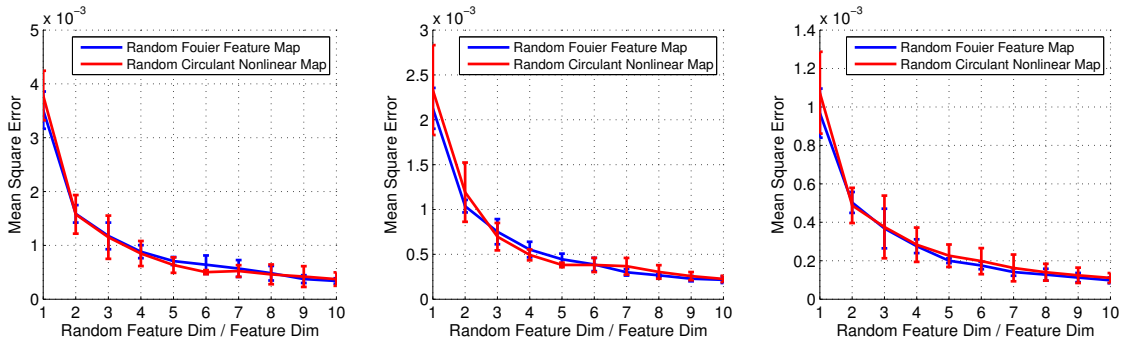
Following Section 2.3, for $\mathbf{x} \in \mathbb{R}^d$, its d-dimensional circulant nonlinear map is defined as:

$$Z(\mathbf{x}) = \sqrt{2/d} \cos(\mathbf{RDx}), \quad \mathbf{R} = \text{circ}(\mathbf{r}), \tag{5.13}$$

where $\mathbf{D}$ is a diagonal matrix with each diagonal entry being a Bernoulli variable (± 1 with probability $1/2$). Same as the former chapters, we omit the $\mathbf{D}$ matrix in the following discussion. Following the analysis in Section 2.3, the proposed approach has time complexity $\mathcal{O}(d \log d)$. Following Section 2.3, when $k < d$, we can still use the circulant matrix $\mathbf{R} \in \mathbb{R}^{d \times d}$ with d parameters, but the output is set to be the first k elements in (5.13). When $k > d$, we use multiple circulant projections, and concatenate their outputs. This gives the computational complexity $\mathcal{O}(k \log d)$, and space complexity $\mathcal{O}(k)$. Note that in such case, (5.13) should be normalized by $\sqrt{2/k}$ instead of $\sqrt{2/d}$.

5.7.2 Randomized Circulant Nonlinear Maps

Similar to the Random Fourier Features, one can generate the parameters of the circulant projection, *i.e.*, the elements of vector $\mathbf{r}$ in (5.13), by random sampling from a Gaussian distribution. We term such a method randomized circulant nonlinear maps. Figure 5.3 shows the kernel approximation MSE of the randomized circulant nonlinear maps and compares



(a) USPS (b) CIFAR (c) MNIST

Figure 5.3: MSE of Random Fourier Feature, and randomized circulant nonlinear map.

it with the Random Fourier Features. Although with much lower computational and space complexities, it is interesting that the circulant nonlinear map can achieve almost identical MSE compared with the Random Fourier Features.

5.7.3 Optimized Circulant Nonlinear Maps

Following the CNM framework, one can optimize the parameters in the projection matrix to improve the performance using alternating minimization procedure with the classification objective. The step to optimize classifier parameters $\mathbf{w}$ is the same as described in section 5.4.2. The parameters of the projection are now given by circulant matrix $\mathbf{R}$. Thus, the step of optimizing $\mathbf{R}$ requires computing the gradient *w.r.t.* each element of vector $\mathbf{r}$ as:

$$\frac{\partial \mathbf{w}^T \cos(\mathbf{R}\mathbf{x})}{\partial r_i} = -\mathbf{w}^T \left(\sin(\mathbf{R}\mathbf{x}) \circ s_{\rightarrow i}(\mathbf{x})\right) = -s_{\rightarrow i}(\mathbf{x})^T(\mathbf{w} \circ \sin(\mathbf{R}\mathbf{x})), \tag{5.14}$$

where $s_{\rightarrow i}(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^d$, circularly (downwards) shifts the vector $\mathbf{x}$ by one element. Therefore,

$$\begin{aligned} \nabla_{\mathbf{r}}(\mathbf{w}^T \cos(\mathbf{R}\mathbf{x})) &= -[s_{\rightarrow 0}(\mathbf{x}), s_{\rightarrow 1}(\mathbf{x}), \cdots, s_{\rightarrow (d-1)}(\mathbf{x})]^T(\mathbf{w} \circ \sin(\mathbf{R}\mathbf{x})) \\ &= -\operatorname{circ}(s_{\rightarrow 1}(\operatorname{rev}(\mathbf{x})))(\mathbf{w} \circ \sin(\mathbf{R}\mathbf{x})) \\ &= -s_{\rightarrow 1}(\operatorname{rev}(\mathbf{x})) \circledast (\mathbf{w} \circ \sin(\mathbf{r} \circledast \mathbf{x})), \end{aligned} \tag{5.15}$$

where $\operatorname{rev}(\mathbf{x}) = (x_{d-1}, x_{d-2}, \cdots, x_0)$, $\quad s_{\rightarrow 1}(\operatorname{rev}(\mathbf{x})) = (x_0, x_{d-1}, x_{d-2}, \cdots, x_1)$.

The above uses the same trick of converting the circulant matrix multiplication to circulant convolution, and this has also been used in the circulant neural networks (Chapter

Dataset (k)	Random Fourier Feature	Circulant-random	Circulant-optimized
USPS (d)	89.05 ± 0.65	89.40 ± 1.02	$\mathbf{91.96 \pm 0.45}$
USPS ($2d$)	91.90 ± 0.29	91.87 ± 0.11	$\mathbf{93.08 \pm 0.96}$
MNIST (d)	91.33 ± 0.05	91.01 ± 0.03	$\mathbf{92.73 \pm 0.21}$
MNIST ($2d$)	92.95 ± 0.42	93.22 ± 0.30	$\mathbf{94.11 \pm 0.24}$
CIFAR (d)	69.14 ± 0.64	65.21 ± 0.18	$\mathbf{71.17 \pm 0.68}$
CIFAR ($2d$)	$\mathbf{71.15 \pm 0.28}$	68.56 ± 0.70	71.11 ± 0.46

Table 5.2: Classification accuracy (%) using circulant nonlinear maps. The randomized circulant nonlinear maps have similar performance as of the Random Fourier Features but with significantly reduced storage and computation time. Optimization of circulant matrices tends to further improve the performance.

4). Therefore, computing the gradient of $\mathbf{r}$ takes only $\mathcal{O}(d \log d)$ time. The classification accuracy on three datasets with relatively large feature dimensions is shown in Table 5.2. The randomized circulant nonlinear maps give similar performance to that from the Random Fourier Features but with much less storage and computation time. Optimization of circulant matrices tends to further improve the performance.

5.8 Conclusion and Future Works

We have presented Compact Nonlinear Maps (CNM), which are motivated by the recent works on kernel approximation that allow very large-scale learning with kernels. This work shows that instead of using randomized feature maps, learning the feature maps directly, even when restricted to shift-invariant kernel family, can lead to substantially compact maps with similar or better performance. The improved performance can be attributed mostly to simultaneous learning of kernel approximation along with the classifier parameters. This framework can be seen as a shallow neural network with a specific nonlinearity (cosine) and provides a bridge between two seemingly unrelated streams of works. To make the proposed approach more scalable for high-dimensional data, we further introduced an extension, which imposes the circulant structure on the projection matrix. This improves the computation complexity from $\mathcal{O}(kd)$ to $\mathcal{O}(k \log d)$ and the space complexity from $\mathcal{O}(kd)$ to

$\mathcal{O}(k)$, where d is the input dimension, and k is the output map dimension.

In the future, it will be interesting to explore if the complex data transforms captured by multiple layers of a deep neural network can be captured by learned nonlinear maps while remaining compact with good training and testing efficiency.

Part II

Learning from Label Proportions

Chapter 6

Learning from Label Proportions

6.1 Introduction

The scalability of learning algorithms is often fundamentally limited by the amount of supervision available. For the massive visual data, it is difficult, and sometimes impossible to collect sufficient amount of conventional supervised information, such as labels on the images, and detailed annotations on the videos. On the other hand, the massive visual data often comes with some weak forms of supervision, such as group-level labels, or label statistics on the groups. The natural question to ask is whether one can utilize such weak supervision in machine learning. For example, in recognition of video events, only the event labels on the video level (a group of frames) are given – can we learn a model to pinpoint the frames in which the event actually happens? In modeling attributes, only some semantic similarities between a set of known categories and a set of attributes are provided – can we leverage such information to model the attributes? Conventional learning algorithms are not designed to incorporate such forms of supervision.

In this part of the thesis, we address machine learning with supervision provided on the group level. To incorporate such types of supervision, we study a learning setting called Learning from Label Proportions (LLP), where the training data is provided in groups, or "bags", and only the proportion of each class in each bag is known. The task is to learn a model to predict the class labels of the individuals. In particular, we study LLP under a binary setting. Figure 6.1 illustrates a toy example of LLP.

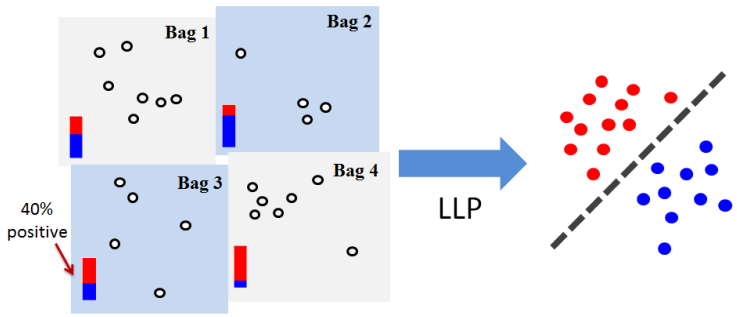


Figure 6.1: Illustration of learning from label proportions (LLP). In this examples, the training data is provided in 4 bags, each with its label proportions. The learned model is a separating hyperplane to classify the individual instances.

This learning setting can be adapted in solving various computer vision problems mentioned above. Besides computer vision, LLP also has broad applications in social science, marketing, healthcare and computer vision. For example, after election, the proportions of votes of each demographic area are released by the government. In healthcare, the proportions of diagnosed diseases of each ZIP code area are available to the public. Is it possible to learn a model to predict the individual labels based on only the group-level label proportions? As non-confidential attributes of individuals can be easily acquired from census survey, digital footprint, shopping history *etc.*, LLP leads to not only promising new applications, but also serious privacy concerns as releasing label proportions may result in the discovery of sensitive personal information.

Learning from label proportion has recently received attentions from the machine learning community. For example, Musicant et al. [149] formulated the problem of learning from aggregative values, with special focus on adapting a few different algorithms under the regression setting. Quadrianto et al. [169] proposed a solution by estimating the mean of each class. Rüeping [179] proposed treating the mean of each bag as a "super-instance", which was assumed to have a soft label corresponding to the label proportion. We will provide a more comprehensive review of the related works in Chapter 7 and Chapter 8.

We note that although this problem has been studied before with various types of algorithms, the previous works lack the fundamental understanding of when and why learning

from label proportion is possible. In the thesis, we provide our answers with an intuitive framework called Empirical Proportion Risk Minimization (EPRM) (Chapter 7). Under such a framework, we provide positive results on the sample complexity of such a learning setting. Motivated from EPRM, we further propose the $\propto$SVM algorithm as an extension of the classic SVM (Chapter 8). We show that the algorithm and its extensions can be successfully applied in computer vision applications, including video event detection (Chapter 9), and attribute learning (Chapter 10).

6.2 Related Works

We begin by reviewing the related learning settings in this section. Note that the limitation of supervised information can also be alleviated by methods such as transfer learning, and more efficient ways of collecting supervision, *e.g.*, by crowdsourcing. We will review those methods in Part III since they are more related to the proposed attribute-based approaches.

6.2.1 Semi-Supervised Learning

Semi-supervised learning addresses the problem of machine learning with limited supervised information. We refer the reader to Zhu [241] for a literature survey. Considering the classification task, for example, semi-supervised learning usually studies the cases of training a classifier with a small set of labeled samples, and a large set of unlabeled samples. The question of interest is whether the unlabeled samples can be used to improve the learning process. There are many ways of addressing semi-supervised learning, including generative models [138], self-training [227], co-training and multiview training [21], transductive learning [96], and graph-based methods [19]. Due to the strong connection to real-world scenario, semi-supervised learning has been applied in solving many computer vision problems including large-scale image search [215], tracking [76], objective detection [178], and face recognition [26]. Due to the difference in the learning setting, semi-supervised learning methods cannot be directly applied to solving learning from label proportions. Yet, the proposed $\propto$SVM algorithm is motivated from the transductive SVM [96] (Chapter 8).

6.2.2 Multiple Instance Learning

In learning with weak supervision, the "weakness" comes not only from the quantity but also from the format of the supervision. In many scenarios, the learning algorithm has only access to labels on the groups (or "bags") of instances. One extensively studied learning setting is *Multiple Instance Learning* (MIL) [50]. In MIL, the learner has access to bags with their labels generated by the Boolean OR operator on the unobserved instance labels, *i.e.*, a bag is positive *iff* it has at least one positive instance. The task is to learn a binary predictor for the *bags*. The two most popular algorithms for MIL are mi-SVM and MI-SVM [5]. The first algorithm emphasizes searching max-margin hyperplanes to separate positive and negative instances, while the second algorithm selects the most representative positive instance for each positive bag and uses it for classification. In computer vision, MIL has been applied in scene classification [139], content-based image retrieval [240], and image classification [35]. In the thesis, the theoretical analysis of learning from label proportions is inspired by the analysis of MIL [181, 180] (Chapter 7). We also use MIL as the baseline method in the video event detection task (Chapter 9).

6.3 Overview of the Proposed Approaches

6.3.1 On Empirical Proportion Risk Minimization (EPRM)

As learning with label proportions is very different from conventional supervised learning, it is important to understand when and why such learning is possible. We have conducted an analysis to answer the above question, and to understand how the parameters, such as group size and group proportions, affect the performance of the new learning paradigm. In Chapter 7, we introduce an intuitive framework, Empirical Proportion Risk Minimization (EPRM). EPRM learns an instance label classifier to match the given label proportions on the training data. Our result is based on a two-step analysis. First, we provide a VC bound on the generalization error of the bag proportions. We show that the bag sample complexity is only mildly sensitive to the bag size. Second, we show that under some mild assumptions, good bag proportion prediction guarantees good instance label prediction. The results together provide a formal guarantee that the individual labels

can indeed be learned in the LLP setting. We also demonstrate the feasibility of LLP based on a case study in real-world setting: predicting income based on census data. The study provides not only theoretical support on the feasibility of learning with label proportions, but also guidance on designing new algorithms, and protecting privacies when releasing group statistics. This work was originally presented in [233].

6.3.2 The proportion-SVM ($\propto$SVM) Algorithm

Based on the theoretical analysis in Chapter 7, we propose a new method called proportion-SVM, or $\propto$SVM, which explicitly models the latent unknown instance labels in a large-margin framework (Chapter 8). Unlike the existing works, our approach avoids making restrictive assumptions about the data. The $\propto$SVM model leads to a non-convex integer programming problem. In order to solve it efficiently, we propose two algorithms: one based on simple alternating optimization and the other based on a convex relaxation. Experiments on standard datasets show that $\propto$SVM outperforms the state-of-the-art. This work was originally presented in [231].

6.3.3 Applications in Computer Vision

There are many challenging problems in vision, where the supervised information is only provided at group level, and the task is to learn a model to classify the individuals. We have applied the proposed LLP tools in solving two of them.

Video Event Recognition by Discovering Discriminative Visual Segments (Chapter 9). In video event detection, usually only the video-level event labels are given. Can we learn a model to localize the event in each video? In this work, we propose an instance-based video event detection approach based on LLP. We treat each video as a bag, each with multiple instances, defined as video segments of different temporal intervals. Different from LLP, for each bag we do not know the exact proportion. Our key assumption is that the positive videos have a large number of positive instances while negative videos have few ones. The $\propto$SVM algorithm is then applied in this setting. Experiments on large-scale video event datasets demonstrate significant performance gains. The proposed method is also useful in explaining the detection results by localizing the temporal segments in a

video which is responsible for the positive detection. The work was originally presented in [118].

Attribute Modeling from Category-Attribute Proportions (Chapter 10). Attribute-based representation has been widely used in visual recognition and retrieval due to its interpretability and cross-category generalization properties. However, classic attribute learning requires manually labeling attributes on the images, which is very expensive, and not scalable. In this work, we propose to model attributes from category-attribute proportion. The proposed framework can model attributes without attribute labels on the images. Specifically, given a multi-class image datasets with M categories, we model an attribute, based on an M-dimensional category-attribute proportion vector, where each element of the vector loosely characterizes the proportion of images in the corresponding category having the attribute. The attribute learning can then be formulated as an LLP problem, where images of one category form a bag. We show that the category-attribute proportions can be estimated from multiple modalities such as human commonsense knowledge, NLP tools, and other domain knowledge. The value of the proposed approach is demonstrated by various applications including modeling animal attributes, visual sentiment attributes, and scene attributes. The work was originally presented in [232].

Chapter 7

On Empirical Proportion Risk Minimization

7.1 Introduction

This chapter studies when and why individual labels can be learned from label proportions, by analyzing a general framework, namely *Empirical Proportion Risk Minimization* (EPRM). EPRM optimizes the instance-level classifier to minimize the empirical proportion loss. In other words, it tries to find an instance hypothesis to match the given label proportions. The main contribution is a formal guarantee that under some mild assumptions, the individual instance labels can be recovered (learned), with the EPRM framework. Specifically, we provide a two-step analysis.

Our first result bounds the generalization error of bag proportions by the empirical bag proportion error (Section 7.5). We show that the sample complexity is only mildly sensitive to the bag size. In other words, given enough training bags, it is possible to learn a bag proportion predictor, which generalizes well to unseen bags. This conclusion in itself is interesting as in some applications we are simply interested in getting good proportion estimates for bags: doctors may want to predict the rate of disease on certain geographical area, and companies may want to predict attrition rate of certain department.

Second, we show that under some mild conditions, the instance label error can be controlled by the bag proportion error (Section 7.6). In other words, "good" bag proportion

predictions imply "good" instance label predictions. This finding is more crucial, as it enables interesting applications, and from the privacy protection point of view, the ability to learn a good instance label predictor given label proportions is of concern. Finally, we demonstrate the feasibility of LLP in a case study: predicting income based on census data (Section 7.8).

7.2 Related Works

In their seminal work, Quadrianto et al. [169] proposed to estimate the mean of each class using the mean of each bag and the label proportions. These estimates are then used in a conditional exponential model to maximize the log likelihood. The algorithm is under a restrictive assumption that the instances are conditionally independent given the label. Rüeping [179] proposed to use a large-margin regression method by assuming the mean instance of each bag having a soft label corresponding to the label proportion. As an extension to multiple-instance learning, Kuck and de Freitas [111] designed a hierarchical probabilistic model to generate consistent label proportions. Similar ideas have also been shown in [33] and [149]. Different from the above works, this paper provides theoretical results addressing when and why bag proportion and instance labels can be learned. Our result is independent of the algorithms.

A related, yet more extensively studied learning setting is *Multiple Instance Learning* (MIL) [50]. In MIL, the learner has access to bags, with their labels generated by the Boolean OR operator on the unobserved instance labels, *i.e.*, a bag is positive *iff* it has at least one positive instance. The task is to learn a binary predictor for the *bags*. It has been shown that if all the instances are drawn *iid* from a single distribution, MIL is as easy as learning from *iid*. instances with one-sided label noise [20]. In real-world applications, the instances inside each bag can have arbitrary dependencies or a manifold structure. The learnability and sample complexity results in the above scenarios are given by [181, 180], and [9], respectively. In this chapter, we use the tools in [180] to analyze the generalization error of bag proportions. More importantly, we show that under some conditions, a good bag proportion predictor implies a good instance label predictor.

$\mathcal{X}$	Domain of the instances
$\mathcal{Y}$	Domain of the labels, $\{-1, 1\}$
$\mathbf{x}$	Instance (the feature of an instance)
B	Bag. $\{\mathbf{x}\|\mathbf{x} \in B\}$ are all the instances inside
$y(\mathbf{x})$	Ground-truth label of $\mathbf{x}$
$\bar{y}(B)$	Ground-truth label proportion of B
$h(\mathbf{x})$	Predicted label of $\mathbf{x}$ based on hypothsis (classifier) h
$\bar{h}(B)$	Predicted label proportion of B based on h
$\mathcal{H}$	Instance label hypothesis class
$\bar{\mathcal{H}}$	Bag label proportion hypothesis class
L	Loss function on the label proportion
$\mathcal{D}$	Distribution of bags
$er_S^L(h)$	Empirical proportion error on S with L and h
$er_{\mathcal{D}}^L(h)$	Expected proportion error on D with L and h
M	The number of bags in training

Table 7.1: Key Notations of Chapter 7.

7.3 The Learning Setting: Learning from Label Proportions

The notation of the learning setting is shown in Table 7.1. Denote by $\mathcal{X}$ the domain of instance attributes, and denote by $\mathcal{Y} = \{-1, 1\}$ the domain of the instance labels. We use $\mathbf{x} \in \mathcal{X}$ as an instance, and $y(\mathbf{x}) \in \mathcal{Y}$ as the binary ground-truth label[1] of $\mathbf{x}$. A bag B is defined as a set of instances[2]. For simplicity, we assume that the bags are of the same size r. Our result can be easily generalized to bags with variable sizes as described in Section 7.7.

The label proportion of bag B is defined as:

$$\bar{y}(B) = \frac{1}{r} \sum_{\mathbf{x} \in B} \frac{y(\mathbf{x}) + 1}{2}. \tag{7.1}$$

[1]Strictly speaking, the label may not be a function of $\mathbf{x}$. In such case, one should define a separate label for each instance. Note that the analysis will remain the same.

[2]Or a multiset if there are duplicates.

In training, for a bag B, the learner receives B, and $\bar{y}(B)$. Note that $y(\mathbf{x}), \mathbf{x} \in B$, the group-truth labels, are not observed by the learner. Let the training set received by the learner be M bags S. For each bag $B \in S$. We use $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$ to denote a hypothesis class on the instances. The learning task is to find an $h \in \mathcal{H}$ which gives low prediction error for *unseen instances* generated by the above process.

In the conventional supervised learning, where all the labels of the training instances are known, a popular framework is the *Empirical Risk Minimization* (ERM), *i.e.* finding the instance hypothesis $h \in \mathcal{H}$ to minimize the empirical instance label error. In LLP, however, the instance labels are not available at the training time. Therefore, we can only try to find $h \in \mathcal{H}$ to minimize the empirical proportion error.

7.4 The Framework: Empirical Proportion Risk Minimization

Definition 7.1. *For $h \in \mathcal{H}$, and a bag B of size r define an operator to predict bag proportion based on the instances $\bar{h}$,*

$$\bar{h}(B) = \frac{1}{r} \sum_{\mathbf{x} \in B} \frac{h(\mathbf{x}) + 1}{2}.$$

And defined the hypothesis class on the bags $\bar{\mathcal{H}} := \{\bar{h} | h \in \mathcal{H}\}$.

The *Empirical Proportion Risk Minimization* (EPRM) selects the instance label hypothesis $h \in \mathcal{H}$ to minimize the empirical bag proportion loss on the training set S. It can be expressed as follows.

$$\operatorname*{argmin}_{h \in \mathcal{H}} \sum_{B \in S} L\left(\bar{h}(B), \bar{y}(B)\right) \tag{7.2}$$

Here, L is a loss function to compute the error of the predicted proportion $\bar{h}(B)$, and the given proportion $\bar{y}(B)$. In this chapter, we assume that L is 1-Lipschitz in $\xi := \bar{h}(B) - \bar{y}(B)$. EPRM is a very general framework for LLP. One immediate question is whether the instance labels can be learned by EPRM. In the following sections, we provide affirmative results. We first bound the generalization error of bag proportions based on the empirical bag proportions. We show that the sample complexity of learning bag proportions is only mildly

sensitive to the bag size (Section 7.5). We then show that, under some mild conditions, instance hypothesis h which can achieve low error of bag proportions with high probability, is guaranteed to achieve low error on instance labels with high probability (Section 7.6).

7.5 Generalization Error of Predicting the Bag Proportions

Given a training set S and a hypothesis $h \in \mathcal{H}$, denote by $er_S^L(h)$ the empirical bag proportion error with a loss function L, and denote by $er_{\mathcal{D}}^L(h)$ the generalization error of bag proportions with a loss function L over distribution $\mathcal{D}$:

$$er_S^L(h) = \frac{1}{|S|}\sum_{B\in S} L(\bar{h}(B), \bar{y}(B)), \quad er_{\mathcal{D}}^L(h) = \mathbb{E}_{B\sim\mathcal{D}} L(\bar{h}(B), \bar{y}(B)). \tag{7.3}$$

In this section, we show that good proportion prediction is possible for unseen bags. Note that learning the bag proportion is basically a regression problem on the bags. Therefore, without considering the instance label hypothesis, for a smooth loss function L, the generalization error of bag proportions can be bounded in terms of the empirical proportion error and some complexity measure, *e.g.*, fat shattering dimension [6], of the hypothesis class on the bags. Unfortunately, the above does not provide us insights into LLP, as it does not utilize the structure of the problem. As we show later, such structure is important in relating the error of bag proportion to the error of instance labels.

Based on the definitions in Section 7.3, our main intuition is that the complexity (or "capacity") of bag proportion hypothesis class $\bar{\mathcal{H}}$ should be dependent on the complexity of the instance label hypothesis class $\mathcal{H}$. Formally, we adapt the MIL analysis in [181, 180], to bound the *covering number* [6] of $\bar{\mathcal{H}}$, by the covering number of $\mathcal{H}$. As in our case $\mathcal{H}$ is a binary hypothesis class, we further bound the covering number of $\mathcal{H}$ based on its *VC-dimension* [207]. This leads to the following theorem on the generalization error of learning bag proportions.

Theorem 7.1. *For any $0 < \delta < 1$, $0 < \epsilon < 1$, $h \in \mathcal{H}$, with probability at least $1-\delta$, $er_D^L(h) \leq er_S^L(h) + \epsilon$, if*

$$M \geq \frac{64}{\epsilon^2}(2VC(\mathcal{H})\ln(12r/\epsilon) + \ln(4/\delta)),$$

in which $VC(\mathcal{H})$ is the VC dimension of the instance label hypothesis class $\mathcal{H}$, and M is the number of training bags, and r is the bag size.

The proof is given in the appendix. From the above, the generalization error of bag proportions can be bounded by the empirical proportion error if there are a sufficient number of bags in training. Note that the sample complexity (smallest sufficient size of M above) grows in terms of the bag size. This is intuitive as larger bags create more "confusions". Fortunately, the sample complexity grows at most logarithmically with r. It means that the generalization error is only mildly sensitive to r. We also note that the above theorem generalizes to the well-known result of binary supervised learning, as shown below.

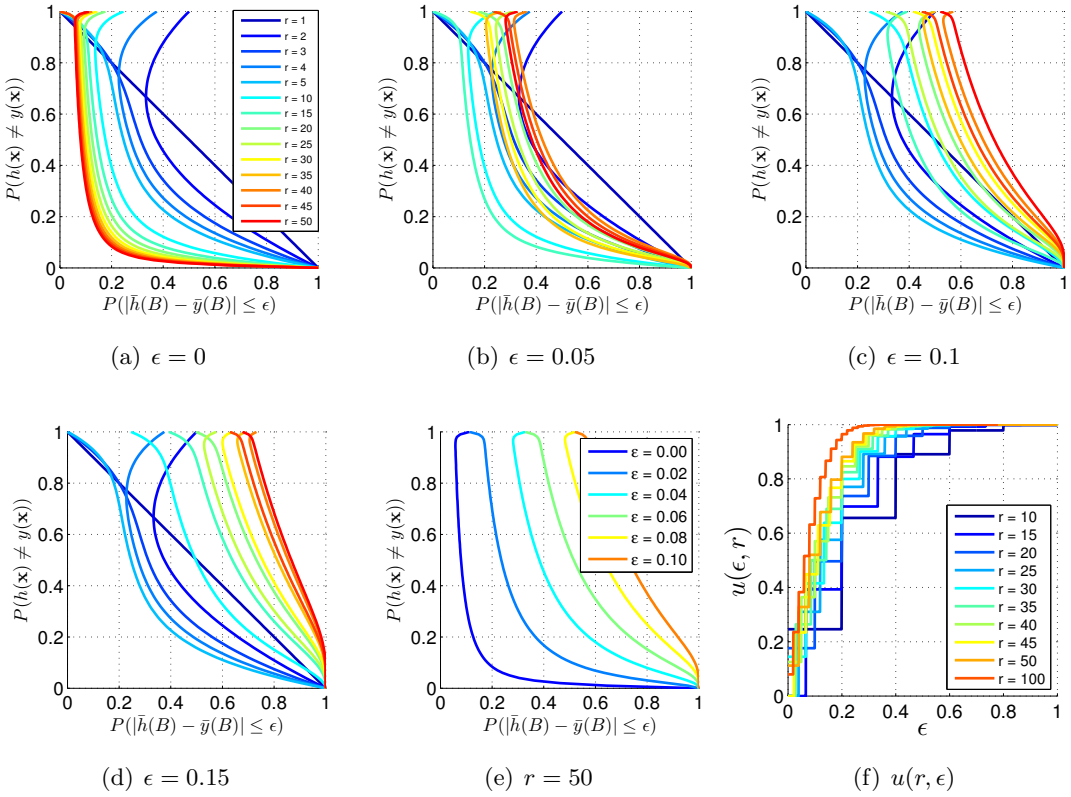
Corollary 7.1. *When bag size $r = 1$, for any $0 < \delta < 1$, $0 < \epsilon < 1$, $h \in \mathcal{H}$, with probability at least $1 - \delta$, $er_D^L(h) \leq er_S^L(h) + \epsilon$, if*

$$M \geq \frac{64}{\epsilon^2}(2VC(\mathcal{H}) \ln(12/\epsilon) + \ln(4/\delta)).$$

7.6 Bounding the Instance Label Error by Bag Proportion Error

From the analysis above, we know that the generalization error of bag proportions can be bounded. The result is without any additional assumptions other than that the bags are *iid*. In this section, based on assumptions on the instances, and the proportions, we present results bounding the error of predicting instance labels by the error of predicting bag proportions. In specific, Section 7.6.1 considers the simple case when all instances are drawn *iid* from a distribution. Section 7.6.2 generalizes to a more general case in which instances are conditionally independent given the bag. Section 7.6.3 justifies the intuition that "pure" bags (bags with proportions close to 0 or 1) are easier. We also discuss the limitations of LLP under our framework, providing insights into protecting the instance labels when releasing label proportions.

The analysis in this section is based on the assumption that we already have an instance hypothesis h, which predicts the proportions well on a bag B: $\mathbb{P}\left(|\bar{h}(B) - \bar{y}(B)| \leq \epsilon\right) \geq 1-\delta$ with some small $0 < \epsilon, \delta < 1$. From Section 7.5, the above is true when we have a sufficient



(a) $\epsilon = 0$ (b) $\epsilon = 0.05$ (c) $\epsilon = 0.1$

(d) $\epsilon = 0.15$ (e) $r = 50$ (f) $u(r, \epsilon)$

Figure 7.1: (a)-(e): Relationship of the probability of making wrong label prediction, $\mathbb{P}(h(\mathbf{x}) \neq y(\mathbf{x}))$, and the probability of bag proportion prediction error is small than ϵ, $\mathbb{P}\left(|\bar{h}(B) - \bar{y}(B)| \leq \epsilon\right)$, under the assumption that the instances are drawn *iid*, and the prior can be matched by the hypothesis, *i.e.*, $\mathbb{P}(h(\mathbf{x}) = 1) = \mathbb{P}(y(\mathbf{x}) = 1)$. $\mathbb{P}(h(\mathbf{x}) \neq y(\mathbf{x}))$ is a monotonically decreasing function of $\mathbb{P}\left(|\bar{h}(B) - \bar{y}(B)| \leq \epsilon\right)$, if $\mathbb{P}\left(|\bar{h}(B) - \bar{y}(B)| \leq \epsilon\right) \in (u(r, \epsilon), 1]$. $u(r, \epsilon)$ is shown in (f). Larger r and larger ϵ will result in larger $u(r, \epsilon)$.

number of training bags, and a good algorithm to achieve small empirical bag proportion error. Formally, from Theorem 7.1, suppose we have a hypothesis h, such that $er^L_{\mathcal{D}}(h) \leq \epsilon'$. Then $\mathbb{P}_{B \sim \mathcal{D}}(|\bar{h}(B) - \bar{y}(B)| \leq \epsilon'')$ for some other small ϵ'' can also be bounded. With Markov's inequality: for any $0 < \delta < 1$, define $\epsilon'' = \epsilon'/\delta$, then $\mathbb{P}_{B \sim \mathcal{D}}(|\bar{h}(B) - \bar{y}(B)| \leq \epsilon'') \geq 1 - \delta$.

7.6.1 Instances Are Generated *IID*

Let's first consider a special case, where *all* the instances are drawn *iid* over a distribution of the instances. In addition, we assume that the prior of the instances can be matched by

the hypothesis, *i.e.*, $\mathbb{P}(h(\mathbf{x}) = 1) = \mathbb{P}(y(\mathbf{x}) = 1)$. This assumption is not too restrictive, as the small empirical bag proportion error already implies that the priors are approximately matched, and for most learning models, we can adjust the hypothesis by a bias term to match the empirical prior estimated on the training data.

Proposition 7.1. *Assuming the instances are generated* iid*, and* $\mathbb{P}(h(\mathbf{x}) = 1) = \mathbb{P}(y(\mathbf{x}) = 1)$*,* $\mathbb{P}\left(\left|\bar{h}(B) - \bar{y}(B)\right| \leq \epsilon\right)$ *can be expressed analytically as a function of* $\mathbb{P}(h(\mathbf{x}) \neq y(\mathbf{x}))$*. Let* $\beta = \mathbb{P}(h(\mathbf{x}) \neq y(\mathbf{x}))$*:*

$$\mathbb{P}\left(\left|\bar{h}(B) - \bar{y}(B)\right| \leq \epsilon\right) = \theta_1^r \sum_{i=0}^{r} \binom{r}{i} \theta_2^i \Big(\mathcal{Q}(i + \lfloor \epsilon r \rfloor; r - i, \theta_2) - \mathcal{Q}(i - \lfloor \epsilon r \rfloor - 1; r - i, \theta_2) \Big),$$

where $\lfloor \cdot \rfloor$ *is the floor operator,* $\theta_1 = (2 - \beta)/2$*,* $\theta_2 = \beta/(2 - \beta)$*,* $0 < \beta < 1$*,* $0 < \epsilon < 1$ *and* $\mathcal{Q}$ *is the CDF of binomial distribution.*

As what we actually want is to bound $\mathbb{P}(h(\mathbf{x}) \neq y(\mathbf{x}))$ based on $\mathbb{P}\left(\left|\bar{h}(B) - \bar{y}(B)\right| \leq \epsilon\right)$, we show the "inverse" function in Figure 7.1. From the curves, we see that β is a monotonically decreasing function of $\mathbb{P}\left(\left|\bar{h}(B) - \bar{y}(B)\right| \leq \epsilon\right)$, when $\mathbb{P}\left(|\bar{h}(B) - \bar{y}(B)| \leq \epsilon\right) \in (u(r, \epsilon), 1]$. $u(r, \epsilon)$ is shown in Figure 7.1 (f). In other words, the instance label error can be controlled by the bag proportion error, when the later is small. The results are of course the "tightest" under the above assumptions.

7.6.2 Instances Are Conditionally Independent Given Bag

One important observation from Figure 7.1 is that the curves are independent of $\mathbb{P}(y(\mathbf{x}) = 1)$. Therefore, the analytical results can be directly applied to the case when the instances are conditionally independent given the bag. A lot of real-world applications follow this assumption. For example, to model the voting behavior, each bag is generated by randomly sampling a number of individuals from a certain location. It is reasonable to assume that the individuals are *iid* given the location. Assume that the bags are generated from $\mathcal{D}$, a "mixture" of multiple components $\mathcal{D}_1, ..., \mathcal{D}_T$, where each $\mathcal{D}_i$ is also a distribution over bags. We consider the process of drawing a bag from $\mathcal{D}$ as firstly picking a distribution $\mathcal{D}_i$ and then generating a bag from $\mathcal{D}_i$. We assume for $\mathcal{D}_i$, $i = 1, \cdots, T$, there exists an instance distribution $\mathcal{D}'_i$, such that generating a bag from $\mathcal{D}_i$ is by drawing r *iid* instances from $\mathcal{D}'_i$,

and $\mathbb{P}_{\mathbf{x}\sim\mathcal{D}_i'}(y(\mathbf{x})=1) = \mathbb{P}_{\mathbf{x}\sim\mathcal{D}_i'}(h(\mathbf{x})=1)$. In such case, Proportion 7.1 and all the analysis in Section 7.6.1 can be directly applied.

7.6.3 Learning with Pure Bags

Intuitively, if the bags are very "pure", *i.e.*, their proportions are either very low or very high, the instance label error can be well controlled by the bag proportion error. The case that all the bags are with proportions 0 or 1 should be the easiest, as it is identical to the conventional supervised learning setting. We justify the intuition in this section. Different from Section 7.6.1 and Section 7.6.2, no generative assumptions are used.

Definition 7.2. *For $0 < \eta < 1$, we say that a bag is* $(1-\eta)$-pure *if at least a fraction $(1-\eta)$ of all instances have the same label.*

Based on the definition above, our results are summarized as below.

Proposition 7.2. *Let h be a hypothesis satisfying $\mathbb{P}_{B\sim\mathcal{D}}(|\bar{h}(B) - \bar{y}(B)| \leq \epsilon) \geq 1-\delta$ for some $0 < \epsilon, \delta < 1$. Assume that the probability that a bag is* $(1-\eta)$-pure *is at least $1-\rho$ for some $0 < \eta, \rho < 1$. Then for a bag, with probability at least $(1-\delta-\rho)$, h classifies correctly at least $r(1-2\eta-\epsilon)$ instances.*

Proposition 7.3. *There exists a distribution $\mathcal{D}$ over all bags of size r and a learner h such that $\bar{h}(B) = \bar{y}(B)$, each bag is* $(1-\eta)$-pure*, but h misclassifies a fraction 2η instances of each bag.*

Proposition 7.2 is shown to be tight based on Proposition 7.3. Proposition 7.3 also shows limitations of LLP. It is interesting to consider an extreme case, where all the bags are with label proportion 50% (they are the least "pure"). Then there exists a hypothesis h which can achieve zero bag proportion prediction error, yet with 100% instance label error. In other words, it is hopeless to learn or recover the instance labels. The result provided in this section is interesting in two ways. On the one hand, it provides us a failure case, in which even perfect bag proportion prediction does not guarantee low error on the instance labels. On the other hand, it provides guidance when releasing sensitive information. For example, it might be safer for the curator to release bags with proportions closer to 50%, compared with those with proportions closer to 0% or 100%.

7.7 Discussions

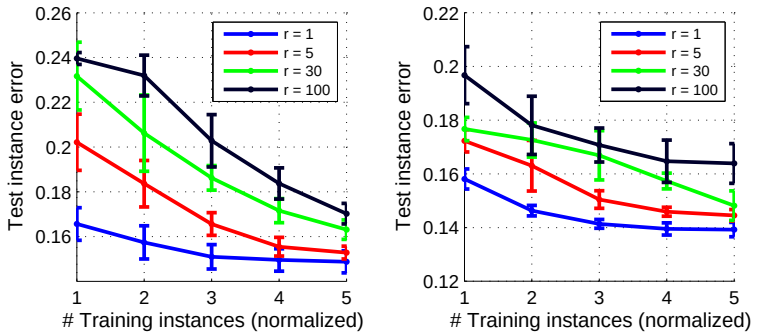
Extending the Results to Variable Bag Size. For simplicity in our analysis, we assumed that all the bags were of size r. In fact, our results also hold for variable bag size. For the result on the bag proportion error, Theorem 7.1 holds immediately by replacing r with the *average bag size in training* $\bar{r}$. This is based on the fact that the covering number bound holds (shown in the proof of Theorem 7.1 in the appendix) with average bag size [180]. For results on the instance label error, our results in Section 7.6.3 and Section 7.6.2 hold for any bag size.

Learning with Population Proportions. Considering the scenario of modeling voting behavior based on government released statistics: the government releases the population proportion (*e.g.* 62.6% in New York voted for Obama in 2012 election) of each location, and we only have a subset of randomly sampled instances for each location. Can LLP be applied to correctly predict labels of the individuals? In such a scenario, EPRM can only minimize the proportion error in terms of the population proportions because the actual proportions of the sampled subsets are not available. We can assume that a bag is formed by randomly sampling r instances from a location with a true (population) proportion p^*, which is released. The Chernoff bound ensures that the sampled proportion is concentrated to the released population proportion: *when* $r \geq \ln(2/\delta)/(2\epsilon^2)$*, with probability at least* $1-\delta$, $|\bar{y}(B) - p^*| \leq \epsilon$. Therefore, with enough training bags, and enough samples per bag, the generalization error can be bounded.

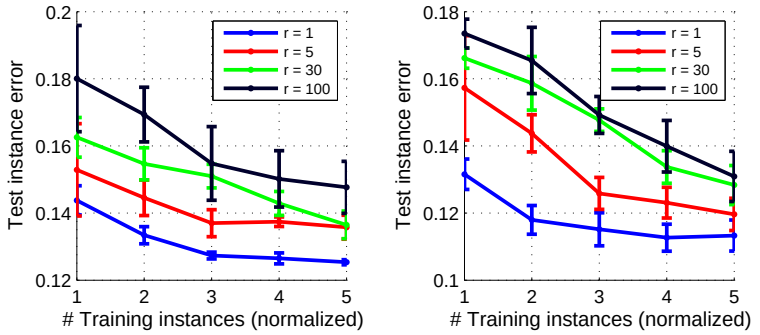
7.8 A Case Study: Predicting Income based on Census Data

So far we have provided formal guarantees that under some conditions, labels of individual instances can be learned or recovered. In this section, we conduct a case study to demonstrate the feasibility of LLP on real-world data. The task is to predict individual income based on census data, and label proportions. We use a dataset which covers a subset of the 1994 census data[3]. It contains 32,561 instances (individual persons), each with 123 binary attributes about education, marital status, sex, occupation, working hours per

[3] `http://archive.ics.uci.edu/ml/datasets/Adult`



(a) IID (b) Occupation



(c) Education (d) Race

Figure 7.2: Predicting income based on census data. (a) All the instances are *iid*. (b)-(d) The instances are conditionally independent given the bag, with the title of each figure as its grouping attributes. The number of training instances is equal-spaced in log-scale, with the smallest number 500, and the largest number 50,000.

week *etc*. Each individual has a binary label indicating whether his/her income $> 50k$. We consider this label as sensitive information, for which we only know its proportions on some bags of people. Next we will show the feasibility of LLP based on different ways of forming the bags. We first divide the dataset to use 80% of the instances for training, and 20% for testing. For all the experiments below, the bags are formed on the 80% training data. The proportion of each training bag is computed based on the ground-truth labels. The experiments are based on $\propto$SVM [231] with linear kernel, and ℓ_1 loss. We use the algorithms proposed in [179] and [169] as the initialization methods for the alter-$\propto$SVM solver. The parameters, $C' \in [0.1, 1, 10]$, $C \in [0.01, 0.1, 1]$, are tuned on the bag proportion prediction

Grouping Attribute	Native Country	Education	Occupation	Relationship	Race
Number of Bags	41	16	15	5	5
$\propto$SVM Error %	18.75 $\pm$ 0.25	19.61 $\pm$ 0.10	18.19 $\pm$ 0.16	18.59 $\pm$ 0.82	24.02 $\pm$ 0.15
Baseline Error %	24.02 $\pm$ 0.56	22.29 $\pm$ 0.55	24.28 $\pm$ 0.28	24.19 $\pm$ 0.72	24.28 $\pm$ 0.40

Table 7.2: Error on predicted income based on the census in a real-world setting.

error by performing cross validation. We report mean and standard deviation of error on the test data based on 5 runs.

All instances are drawn *iid*. We first simulate a simple case where all the instances are drawn *iid* from a distribution of instances. We achieve this by assuming that all the instances of the dataset form a "distribution" of individuals. For all the bags (with r instances), each instance is drawn by randomly sampling the whole training set with replacement. Figure 7.2 (a) shows the experimental results: more instances (bags) lead to lower test error. The relatively low test error demonstrates the feasibility of LLP under such settings. Fewer training bags (and larger training bag size) generally result in larger test error variance, as the algorithm is more likely to converge to worse local solutions.

Instances are conditionally independent given bag. We simulate the case by a "hierarchical bag generation" process. The individuals are first grouped by a *grouping attribute*. For example, if the grouping attribute is "occupation", the individuals are groups into 15 groups, corresponding to the 15 occupations. Each group is assigned a prior, which is simply uniform in this experiment. To generate a bag of r instances, we first pick a group based on the priors and then perform random sampling with replacement r times in the selected group. Figure 7.2 (b)-(d) show the experiment results. The trend of the learning curves is the same as in Figure 7.2 (a), showing that LLP is indeed possible in such a setting.

A challenging real-world scenario. For real-world applications, the bags are predefined rather than "randomly" generated. In this experiment, we simply group the individuals defined by different grouping attributes. Different from the setting in the previous section, the whole group is treated as a single bag, without further sampling process. Table 7.2 shows the performance of LLP. The "Baseline" result is formed by the following: a new instance is predicted positive if the training bag with the same grouping attribute has

proportion larger than 50%; otherwise it is predicted as negative. For example, it predicts a person with elementary education as negative, as the label proportion of elementary education group is less than 50%. For most experiments, this result is similar to assigning +1 to all test instances because most training bags are with proportion larger than 50%. This scheme provides performance gain for "education", as the label proportion for individuals with low education level is also low. We find that for most grouping attributes, the performance of LLP significantly improves over the baseline. This is also true for some cases where the number of bags is very small. We do observe that for a certain way of forming the bags, *e.g.*, by Race, the improvement is quite limited. This is due to the fact that the instance distributions of the bags are very similar, and, therefore, most of the bags are redundant for the task.

7.9 Conclusion and Future Works

This chapter proposed a novel two-step analysis to answer the question whether the individual labels can be learned when only the label proportions are observed. We showed how parameters such as bag size and bag proportion affected the bag proportion error and instance label error. Our first result shows that the generalization error of bag proportions is only mildly sensitive to the size of the bags. Our second result shows that under different mild assumptions, a good bag proportion predictor guarantees a good instance label predictor. We have also demonstrated the feasibility of LLP based on a case study.

As the future works, we are extending the analysis to cover the multi-class case. Data dependent measure, *e.g.*, Rademacher complexity [12], may lead to tighter bound for practical use. Some alternative tools, *e.g.*, sample complexity results of learning $\{0, ..., n\}$-valued functions [81], can be used for analyzing the generalization error of bag proportions. The failure cases of LLP worth further study as they can be utilized to protect sensitive personal information when releasing label proportions.

We are also extending the work to an active setting, where the learner can actively query a group-level label statistic, in addition to the instance labels. This can be seen as extending combinatorial group testing [54] into a learning scenario. This research direction also has

strong connection with active learning [186], and compressed sensing [52]. In addition, we are studying the learning setting from a privacy-preserving perspective. A preliminary discussion can be found in [233].

Chapter 8

The ∝SVM Algorithm

8.1 Introduction

In the previous section, we proposed the Empirical Proportion Risk Minimization (EPRM) framework which is shown to be able to recover the individual labels under the LLP setting based on mild assumptions. EPRM tries to find a classifier on the individuals to match the proportions on the bags provided in training. In this Chapter, we propose the ∝SVM[1] algorithm, which is motivated from EPRM.

∝SVM explicitly models the unknown instance labels as latent variables. It jointly optimizes the instance labels as well as the classification model based on the known label proportions (Section 8.3). In order to solve ∝SVM efficiently, we propose two algorithms - one based on simple alternating optimization (Section 8.4), and the other based on a convex relaxation (Section 8.5). We show that our approach outperforms the existing methods for various datasets and settings (Section 8.6). We begin by reviewing related algorithms in Section 8.2.

8.2 Related Works

Quadrianto et al. [169] proposed a theoretically sound method to estimate the mean of each class using the mean of each bag and the label proportions. These estimates are then

[1] ∝ is the symbol for "proportional-to".

used in a conditional exponential model to maximize the log likelihood. The key assumption in MeanMap is that the class-conditional distribution of data is independent of the bags. Unfortunately, this assumption does not hold for many real world applications. For example, in modeling voting behaviors, in which the bags are different demographic regions, the data distribution can be highly dependent on the bags. Very recently, Patrini et al. [162] proposed an improved version of MeanMap. It drops the conditional independence assumption and uses a Laplacian regularizer and an alternating minimization to estimate the mean of each class.

Rüeping [179] proposed treating the mean of each bag as a "super-instance", which was assumed to have a soft label corresponding to the label proportion. The "super-instances" can be poor in representing the properties of the bags. Our work also utilizes a large-margin framework, but we explicitly model the instance labels. Section 8.3.3 gives a detailed comparison with InvCal. Figure 8.1 provides a toy example to highlight the problems with MeanMap and InvCal, which are the state-of-the-art methods.

In semi-supervised learning, Mann and McCallum [138] and Bellare et al. [14] used an expectation regularization term to encourage model predictions on the unlabeled data to match the given proportions. Similar ideas were also studied in the generalized regularization method [68]. Li et al. [129] proposed a variant of semi-supervised SVM to incorporate the label mean of the unlabeled data. Unlike semi-supervised learning, the learning setting we are considering requires no instance labels for training. As an extension of multiple-instance learning, Kuck and de Freitas [111] designed a hierarchical probabilistic model to generate consistent label proportions. Besides the inefficiency in optimization, the method was shown to be inferior to MeanMap [169]. Similar ideas have also been studied by Chen et al. [33] and Musicant et al. [149]. Stolpe and Morik [196] proposed an evolutionary strategy paired with a labeling heuristic for clustering with label proportions. Different from clustering, the proposed ∝SVM framework jointly optimizes the latent instance labels and a large-margin classification model. The ∝SVM formulation is related to large-margin clustering [221], with an additional objective to utilize the label proportions. Specifically, the convex relaxation method we used is inspired by the works of Li et al. [129] and Xu et al. [221].

N	The number of instances
$\mathbf{x}_i$	Feature of the i-th instance
$y(\mathbf{x}_i)$	Ground-truth label of the i-th instance
B_k	k-th bag (a set of instances)
$\mathcal{B}_k$	Indices of instances in the k-th bag
y_i	The latent label of $\mathbf{x}_i$
$\mathbf{y}$	$\mathbf{y} = (y_1, \cdots, y_M)$
C	Weight of the label proportion loss
C'	Weight of the instance label loss
L'	Loss function on the instance label

Table 8.1: Notations in Addition to Table 7.1.

8.3 The ∝SVM Framework

8.3.1 Learning Setting

We introduce subindex of the feature $\mathbf{x}$ and the bag B in this chapter to better present the algorithm. The training set $\{\mathbf{x}_i\}_{i=1}^N$ is given in the form of M bags, $B_1, \cdots, B_M$. We use $\mathcal{B}_k$ to denote the subindices for the k-th bag:

$$B_k = \{\mathbf{x}_i | i \in \mathcal{B}_k\}, \quad k = 1, \cdots, M. \tag{8.1}$$

We assume that the bags are disjoint, *i.e.*, $\mathcal{B}_k \cap \mathcal{B}_l = \varnothing$, $\forall k \neq l$. Note that this assumption is only introduced for clearer presentation of the algorithm. The approach can be easily used to handle the case when the bags are overlapped. In that case, one can simply duplicate the instance for all the bags containing it. Following the notation of Chapter 7, the k-th bag is with label proportion $\bar{y}(B_k)$:

$$\bar{y}(B_k) = \frac{1}{|\mathcal{B}_k|} \sum_{i \in \mathcal{B}_k} \frac{y(\mathbf{x}_i) + 1}{2}. \tag{8.2}$$

in which $y(\mathbf{x}_i) \in \mathcal{Y}$ denotes the *unknown* ground-truth label of $\mathbf{x}_i$, $\forall_{i=1}^N$.

8.3.2 Formulation

Following the EPRM framework proposed in Chapter 7, the goal is to find an instance label classifiers $h, h(\mathbf{x}) \in \mathcal{Y}$, such that it is compatible with the given label proportions.

$$\operatorname*{argmin}_{h\in\mathcal{H}} \Psi(h) + C\sum_{k=1}^{M} L(\bar{h}(B_k), \bar{y}(B_k)), \tag{8.3}$$

where $\Psi(h)$ controls the capacity of the classifier, and the loss term controls the "compatibility" between the predicted proportion and the given proportion of the training bags. Note that $\Psi(h)$ is required to improve the generalization power of the classifier. For example, it could be the VC dimension of h as analyzed in Chapter 7.

In this chapter, we consider using the widely used large-margin framework: $h(x) = \text{sign}(\mathbf{w}^T\varphi(\mathbf{x})+b)$, where $\varphi(\cdot)$ is a map of the input data, and $\Psi(h) = \frac{1}{2}\mathbf{w}^T\mathbf{w}$. Unfortunately, the above problem then becomes very difficult to be optimized due to the non-convex and discrete nature of the loss term. We instead solve a slightly different problem, which has very strong connections to the conventional large-margin method. The idea lies in explicitly modeling the instance labels, and jointly solving both the instance labels as well the classification model.

We model the unknown instance labels as $\mathbf{y} = (y_1, \cdots, y_N)^T$, in which $y_i \in \mathcal{Y}$ denotes the unknown label of $\mathbf{x}_i$, $i = 1, \cdots, N$. Thus the label proportion of the k-th bag can be modeled as

$$p_k(\mathbf{y}) = \frac{\sum_{i\in\mathcal{B}_k} y_i}{2|\mathcal{B}_k|} + \frac{1}{2}. \tag{8.4}$$

We formulate the ∝SVM under the large-margin framework as below[2].

$$\operatorname*{argmin}_{\mathbf{y},\mathbf{w},b} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C'\sum_{i=1}^{N} L'(y_i, \mathbf{w}^T\varphi(\mathbf{x}_i) + b) + C\sum_{k=1}^{M} L\left(p_k(\mathbf{y}), \bar{y}(B_k)\right), \tag{8.5}$$

in which L' is the loss function for classic supervised learning. L is a function to penalize the difference between the true label proportion and the estimated label proportion based on $\mathbf{y}$. The task is to simultaneously optimize the labels $\mathbf{y}$ and the model parameters $\mathbf{w}$ and b. The above formulation permits using different loss functions for L' and L. One can also add weights for different bags. Throughout this paper, we consider L' as the hinge loss, which is

[2]The constraint, $y_i \in \{-1, 1\}$, $i = 1, \cdots, N$, is omitted throughout this chapter to simplify the notation.

widely used for large-margin learning: $L'(y_i, \mathbf{w}^T\varphi(\mathbf{x}_i)+b) = \max\big(0, 1-y_i(\mathbf{w}^T\varphi(\mathbf{x}_i)+b)\big)$. We consider L as the absolute loss: $L(p_k(\mathbf{y}), \bar{y}(B_k)) = |p_k(\mathbf{y}) - \bar{y}(B_k)|$. Additionally, we have the following remarks for the proportion ∝SVM.

- When C' is set to be very large, $y_i \equiv h(\mathbf{x}_i)$. Thus $p_k(\mathbf{y}) = \bar{h}(B_k)$, and the bag proportion loss term in (8.5) is equivalent to that of (8.3).
- The formulation of (8.5) is more flexible than (8.3). The first two terms of the ∝SVM is exactly the conventional SVM. Therefore, ∝SVM can naturally incorporate any amount of supervised data without modification. The labels for such instances will be observed variables instead of being hidden. In the special case where no label proportions are provided, ∝SVM becomes large-margin clustering [221, 129], whose solution depends only on the data distribution. Compared with [179, 169], ∝SVM requires no restrictive assumptions on the data. ∝SVM can also be easily extended to the multi-class case, similar to [106].
- The strong connection of ∝SVM and conventional SVM enables efficient optimization algorithms, which will be presented in the following sections.

8.3.3 Connections to InvCal

As stated in Section 8.2, the Inverse Calibration method (InvCal) [179] treats the mean of each bag as a "super-instance", which is assumed to have a soft label corresponding to the label proportion. It is formulated as below.

$$\begin{aligned} \underset{\mathbf{w},b,\boldsymbol{\xi},\boldsymbol{\xi}^*}{\operatorname{argmin}} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{k=1}^{K}(\xi_k + \xi_k^*) && (8.6) \\ \text{s.t.} \quad & \xi_k \geq 0, \quad \xi_k^* \geq 0, \\ & \mathbf{w}^T\boldsymbol{m}_k + b \geq -\log(\frac{1}{\bar{y}(B)} - 1) - \epsilon_k - \xi_k, \\ & \mathbf{w}^T\boldsymbol{m}_k + b \leq -\log(\frac{1}{\bar{y}(B)} - 1) + \epsilon_k + \xi_k^*, \\ & k = 1, \cdots, M. \end{aligned}$$

(8.7)

in which the k-th bag mean is $\boldsymbol{m}_k = \frac{1}{|\mathcal{B}_k|}\sum_{i\in\mathcal{B}_k}\varphi(\mathbf{x}_i)$, $\forall_{k=1}^{K}$. Unlike ∝SVM, the proportion of the k-th bag is modeled on top of this "super-instance" $\boldsymbol{m}_k$ as:

$$q_k := \left(1+\exp\left(-\mathbf{w}^T\boldsymbol{m}_k + b\right)\right)^{-1}. \tag{8.8}$$

The second term of the objective function (8.6) tries to impose $q_k \approx \bar{y}(B_k)$, $\forall_{k=1}^{K}$, albeit in an inverse way. Though InvCal is shown to outperform other alternatives, including MeanMap [169] and several simple large-margin heuristics, it has a crucial limitation. Note that (8.8) is not a good way of measuring the proportion predicted by the model, especially when the data has high variance, or the data distribution is dependent on the bags. In our formulation (8.5), by explicitly modeling the unknown instance labels $\mathbf{y}$, the label proportion can be directly modeled as $p_k(\mathbf{y})$ given in (8.4). The advantage of our method is illustrated in a toy experiment shown in Figure 8.1 (for details see Section 8.6.1).

8.4 The alter-∝SVM Algorithm

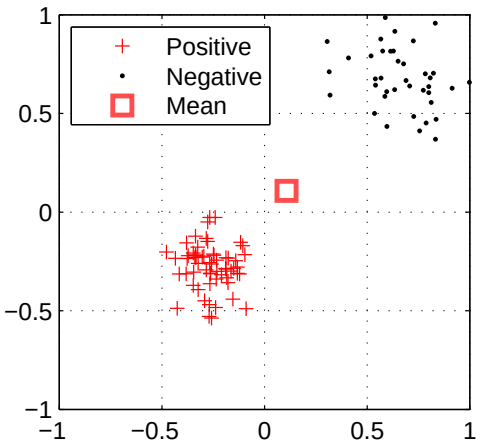
The ∝SVM formulation introduced in the previous section is fairly intuitive and straightforward. It is, however, a non-convex integer programming problem which is challenging to solve. In this chapter, we provide two solutions to efficiently find a local solution to it: a simple alternating optimization method (Section 8.4), and a convex relaxation method (Section 8.5).

In ∝SVM, the unknown instance labels $\mathbf{y}$ can be seen as a bridge between supervised learning loss and label proportion loss. Therefore, one natural way for solving (8.5) is via alternating optimization as,

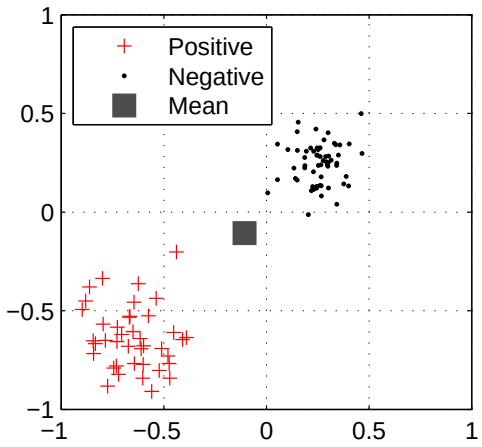
- For a fixed $\mathbf{y}$, the optimization of (8.5) $w.r.t$ $\mathbf{w}$ and b becomes a classic SVM problem.
- When $\mathbf{w}$ and b are fixed, the problem becomes:

$$\underset{\mathbf{y}}{\operatorname{argmin}} \quad \sum_{i=1}^{N} L'(y_i, \mathbf{w}^T\varphi(\mathbf{x}_i)+b) + \frac{C}{C'}\sum_{k=1}^{M} L\left(p_k(\mathbf{y}), \bar{y}(B_k)\right) \tag{8.9}$$

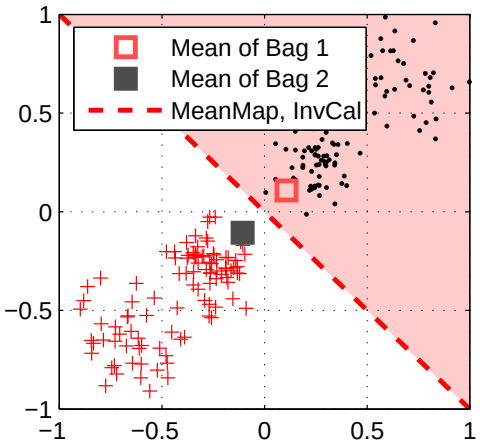
We show that the second step above can be solved efficiently. Because the influence of each bag $\{y_i | i \in \mathcal{B}_k\}$, $\forall_{k=1}^{M}$ on the objective is independent, we can optimize (8.9) on each



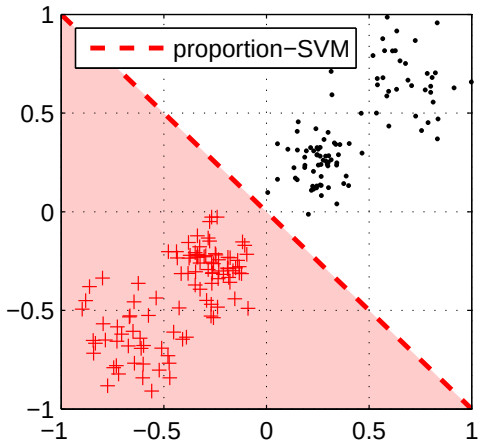
(a) Bag 1, with $p_1 = 0.6$



(b) Bag 2, with $p_2 = 0.4$



(c) MeanMap and InvCal with 0% accuracy.



(d) ∝SVM with 100% accuracy.

Figure 8.1: An example of learning with two bags to illustrate the drawbacks of the existing methods. (a) Data of bag 1. (b) Data of bag 2. (c) Learned separating hyperplanes of MeanMap and InvCal. (d) Learned separating hyperplane of ∝SVM (either alter-∝SVM or conv-∝SVM). More details are given in Section 8.6.1. Note that the algorithms do not have access to the individual instance labels.

bag separately. In particular, solving $\{y_i | i \in \mathcal{B}_k\}$ yields the following optimization problem:

$$\underset{\{y_i | i \in \mathcal{B}_k\}}{\operatorname{argmin}} \sum_{i \in \mathcal{B}_k} L'(y_i, \mathbf{w}^T \varphi(\mathbf{x}_i) + b) + \frac{C}{C'} L\left(p_k(\mathbf{y}), \bar{y}(B_k)\right) \tag{8.10}$$

Proposition 8.1. *For a fixed $p_k(\mathbf{y}) = \theta$, (8.10) can be optimally solved by the following steps.*

- Initialize $y_i = -1$, $i \in \mathcal{B}_k$. The optimal solution can be obtained by flipping the signs as below.
- When only flipping the sign of y_i, $i \in \mathcal{B}_k$, denote the reduction of the first term in (8.10) is δ_i. Sort δ_i, $i \in \mathcal{B}_k$.
- Flip the signs of the top-$(\theta|\mathcal{B}_k|)$ y_i's which have the highest reduction δ_i.

The proof of the above is shown in the Appendix.

For bag $\mathcal{B}_k$, we only need to sort the corresponding δ_i, $i \in \mathcal{B}_k$ once. Sorting takes $\mathcal{O}(|\mathcal{B}_k| \log(|\mathcal{B}_k|))$ time. After that, for each $\theta \in \{0, \frac{1}{|\mathcal{B}_k|}, \frac{2}{|\mathcal{B}_k|}, \cdots, 1\}$, the optimal solution can be computed incrementally, each taking $\mathcal{O}(1)$ time. We then pick the solution with the smallest objective value, yielding the optimal solution of (8.10).

Proposition 8.2. *Following the above steps, (8.9) can be solved in $\mathcal{O}(N \log(J))$ time, where $J = \max_{k=1 \cdots M} |\mathcal{B}_k|$, and N is the number of instances.*

As described in the paper, the influences of the bags in the objective function (6) are independent, and for the k-th bag, the algorithm takes $\mathcal{O}(|\mathcal{B}_k| \log(|\mathcal{B}_k|)$, $\forall k = 1, \cdots, M$. Overall, the complexity is $\mathcal{O}(\sum_{k=1}^{M} |\mathcal{B}_k| \log(|\mathcal{B}_k|))$. We know that $\sum_{k=1}^{M} |\mathcal{B}_k| = N$, $J = \max_{k=1,\cdots,M} |\mathcal{B}_k|$. Therefore we have:

$$\sum_{k=1}^{M} |\mathcal{B}_k| \log(|\mathcal{B}_k|) \leq \sum_{k=1}^{M} |\mathcal{B}_k| \log(J) = N \log(J).$$

By alternating between solving ($\mathbf{w}$, b) and $\mathbf{y}$, the objective is guaranteed to converge. This is due to the fact that the objective function is lower bounded and non-increasing. Furthermore, the objective converges in finite steps as there are finite ways of labeling the instances. In practice, we terminate the procedure when the objective no longer decreases (or if its decrease is smaller than a threshold). Empirically, the alternating optimization

Algorithm 4 alter-∝SVM (with annealing)

Initialize $C' = 10^{-5}C^*$.

Randomly initialize $y_i \in \{-1, 1\}$, $\forall_{i=1}^{N}$.

while $C' < C^*$ **do**

$C' = \min\{(1 + \Delta)C', C^*\}$

repeat

Fix $\mathbf{y}$ to solve $\mathbf{w}$ and b.

Fix $\mathbf{w}$ and b to solve $\mathbf{y}$ (Eq. (8.9) with C').

until The decrease of the objective is smaller than a threshold.

end while

typically terminates fast within tens of iterations, but one obvious problem is the possibility of local solutions.

To alleviate this problem, similar to T-SVM [96, 31], the proposed alter-∝SVM algorithm takes an additional annealing loop to gradually increase C'. The algorithm is shown in Algorithm 4. Because the nonconvexity of the objective function mainly comes from the second term of (8.5), the annealing can be seen as a "smoothing" step to alleviate the local minima problem. To justify the requirement of the annealing loop, we keep repeating the alter-∝SVM algorithm with/without the annealing loop, with different random initializations, on the same dataset. We record the smallest objective value found so far. As shown in Figure 8.2, alter-∝SVM without the annealing loop is much slower in finding a low objective value compared with alter-∝SVM with the annealing loop. Similar results can be found in other datasets, and other bag sizes. Following [31], we set $\Delta = 0.5$ in Algorithm 4. In addition, to further alleviate the local minima issues, in the experiment section we empirically choose to initialize alter-∝SVM 10 times, which gives us quite stable results.

Empirically, the inner loop of alter-∝SVM terminates (with a fixed C') within a few iterations. From Proposition 8.2, optimizing $\mathbf{y}$ has linear complexity in N (when J is small). Therefore, the overall complexity of the algorithm depends on the SVM solver. Specifically, when linear SVM is used [97], alter-∝SVM has linear complexity. In practice, to further alleviate the influence of the local solutions, similar to clustering, *e.g.*, kmeans, we repeat alter-∝SVM multiple times by randomly initializing $\mathbf{y}$, and then picking the solution

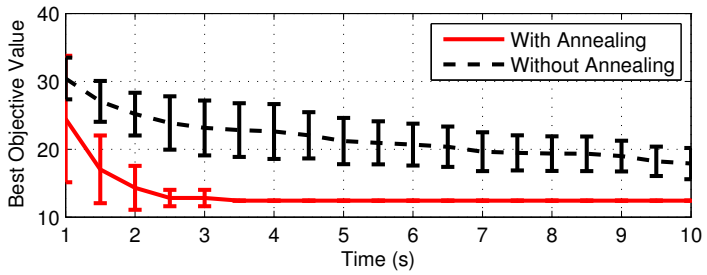


Figure 8.2: The smallest objective value with/without the annealing loop. The above results are based on experiments on the vote dataset with bag size of 32, linear kernel, $C^* = 1$, $C = 10$.

with the smallest objective value.

8.5 The conv-∝SVM Algorithm

In this section, we show that with proper relaxation of the ∝SVM formulation (8.5), the objective function can be transformed to a convex function of $\boldsymbol{M} := \mathbf{y}\mathbf{y}^T$. We then relax the solution space of $\boldsymbol{M}$ to its convex hull, leading to a convex optimization problem of $\boldsymbol{M}$. The conv-∝SVM algorithm is proposed to solve the relaxed problem. Unlike alter-∝SVM, conv-∝SVM does not require multiple initializations. This method is motivated by the techniques used in large-margin clustering [130, 221].

8.5.1 Convex Relaxation

We change the label proportion term in the objective function (8.5) as a constraint $\mathbf{y} \in \mathbb{Y}$, and we drop the bias term b[3]. Then at optimality the objective function can be written as:

[3]If the bias term is not dropped, there will be constraint $\boldsymbol{\alpha}^T\mathbf{y} = 0$ in the dual, leading to non-convexity. Such a difficulty has also been discussed in [221]. Fortunately, the effect of removing the bias term can be alleviated by zero-centering the data or augmenting the feature vector with an additional dimension with value 1.

$$\min_{\mathbf{y}\in\mathbb{Y}} \min_{\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C' \sum_{i=1}^{N} L'(y_i, \mathbf{w}^T\varphi(\mathbf{x}_i)) \tag{8.11}$$
$$\mathbb{Y} = \Big\{\mathbf{y}\Big||p_k(\mathbf{y}) - \bar{y}(B_k)| \leq \epsilon, y_i \in \{-1, 1\}, \forall_{k=1}^{M}\Big\},$$

in which ϵ controls the compatibility of the label proportions. The constraint $\mathbf{y} \in \mathbb{Y}$ can be seen as a special loss function:

$$L(p_k(\mathbf{y}), \bar{y}(B_k)) = \begin{cases} 0, & \text{if } |p_k(\mathbf{y}) - \bar{y}(B_k)| < \epsilon, \\ \infty, & \text{otherwise.} \end{cases} \tag{8.12}$$

We then write the inner problem of (8.11) as its dual:

$$\min_{\mathbf{y}\in\mathbb{Y}} \max_{\boldsymbol{\alpha}\in\mathcal{A}} -\frac{1}{2}\boldsymbol{\alpha}^T \left(\boldsymbol{\mathcal{K}} \odot \mathbf{y}\mathbf{y}^T\right)\boldsymbol{\alpha} + \boldsymbol{\alpha}^T\mathbf{1}, \tag{8.13}$$

in which $\boldsymbol{\alpha} \in \mathbb{R}^N$, $\odot$ denotes pointwise-multiplication, $\boldsymbol{\mathcal{K}}$ is the kernel matrix with $\mathcal{K}_{ij} = \varphi(\mathbf{x}_i)^T\varphi(\mathbf{x}_j)$, $\forall_{i,j=1}^{N}$, and $\mathcal{A} = \{\boldsymbol{\alpha}|0 \leq \boldsymbol{\alpha} \leq C'\}$.

The objective in (8.13) is non-convex in $\mathbf{y}$, but convex in $\boldsymbol{M} := \mathbf{y}\mathbf{y}^T$. Following [130, 221], we instead solve the optimal $\boldsymbol{M}$. However, the feasible space of $\boldsymbol{M}$ is

$$\mathcal{M}_0 = \{\mathbf{y}\mathbf{y}^T|\mathbf{y} \in \mathbb{Y}\}, \tag{8.14}$$

which is a non-convex set. In order to get a convex optimization problem, we relax $\mathcal{M}_0$ to its convex hull, the tightest convex relaxation of $\mathcal{M}_0$:

$$\mathcal{M} = \Big\{\sum_{\mathbf{y}\in\mathbb{Y}} \mu_{(\mathbf{y})}\mathbf{y}\mathbf{y}^T\Big|\boldsymbol{\mu} \in \mathcal{U}\Big\}, \tag{8.15}$$

in which $\mathcal{U} = \{\boldsymbol{\mu}|\sum_{\mathbf{y}\in\mathbb{Y}} \mu_{(\mathbf{y})} = 1, \mu_{(\mathbf{y})} \geq 0\}$. Thus solving the relaxed $\boldsymbol{M}$ is identical to finding $\boldsymbol{\mu}$:

$$\min_{\boldsymbol{\mu}\in\mathcal{U}} \max_{\boldsymbol{\alpha}\in\mathcal{A}} -\frac{1}{2}\boldsymbol{\alpha}^T \left(\sum_{\mathbf{y}\in\mathbb{Y}} \mu_{(\mathbf{y})}\boldsymbol{\mathcal{K}} \odot \mathbf{y}\mathbf{y}^T\right)\boldsymbol{\alpha} + \boldsymbol{\alpha}^T\mathbf{1}. \tag{8.16}$$

(8.16) can be seen as Multiple Kernel Learning (MKL) [10], which is a widely studied problem. However, because $|\mathbb{Y}|$ is very large, it is not tractable to solve (8.16) directly.

8.5.2 Cutting Plane Training

Fortunately, we can assume that at optimality only a small number of $\mathbf{y}$'s are active in (8.16). Define $\mathbb{Y}_{active} \subset \mathbb{Y}$ as the set containing all the active $\mathbf{y}$'s. We show that $\mathbf{y} \in \mathbb{Y}_{active}$ can be incrementally found by the cutting plane method.

Because the objective function of (8.16) is convex in $\boldsymbol{\mu}$, and concave in $\boldsymbol{\alpha}$, it is equivalent to solve the following problem [57],

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}} \min_{\boldsymbol{\mu}\in\mathcal{U}} -\frac{1}{2}\boldsymbol{\alpha}^T \left(\sum_{\mathbf{y}\in\mathbb{Y}} \mu_{(\mathbf{y})} \boldsymbol{\mathcal{K}} \odot \mathbf{y}\mathbf{y}^T \right) \boldsymbol{\alpha} + \boldsymbol{\alpha}^T \mathbf{1}. \tag{8.17}$$

It is easy to verify that the above is equivalent to:

$$\begin{aligned} \max_{\boldsymbol{\alpha}\in\mathcal{A},\beta} \quad & -\beta \\ \text{s.t.} \quad & \beta \geq \frac{1}{2}\boldsymbol{\alpha}^T \left(\boldsymbol{\mathcal{K}} \odot \mathbf{y}\mathbf{y}^T \right) \boldsymbol{\alpha} + \boldsymbol{\alpha}^T \mathbf{1}, \forall \mathbf{y} \in \mathbb{Y}. \end{aligned} \tag{8.18}$$

This form enables us to apply the cutting plane method [107] to incrementally include the most violated $\mathbf{y}$ into $\mathbb{Y}_{active}$, and then solve the MKL problem, (8.16) with $\mathbb{Y}$ replaced as $\mathbb{Y}_{active}$. The above can be repeated until no violated $\mathbf{y}$ exists.

In the cutting plane training, the critical step is to obtain the most violated $\mathbf{y} \in \mathbb{Y}$:

$$\arg\max_{\mathbf{y}\in\mathbb{Y}} \frac{1}{2}\boldsymbol{\alpha}^T \left(\boldsymbol{\mathcal{K}} \odot \mathbf{y}\mathbf{y}^T \right) \boldsymbol{\alpha} + \boldsymbol{\alpha}^T \mathbf{1}, \tag{8.19}$$

which is equivalent to

$$\arg\max_{\mathbf{y}\in\mathbb{Y}} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j \varphi(\mathbf{x}_i)^T \varphi(\mathbf{x}_j). \tag{8.20}$$

This is a 0/1 concave QP, for which there exists no efficient solution. However, instead of finding the most violated constraint, if we find any violated constraint $\mathbf{y}$, the objective function still decreases. We therefore relax the objective in (8.20), which can be solved efficiently. Note that the objective of (8.20) is equivalent to a ℓ_2 norm $\sum_{i=1}^{N} \parallel \alpha_i y_i \varphi(\mathbf{x}_i) \parallel_2$. Following [130], we approximate it as the ℓ_∞ norm:

$$\sum_{i=1}^{N} \parallel \alpha_i y_i \varphi(\mathbf{x}_i) \parallel_\infty \equiv \max_{j=1\cdots d} \left| \sum_{i=1}^{N} \alpha_i y_i x_i^{(j)} \right|, \tag{8.21}$$

Algorithm 5 conv-∝SVM

Initialize $\alpha_i = 1/N$, $\forall_{i=1}^{N}$. $\mathbb{Y}_{active} = \varnothing$. Output: $\boldsymbol{M} \in \mathcal{M}$

repeat

Compute $\mathbf{y} \in \mathbb{Y}$ based on (8.20) − (8.23).

$\mathbb{Y}_{active} \leftarrow \mathbb{Y}_{active} \cup \{\mathbf{y}\}$.

Solve the MKL problem in (8.16) with $\mathbb{Y}_{active}$ to get $\mu_{(\mathbf{y})}$, $\mathbf{y} \in \mathbb{Y}_{active}$.

until The decrease of the objective is smaller than a threshold.

in which $x_i^{(j)}$ is the j-th dimension of the i-th feature vector. These can be obtained by eigendecomposition of the kernel matrix $\boldsymbol{\mathcal{K}}$, when a nonlinear kernel is used. The computational complexity is $\mathcal{O}(dN^2)$. In practice, we choose d such that 90% of the variance is preserved. We further rewrite (8.21) as:

$$\begin{aligned} &\max_{j=1\cdots d} \max \left(\sum_{i=1}^{N} \alpha_i y_i x_i^{(j)}, -\sum_{i=1}^{N} \alpha_i y_i x_i^{(j)} \right) \\ = &\max_{j=1\cdots d} \max \left(\sum_{k=1}^{M} \sum_{i \in \mathcal{B}_k} \alpha_i y_i x_i^{(j)}, \sum_{k=1}^{M} \sum_{i \in \mathcal{B}_k} -\alpha_i y_i x_i^{(j)} \right). \end{aligned} \tag{8.22}$$

Therefore the approximation from (8.20) to (8.21) enables us to consider each dimension and each bag separately. For the j-th dimension, and the k-th bag, we only need to solve two sub-problems $\max_{\mathbf{y} \in \mathbb{Y}} \sum_{i \in \mathcal{B}_k} \alpha_i y_i x_i^{(j)}$, and $\max_{\mathbf{y} \in \mathbb{Y}} -\sum_{i \in \mathcal{B}_k} \alpha_i y_i x_i^{(j)}$. The former, as an example, can be written as

$$\min_{\{y_i | i \in \mathcal{B}_k\}} \sum_{i \in \mathcal{B}_k} \left(-\alpha_i x_i^{(j)} \right) y_i, \quad |p_k(\mathbf{y}) - \bar{y}(B_k)| \leq \epsilon. \tag{8.23}$$

This can be solved in the same way as (8.10), which takes $\mathcal{O}(|\mathcal{B}_k| \log |\mathcal{B}_k|)$ time. Because we have d dimensions, similar to Proposition 8.2, one can show that:

Proposition 8.3. *(8.20) with the ℓ_2 norm approximated as the ℓ_∞ norm can be solved in $\mathcal{O}(dN \log(J))$ time, where $J = \max_{k=1\cdots K} |\mathcal{B}_k|$, d is the feature dimension, and N is the number of instances.*

8.5.3 The Algorithm

The overall algorithm, called conv-∝SVM, is shown in Algorithm 5. Following [130], we use an adapted SimpleMKL algorithm [172] to solve the MKL problem. As an additional step, we need to recover $\mathbf{y}$ from $\boldsymbol{M}$. This is achieved by rank-1 approximation of $\boldsymbol{M}$ (as $\mathbf{y}\mathbf{y}^T$)[4]. Because of the convex relaxation, the computed $\mathbf{y}$ is not binary. However, we can use the real-valued $\mathbf{y}$ directly in our prediction model (with dual):

$$h(\mathbf{x}) = \text{sign}\left(\sum_{i=1}^{N} \alpha_i y_i \varphi(\mathbf{x}_i)^T \varphi(\mathbf{x})\right). \tag{8.24}$$

Similar to alter-∝SVM, the objective of conv-∝SVM is guaranteed to converge. In practice, we terminate the algorithm when the decrease of the objective is smaller than a threshold. Typically the SimpleMKL converges in less than 5 iterations, and conv-∝SVM terminates in less than 10 iterations. The SimpleMKL takes $\mathcal{O}(N^2)$ (computing the gradient) time, or the complexity of SVM, whichever is higher. Recovering $\mathbf{y}$ takes $\mathcal{O}(N^2)$ time and computing eigendecomposition with the first d singular values takes $\mathcal{O}(dN^2)$ time.

8.6 Experiments

MeanMap [169] was shown to outperform alternatives including kernel density estimation, discriminative sorting and MCMC [111]. InvCal [179] was shown to outperform MeanMap and several large-margin alternatives. Therefore, in the experiments, we only compare our approach with MeanMap and InvCal.

8.6.1 A Toy Experiment

To visually demonstrate the advantage of our approach, we first show an experiment on a toy dataset with two bags. Figure 8.1 (a) and (b) show the data of the two bags, and Figure 8.1 (c) and (d) show the learned separating hyperplanes from different methods. Linear kernel is used in this experiment. For this specific dataset, the restrictive data assumptions of MeanMap and InvCal do not hold: the mean of the first bag (60% positive) is on the "negative side", whereas, the mean of the second bag (40% positive) is on the "positive side".

[4]Note that $\mathbf{y}\mathbf{y}^T = (-\mathbf{y})(-\mathbf{y})^T$. This ambiguity can be resolved by validation on the training bags.

Dataset	Size	# Attributes	# Classes
HEART	270	13	2
COLIC	366	22	2
VOTE	435	16	2
AUSTRALIAN	690	14	2
BREAST-W	699	9	2
DNA	2,000	180	3
SATIMAGE	4,435	36	6

Table 8.2: Datasets used in experiments.

Consequently, both MeanMap and InvCal completely fail, with the classification accuracy of 0%. On the other hand, our method, which does not make strong data assumptions, achieves the perfect performance with 100% accuracy.

8.6.2 UCI/LibSVM Datasets

Datasets. We compare the performance of different techniques on various datasets from the UCI repository[5] and the LibSVM collection[6]. The details of the datasets are listed in Table 8.2. In this chapter, we focus on the binary classification settings. For the datasets with multiple classes (DNA and SATIMAGE), we test the one-vs-rest binary classification performance, by treating data from one class as positive, and randomly selecting same amount of data from the remaining classes as negative. For each dataset, the attributes are scaled to $[-1, 1]$.

Experimental Setup. Following [179], we first randomly split the data into bags of a fixed size. Bag sizes of 2, 4, 8, 16, 32, 64 are tested. We then conduct experiments with 5-fold cross validation. The performance is evaluated based on the average classification accuracy on the individual test instances. We repeat the above processes 5 times (randomly selecting negative examples for the multi-class datasets, and randomly splitting the data

[5] http://archive.ics.uci.edu/ml/

[6] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/

into bags), and report the mean accuracies with standard deviations.

The parameters are tuned by an inner cross-validation loop on the training subset of each partition of the 5-fold validation. Because no instance-level labels are available during training, we use the bag-level error on the validation bags to tune the parameters:

$$Err = \sum_{k=1}^{T} |p_k - \bar{y}(B_k)|\,, \tag{8.25}$$

in which p_k and $\bar{y}(B_k)$ denote the predicted and the ground-truth proportions for the k-th validation bag.

For MeanMap, the parameter is tuned from $\lambda \in \{0.1, 1, 10\}$. For InvCal, the parameters are tuned from $C' \in \{0.1, 1, 10\}$, and $\epsilon \in \{0, 0.01, 0.1\}$. For alter-∝SVM, the parameters are tuned from $C^* \in \{0.1, 1, 10\}$, and $C \in \{1, 10, 100\}$. For conv-∝SVM, the parameters are tuned from $C' \in \{0.1, 1, 10\}$, and $\epsilon \in \{0, 0.01, 0.1\}$. Two kinds of kernels are considered: linear and RBF. The parameter of the RBF kernel is tuned from $\gamma = \{0.01, 0.1, 1\}$.

We randomly initialize alter-∝SVM 10 times and pick the result with the smallest objective value. Empirically, the influence of random initialization to other algorithms is minimal.

Results. Table 8.3 and Table 8.4 show the results with linear kernel, and RBF kernel, respectively. Our methods consistently outperform MeanMap and InvCal, with p-value < 0.05 for most of the comparisons (more than 70%). For larger bag sizes, the problem of learning from label proportions becomes more challenging due to the limited amount of supervision. For these harder cases, the gains from ∝SVM are typically even more significant. For instance, on the `DNA-2` dataset, with RBF kernel and bag size 64, alter-∝SVM outperforms the former works by 19.82% and 12.69%, respectively (Table 8.4).

A Large-Scale Experiment. We also conduct a large-scale experiment on the cod-rna.t dataset containing about 271K points. The performance of InvCal and alter-∝SVM with linear kernel are compared. The experimental setting is the same as for the other datasets. The results in Table 8.5 show that alter-∝SVM consistently outperforms InvCal. For smaller bag sizes also, alter-∝SVM outperforms InvCal, though the improvement margin reduces due to sufficient amount of supervision.

Dataset	**Method**	2	4	8	16	32	64
HEART	MeanMap	81.85±1.28	80.39±0.47	79.63±0.83	79.46±1.46	79.00±1.42	76.06±1.25
	InvCal	81.78±0.55	80.98±1.35	79.45±3.07	76.94±3.26	73.76±2.69	73.04±6.46
	alter-∝SVM	**83.41±0.71**	**81.80±1.25**	79.91±2.11	79.69±0.64	77.80±2.52	76.58±2.00
	conv-∝SVM	83.33±0.59	80.61±2.48	**81.00±0.75**	**80.72±0.82**	**79.32±1.14**	**79.40±0.72**
COLIC	MeanMap	80.00±0.80	76.14±1.69	75.52±0.72	74.17±1.61	76.10±1.92	76.74±6.10
	InvCal	81.25±0.24	78.82±3.24	77.34±1.62	74.84±4.14	69.63±4.12	69.47±6.06
	alter-∝SVM	**81.42±0.02**	**80.79±1.48**	**79.59±1.38**	**79.40±1.06**	**78.59±3.32**	**78.49±2.93**
	conv-∝SVM	81.42±0.02	80.63±0.77	78.84±1.32	77.98±1.14	77.49±0.66	76.94±1.07
VOTE	MeanMap	87.76±0.20	91.90±1.89	90.84±2.33	88.72±1.45	87.63±0.26	88.42±0.80
	InvCal	95.57±0.11	95.57±0.42	94.43±0.24	94.00±0.61	91.47±2.57	91.13±1.07
	alter-∝SVM	**95.62±0.33**	**96.09±0.41**	**95.56±0.47**	**94.23±1.35**	**91.97±1.56**	**92.12±1.20**
	conv-∝SVM	91.66±0.19	90.80±0.34	89.55±0.25	88.87±0.37	88.95±0.39	89.07±0.24
AUSTRALIAN	MeanMap	**86.03±0.39**	85.62±0.17	84.08±1.36	83.70±1.45	83.96±1.96	82.90±1.96
	InvCal	85.42±0.28	**85.80±0.37**	84.99±0.68	83.14±2.54	80.28±4.29	80.53±6.18
	alter-∝SVM	85.42±0.30	85.60±0.39	85.49±0.78	84.96±0.96	85.29±0.92	84.47±2.01
	conv-∝SVM	85.51±0.00	85.54±0.08	**85.90±0.54**	**85.67±0.24**	**85.67±0.81**	**85.47±0.89**
BREAST-W	MeanMap	96.11±0.06	95.97±0.25	96.13±0.16	96.26±0.32	95.96±0.42	95.80±0.92
	InvCal	95.88±0.33	95.65±0.36	95.53±0.24	95.39±0.57	95.23±0.52	94.31±0.77
	alter-∝SVM	**96.71±0.29**	**96.77±0.13**	**96.59±0.24**	**96.41±0.50**	**96.41±0.21**	**96.25±0.49**
	conv-∝SVM	92.27±0.27	92.25±0.16	92.32±0.13	94.03±0.18	94.60±0.10	94.57±0.21
DNA-1	MeanMap	86.38±1.33	82.71±1.26	79.89±1.55	78.46±0.53	80.20±1.44	78.83±1.73
	InvCal	93.05±1.45	90.81±0.87	86.27±2.43	81.58±3.09	78.31±3.28	72.98±2.33
	alter-∝SVM	**94.93±1.05**	**94.31±0.62**	**92.86±0.78**	**90.72±1.35**	**90.84±0.52**	**89.41±0.97**
	conv-∝SVM	92.78±0.66	90.08±1.18	85.38±2.05	84.91±2.43	82.77±3.30	85.66±0.20
DNA-2	MeanMap	88.45±0.68	83.06±1.68	78.69±2.11	79.94±5.68	79.72±3.73	74.73±4.26
	InvCal	93.30±0.88	90.32±1.89	87.30±1.80	83.17±2.18	79.47±2.55	76.85±3.42
	alter-∝SVM	**94.74±0.56**	**94.49±0.46**	**93.06±0.85**	**91.82±1.59**	**90.81±1.55**	**90.08±1.45**
	conv-∝SVM	94.35±1.01	92.08±1.48	89.72±1.26	88.27±1.87	87.58±1.54	86.55±1.18
DNA-3	MeanMap	87.57±0.74	83.95±1.34	80.22±0.65	79.14±2.39	75.21±0.89	74.99±1.53
	InvCal	91.77±0.42	89.38±0.41	87.98±0.83	84.28±1.63	79.65±3.55	75.22±5.64
	alter-∝SVM	**93.21±0.33**	**92.83±0.40**	**91.80±0.52**	**88.77±1.10**	**86.94±0.41**	**86.39±1.70**
	conv-∝SVM	91.72±0.26	87.93±1.32	80.13±2.39	73.93±0.46	73.38±0.56	72.87±0.79
SATIMAGE-2	MeanMap	97.21±0.38	96.27±0.77	95.85±1.12	94.65±0.31	94.49±0.37	94.52±0.28
	InvCal	88.41±3.14	94.65±0.56	94.70±0.20	94.49±0.31	92.90±1.05	93.82±0.60
	alter-∝SVM	**97.83±0.51**	**97.75±0.43**	**97.52±0.48**	**97.52±0.51**	**97.51±0.20**	**97.11±0.26**
	conv-∝SVM	96.87±0.23	96.63±0.09	96.40±0.22	96.87±0.38	96.29±0.40	96.50±0.38

Table 8.3: Accuracy with linear kernel, with bag size 2, 4, 8, 16, 32, 64.

Dataset	**Method**	2	4	8	16	32	64
HEART	MeanMap	82.69±0.71	80.80±0.97	79.65±0.82	79.44±1.21	80.03±2.05	77.26±0.85
	InvCal	**83.15±0.56**	81.06±0.70	80.26±1.32	79.61±3.84	76.36±3.72	73.90±3.00
	alter-∝SVM	83.15±0.85	**82.89±1.30**	**81.51±0.54**	80.07±1.21	79.10±0.96	78.63±1.85
	conv-∝SVM	82.96±0.26	82.20±0.52	81.38±0.53	**81.17±0.55**	**80.94±0.86**	**78.87±1.37**
COLIC	MeanMap	82.45±0.88	81.38±1.26	81.71±1.16	79.94±1.33	76.36±2.43	77.84±1.69
	InvCal	82.20±0.61	81.20±0.87	81.17±1.74	78.59±2.19	74.09±5.26	72.81±4.80
	alter-∝SVM	**83.28±0.50**	**82.97±0.39**	**82.03±0.44**	**81.62±0.46**	**81.53±0.21**	**81.39±0.34**
	conv-∝SVM	82.74±1.15	81.83±0.46	79.58±0.57	79.77±0.84	78.22±1.19	77.31±1.76
VOTE	MeanMap	91.15±0.33	90.52±0.62	91.54±0.20	90.28±1.63	89.58±1.09	89.38±1.33
	InvCal	95.68±0.19	94.77±0.44	93.95±0.43	**93.03±0.37**	87.79±1.64	86.63±4.74
	alter-∝SVM	**95.80±0.20**	**95.54±0.25**	**94.88±0.94**	92.44±0.60	**90.72±1.11**	**90.93±1.30**
	conv-∝SVM	92.99±0.20	92.01±0.69	90.57±0.68	88.98±0.35	88.74±0.43	88.62±0.60
AUSTRALIAN	MeanMap	85.97±0.72	85.88±0.34	85.34±1.01	83.36±2.04	83.12±1.52	80.58±5.41
	InvCal	**86.06±0.30**	86.11±0.26	**86.32±0.45**	84.13±1.62	82.73±1.70	81.87±3.29
	alter-∝SVM	85.74±0.22	85.71±0.21	86.26±0.61	**85.65±0.43**	83.63±1.83	**83.62±2.21**
	conv-∝SVM	85.97±0.53	**86.46±0.23**	85.30±0.70	84.18±0.53	**83.69±0.78**	82.98±1.32
BREAST-W	MeanMap	96.42±0.18	96.45±0.27	96.20±0.27	96.14±0.46	94.91±1.02	94.53±1.24
	InvCal	96.85±0.23	96.91±0.13	96.77±0.22	96.75±0.22	96.65±0.29	94.58±1.76
	alter-∝SVM	**96.97±0.07**	**97.00±0.18**	**96.94±0.07**	**96.87±0.15**	**96.88±0.25**	**96.70±0.14**
	conv-∝SVM	96.71±0.10	96.60±0.06	96.57±0.08	96.54±0.19	96.77±0.17	96.66±0.14
DNA-1	MeanMap	91.53±0.25	90.58±0.34	86.00±1.04	80.77±3.69	77.35±3.59	68.47±4.30
	InvCal	89.32±3.39	92.73±0.53	87.99±1.65	81.05±3.14	74.77±2.95	67.75±3.86
	alter-∝SVM	**95.67±0.40**	**94.65±0.52**	**93.71±0.47**	**92.52±0.63**	**91.85±1.42**	**90.64±1.32**
	conv-∝SVM	93.36±0.53	86.75±2.56	81.03±3.58	75.90±4.56	76.92±5.91	77.94±2.48
DNA-2	MeanMap	92.08±1.54	91.03±0.69	87.50±1.58	82.21±3.08	76.77±4.33	72.56±5.32
	InvCal	89.65±4.05	93.12±1.37	89.19±1.17	83.52±2.57	77.94±2.82	72.64±3.89
	alter-∝SVM	**95.63±0.45**	**95.05±0.75**	**94.25±0.50**	**93.95±0.93**	**92.74±0.93**	**92.46±0.90**
	conv-∝SVM	94.06±0.86	90.68±1.18	87.64±0.76	87.32±1.55	85.74±1.03	85.33±0.79
DNA-3	MeanMap	90.99±0.65	89.45±1.12	88.01±0.65	84.30±1.36	79.59±2.49	73.88±4.89
	InvCal	93.23±0.44	91.83±0.63	89.49±0.52	85.47±1.33	78.26±3.57	70.91±3.00
	alter-∝SVM	**94.36±0.31**	**93.28±0.25**	**92.40±0.35**	**90.04±0.65**	**87.89±1.10**	**86.40±1.26**
	conv-∝SVM	91.75±0.45	87.48±2.02	80.41±0.70	75.91±0.29	75.37±1.66	74.63±0.21
SATIMAGE-2	MeanMap	97.08±0.48	96.82±0.38	96.50±0.43	96.45±1.16	95.51±0.73	94.26±0.22
	InvCal	97.53±1.33	98.33±0.13	98.38±0.23	97.99±0.54	96.27±1.15	94.47±0.27
	alter-∝SVM	**98.83±0.36**	**98.69±0.37**	**98.62±0.27**	**98.72±0.37**	**98.51±0.22**	**98.25±0.41**
	conv-∝SVM	96.55±0.13	96.45±0.19	96.45±0.39	96.14±0.49	96.16±0.35	95.93±0.45

Table 8.4: Accuracy with RBF kernel, with bag size 2, 4, 8, 16, 32, 64.

Method	2^{11}	2^{12}	2^{13}
InvCal	88.79±0.21	88.20±0.62	87.89±0.79
alter-∝SVM	**90.32±1.22**	**90.28±0.94**	**90.21±1.53**

Table 8.5: Accuracy on cod-rna.t, with linear kernel, with bag size 2^{11}, 2^{12}, 2^{13}.

8.7 Discussions

Robustness to the Given Proportions $\{\bar{y}(B_k)\}_{k=1}^M$. In Section 8.6.2, because the bags were randomly generated, distribution of $\{\bar{y}(B_k)\}_{k=1}^M$ is approximately Gaussian for moderate to large K. It is intuitive that the performance will depend on the distribution of proportions $\{\bar{y}(B_k)\}_{k=1}^M$. If $\bar{y}(B_k)$ is either 0 or 1, the bags are most informative, because this leads to the standard supervised learning setting. On the other hand, if $\bar{y}(B_k)$'s are close to each other, the bags will be least informative. In fact, both MeanMap and InvCal cannot reach a numerically stable solution in such case. For MeanMap, the linear equations for solving class means will be ill-posed. For InvCal, because all the "super-instances" are assumed to have the same regression value, the result is similar to random guess. ∝SVM, on the other hand, can achieve good performance even in this challenging situation. For example, when using the vote dataset, with bag sizes 8 and 32, $\bar{y}(B_k) = 38.6\%$, $\forall_{k=1}^M$(same as prior), with linear kernel, alter-∝SVM has accuracies(%) 94.23 ± 1.02 and 86.71 ± 1.30, and conv-∝SVM has accuracies(%) 89.60 ± 0.59 and 87.69 ± 0.51, respectively. These results are close to those obtained for randomly generated bags in Table 8.3. This indicates that our method is less sensitive to the distribution of $\{\bar{y}(B_k)\}_{k=1}^M$.

Choice of Algorithms. Empirically, when nonlinear kernel is used, the run time of alter-∝SVM is longer than that of conv-∝SVM, because we are repeating alter-∝SVM multiple times to pick the solution with the smallest objective value. For instance, on a machine with 4-core 2.5GHz CPU, on the vote dataset with RBF kernel and 5-fold cross validation, the alter-∝SVM algorithm (repeating 10 times with the annealing loop, and one set of parameters) takes 15.0 seconds on average while the conv-∝SVM algorithm takes only 4.3 seconds. But as shown in the experimental results, for many datasets, the performance of conv-∝SVM is marginally worse than that of alter-∝SVM. This can be explained by the multiple relaxations used in conv-∝SVM, and also the 10 time initializations of alter-

∝SVM. As a heuristic solution for speeding up the computation, one can use conv-∝SVM (or InvCal) to initialize alter-∝SVM. For large-scale problems, in which linear SVM is used, alter-∝SVM is preferred, because its computational complexity is $\mathcal{O}(N)$.

8.8 Conclusion and Future Works

We proposed the ∝SVM framework for learning with label proportions, and introduced algorithms to efficiently solve the optimization problem. Experiments on several standard and one large-scale dataset showed the advantage of the proposed approach over the existing methods. The simple, yet flexible form of ∝SVM framework naturally spans supervised, unsupervised and semi-supervised learning. Due to the usage of latent labels, ∝SVM can also be potentially used in learning with label errors.

For the future directions, we are working on improving both the efficiency and the stability of ∝SVM. Due to the usefulness of annealing for ∝SVM, deterministic annealing [191] can be explored to further improve the algorithm. The speed of both alter-∝SVM and conv-∝SVM can be improved further by solving the SVM in their inner loops incrementally. For example, one can use warm start and partial active-set methods [189]. In addition, one can linearize kernels using explicit feature maps (Chapter 5), so that alter-∝SVM has linear complexity even for certain nonlinear kernels.

Chapter 9

Applications: Video Event Recognition by Discovering Discriminative Visual Segments

9.1 Introduction

Video event detection is useful in many applications such as video search, consumer video analysis, personalized advertising, and video surveillance, to name a few [156]. The most commonly used approach for video event detection is to represent a video as a global Bag-of-Word (BoW) vector [192]. Representing a video as a single vector is simple and efficient. Unfortunately, much information may be lost in this paradigm. In fact, a video is comprised of multiple "instances", such as frames and shots. Some instances contain key evidence of the event being considered. For example, event like "birthday party" may be well detected by frames containing cakes, and candles, and "parkour" may be well detected by shots of person jumping up and down on the street [17]. Intuitively, by considering the instances of the videos, more distinctive event patterns can be learned, and better event recognition can be achieved.

In this chapter, we study instance-based video classification. Each video contains multiple "instances", defined as video segments of different temporal lengths. Our goal is to

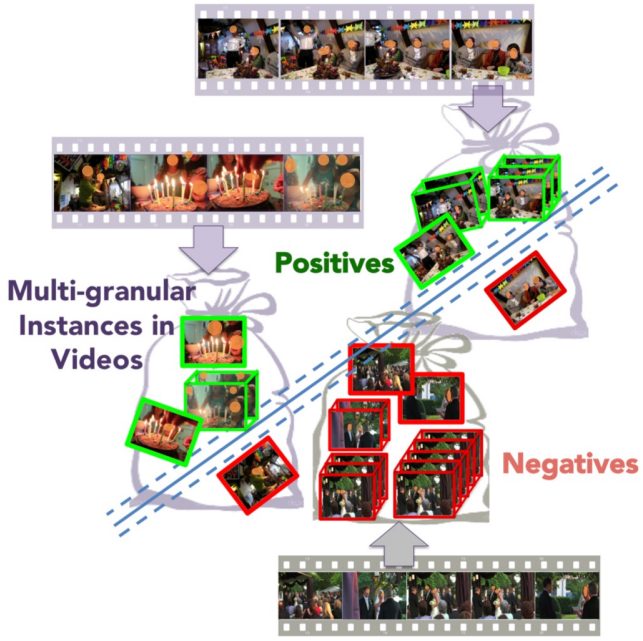


Figure 9.1: Illustration of the proposed framework. The event "birthday party" can be recognized by instances containing "birthday cake" and "blowing candles". Our method simultaneously infers hidden instance labels and instance-level classification model (the separating hyperplane) based on only video labels.

learn an instance-level event detection model, while assuming only a video-level event label (whether the event happens in the video) is available. Instance labels, on the other hand, are not available due to the prohibitive cost in the annotation. We model the problem as a modified LLP setting. We treat one video as a bag. Compared with the conventional LLP setting, we only know a bag-level label, instead of the exact proportion. Our key assumption is that, for an event, the positive videos usually have a large proportion of positive instances which are discriminative for the event, while the negative videos have few positive instances. The idea is illustrated in Figure 9.1. The proposed method not only leads to more accurate event detection results but also learns the instance-level detector, explaining when and why a certain event happens in the video.

9.2 Related Works

Video Event Detection. Video event detection is a widely studied topic in computer vision. A good survey of state-of-the-arts was made in [95]. Generally speaking, a video event detection system can be divided into three stages: feature extraction, feature quantization and pooling, training/recognition.

One focus of previous research is on designing new features, including low-level features of visual features [43, 135], action features [214], audio features [146], and mid-level representation including concept feature, attributes [203] *etc.* There are also significant efforts on improving the event recognition modeling, such as max-margin based methods, graphical models, and some knowledge based techniques, as reviewed in [95]. However, most former approaches rely on a global vector to represent one video. The global approach neglects important local information of the events. Recently some researchers attempted to address this problem and proposed several new algorithms. Tang et al. [201] treated video segments as latent variables and adopted variable-duration hidden Markov model to represent events. Cao et al. [27] proposed scene aligned pooling, which divides videos into shots with different scenes, and pooling local features under each scene. Li et al. [128] proposed dynamic pooling, which employs various strategies to split videos into segments based on temporal structures. Different from their methods, which focus on exploiting temporal structures for pooling, our framework focuses on learning "instance" labels. The proposed approach can also be seen as complementary to the above pooling strategies, for which the video instances can be formed by dynamic pooling or scene aligned pooling.

Recently, it has also been shown that visual event recognition can be achieved efficiently by considering the "key" instances which may sometimes occupy a small portion of the whole video [17, 117, 197]. It will be worthwhile to compare the discriminative instances discovered by this work, and those by the works above.

Multiple-Instance Learning. In order to use local patterns in a video, one readily available learning method is Multiple-Instance Learning (MIL) [50]. In MIL, the training data is provided in "bags". And the labels are only provided on the bag-level. A bag is labeled as positive *iff* one or more instances inside the bag are positive. In computer vision, MIL has been applied in scene classification [139], image retrieval [240], and image classifi-

cation [35]. The two most popular algorithms for MIL are mi-SVM and MI-SVM [5]. The first algorithm emphasizes searching max-margin hyperplanes to separate positive and negative instances, while the second algorithm selects the most representative positive instance for each positive bag during optimization iterations, and concentrates on bag classification. Several methods have been proposed to address the limitations of MIL. Chen et al. [36] proposed to embed bags into instance space via instance similarity measure. Zhang et al. [238] proposed a new method which considers both local (instance) and global (bag) feature vectors.

In event detection, a video can be seen as a bag containing multiple instances. The labels are only provided on video-level. Therefore, algorithms of MIL can be directly applied. However, existing algorithms of MIL are not suitable for video event classification. One restriction is that MIL relies on a single instance (often computed based on the max function) in prediction, making the method very sensitive to false alarm outliers; another drawback is that it assumes that negative bags have no positive instances, leading to unstable results for complicated events.

9.3 LLP for Video Event Detection

We propose a learning from label proportions (LLP) based framework. Each video is treated as a bag, containing multiple instances. Hypothetically, for one event, if we know the proportion of positive instances for each video, the setting is exactly learning from label proportions, and the $\propto$SVM algorithm can be used to train a model to predict the binary label for each instance.

Unfortunately, in this setting, we only have a video-level event label, without knowing the exact proportions. Our assumption is that any positive video contains a large amount of positive instances, while any negative video contains few or none of them. To incorporate such supervision in the $\propto$SVM framework (8.5), we simply set the proportion of the positive bags to be 1, and the proportion of the negative bags to be 0. Because the label proportion loss is a "soft" loss function whose contribution is controlled by a weight C, this encourages large proportions of positive instances in positive videos while penalizes the

positive instances in negative videos. In practice, C can be tuned based on cross-validation.

One key question left unanswered is how to design the instances for each video. The instances can be frames, shots, video segments or even whole videos. Instances with different temporal lengths can be useful for recognizing different events. For example, "birthday party" can be identified by single frames containing cakes and candles, whereas sport-like actions such as "attempting board trick" and "parkour" are better detected by video segments characterizing actions.

Motivated by the observation, we consider instances of multiple granularities based on different length of time intervals. The feature representation of multiple-granular instances is obtained by pooling the local features into segment-level BoW with specific time lengths. Note that the video BoW is one special case in our framework. The original $\propto$SVM framework treats all instances equally and can not differentiate instances of multiple granularities. Therefore, we modify the formulation by including the weights for each instance. Formally, for the k-th bag, we modify the modeling of the proportion (8.20) as:

$$p_k(\mathbf{y}) = \frac{\sum_{i \in \mathcal{B}_k} \varpi_i y_i}{2 \sum_{i \in \mathcal{B}_k} \varpi_i} + \frac{1}{2}, \tag{9.1}$$

where y_i is the latent label of the i-th instance, and ϖ_i is the weight of the i-th instance. In this work, we simply set the weight to be the length of the segment. The optimization problem can be solved same as alter-$\propto$SVM. The only difference is that for a fixed $\mathbf{w}$, we flip the sign of the latent labels in a greedy fashion to find a sub-optimal $\mathbf{y}$. We terminate the optimization when the objective function is no longer decreasing.

9.4 Discussions

Event Detection at Video Level. In the previous section, we propose to learn an event detection model on the instance level, based on video-level labels. One intrinsic advantage of our method is that it can naturally discover the key evidence supporting the existence of a specific event. The top ranked 16 pieces of evidence selected by our method are shown in Figure 9.3 and Figure 9.6. Some selected single-frame instances are strong evidence, by which human can confirm the existence of target event by seeing those frames. In order to perform event detection on video level, we can first apply the instance classifier

on all instances of the test videos. The video-level detection score can then be obtained by performing a weighted average of all instance scores. Intuitively, a video containing more positive instances tend to have a higher probability of being positive. We will later show by experiment that our approach can lead to significant performance improvement for video-level event detection.

Learning with Heterogeneous Instances. In the previous section, we are considering the multiple granularities of instances with the same underlying feature representation. In practice, the instances may come with different representations. For example, we may have instances represented by image/audio/action features respectively. In such case, the proposed approach can be applied with minor changes to learn a classification model for each type of feature representation. We can also jointly learn the classification models with a modified objective function. We leave this task to our future work.

9.5 Experiments

Datasets. To evaluate our framework, we conduct experiments on three large-scale video datasets: TRECVID Multimedia Event Detection (MED) 2011, MED 2012 [156] and Columbia Consumer Videos (CCV) [94] datasets. All our experiments are based on linear kernel. In this chapter, we select SIFT [135] as underlying local features for initial evaluation. Note that our method can be easily extended to include multiple features by using fusion techniques. For example, we can train different instance-based detection models for each feature independently, and fuse detected scores of detectors using different features for final event detection. Additionally, by employing multiple features, we can discover unique cues, *e.g.* actions, colors, audio, for each video event.

Settings. For each video, we extract frames at every 2 seconds. Each frame is resized to 320×240 pixels, and the SIFT features are extracted densely with 10-pixel step size. The frame features are then quantized into 5,000 Bag-of-Word vectors. The frame-level SIFT BoWs are used as instance feature vectors. We evaluate four baseline algorithms on the dataset: mi-SVM, MI-SVM [5], video BOW, and $\propto$SVM with single frame instance.

Granularities	1	1 + 3	1 + 3 + 5	3 + all	1 + 3 + 5 + all	1 + all
Mean AP	0.39	0.38	0.37	0.41	0.43	0.41

Table 9.1: Mean APs of multi-granular instances combinations on CCV. The number represents the number of frames. "all" represents whole video instance.

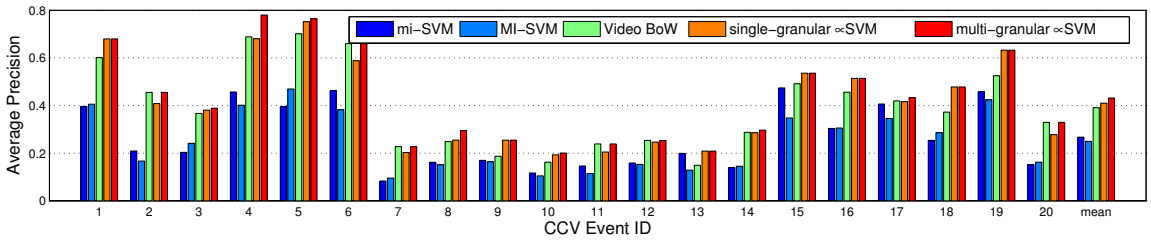


Figure 9.2: Experimental results of 20 complex events in Columbia Consumer Videos (CCV) dataset. The mean APs are 0.26 (mi-SVM), 0.25 (MI-SVM), 0.39 (Video BoW), 0.41 ($\propto$SVM) and 0.43 (multi-granular-$\propto$SVM).

9.5.1 Columbia Consumer Videos (CCV)

The Columbia Consumer Video (CCV) benchmark defines 20 events and contains 9,317 videos downloaded from YouTube. The event names and train/test splits can be found in [94]. We first evaluate different combinations of instance granularities. Table 9.1 shows the result with the combinations of single frame, 3-frame shot, and 5-frame shot and whole video. From the results, combining more granularities leads to better performance. However, increasing the number of granularities will cause higher computation cost. As a trade-off between time and performance, in the following experiments, we only use single-frame and whole video instances.

The experiment results on CCV are shown in Figure 9.2. The mi-SVM and MI-SVM are inferior to the standard video-level BoW method. This is due to the restrictive assumption of MIL, which focuses on searching one most representative instance in each video and treats all instances in a negative video as negatives. On the contrary, $\propto$SVM doesn't make this assumption and outperforms video BoW.

(a) Attempting bike trick (b) Rock climbing (c) Town hall meeting (d) Win race without vehicles

Figure 9.3: The top 16 key positive frames selected for the events in MED12. The proposed method can successfully detect important visual cues for each event. For example, the top ranked instances of "winning race without vehicles" are about tracks and fields.

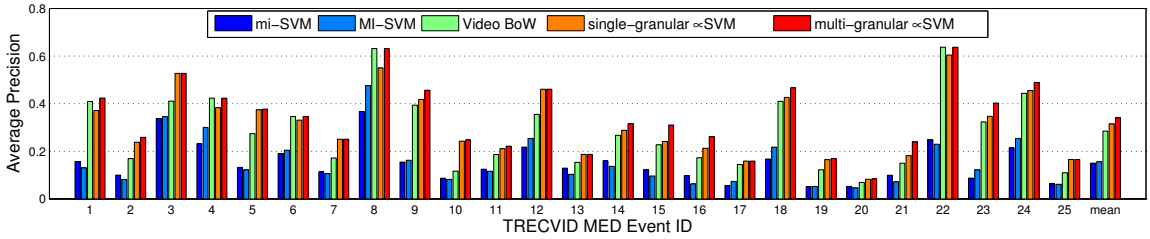


Figure 9.4: Evaluation results of 25 complex events in TRECVID MED 12 video dataset. The mean APs are 0.15 (mi-SVM), 0.16 (MI-SVM), 0.28 (Video BoW), 0.31 ($\propto$SVM) and 0.34 (multi-granular-$\propto$SVM).

9.5.2 TRECVID MED 12

The MED12 dataset contains 25 complex events and 5,816 videos. We use two-thirds of the data in training (3,878 videos) and the rest in testing (1,938 videos). The average number of extracted frames in each video is 79.4, and the average learning time of one event on a single Intel 2.53GHz core is around 40 minutes. The experimental results are shown in Figure 9.4. The conclusions are similar to those observed for the CCV dataset. The mi-SVM and MI-SVM are inferior to the standard video-level BoW method. The multi-granular $\propto$SVM outperforms video BoW by 21.4%.

As mentioned earlier, our method also offers benefits in pinpointing the specific local segments that signify the events. Figure 9.3 shows the automatically selected key frames in

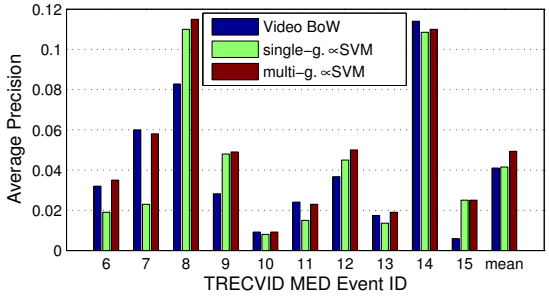


Figure 9.5: The APs from Event 6 to Event 15 in MED 2011.

videos that are detected as positive.

9.5.3 TRECVID MED11

There are three parts of the MED11 data: event collection (EC), the development collection (DEVT) and test collection (DEVO). We train the models based on EC and DEVO (2,680 + 10,403 videos), and evaluate on DEVT (32,061 videos). The average number of extracted frames per video is 59.8. The experimental results are shown in Figure 9.5. Because the DEVO set does not include any video of Event 1 to Event 5, only results of Event 6 to Event 15 are reported. The $\propto$SVM with single granularity outperforms Video BoW on "Flash mob gathering", "Getting vehicle unstuck", and "Parade", but produced worse results for other events. This is an interesting finding which confirms that instances of different lengths are needed for representing different events. Our method outperforms other methods by around 20% in this experiment. Some top-ranked frame instances learned by our method are shown in Figure 9.6.

9.6 Conclusion and Future Works

We proposed a novel approach to conduct video event detection by simultaneously inferring instance labels, and learning the instance-level event detection model. The proposed method considers multiple granularities of instances, leveraging both local and global patterns to achieve best results, as clearly demonstrated by extensive experiments. The proposed methods also provide an intuitive explanation of detection results by localizing the

(a) Attempting board trick (b) feeding animal (c) Landing a fish (d) Woodworking project

Figure 9.6: The top 16 key positive frames selected by our algorithm for some events in TRECVID MED11.

specific temporal frames/segments that signify the presence of the event.

For the future works, we are working on applying the learning setting into object detection with only image-level objective labels. This can be seen as an alternative way of discovering discriminative image patches for each category [101].

Chapter 10

Application: Attribute Modeling based on Category-Attribute Proportions

10.1 Introduction

Conventional attribute modeling requires expensive human efforts to label the attributes on a set of images. In this chapter, we apply the learning from label proportion setting in learning attribute based on category-attribute proportions. The framework requires no attribute labeling on the images. We refer the reader to Part III for a review of attribute-based approaches in computer vision. Different from methods in Part III, here we are modeling attributes with clear semantic names, with weak supervision.

Figure 10.1 illustrates our framework by a conceptual example of modeling the attribute "has TV". The input includes two parts:

- A multi-class image datasets of M categories, *i.e.*igned
- a set of images, each with a category label. Such datasets are widely available in various visual domains, such as objects, scenes, animals, human faces *etc.*
- An M-dimensional category-attribute proportion vector, where the i-th dimension of the vector characterizes the *proportion* of positive images of the attribute in the i-th category.

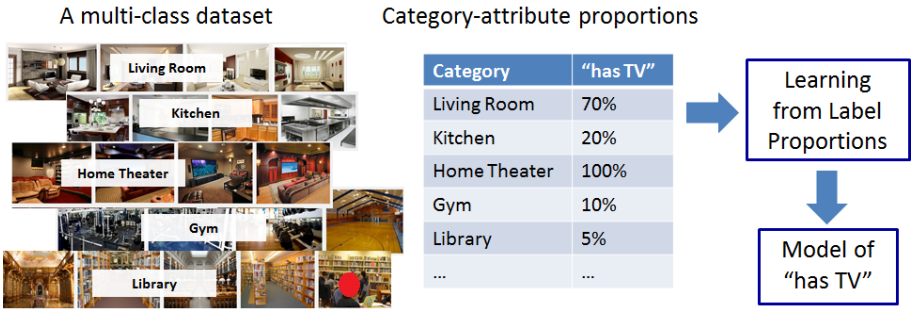


Figure 10.1: Illustration of the proposed framework for modeling the attribute "has TV". The input includes a multi-class dataset and a category-attribute proportion vector. The output is an attribute model to predict "has TV" for new images.

Given the above input, the attribute learning problem naturally fits the machine learning from label proportions (LLP) framework. We can then use the existing LLP techniques to train an attribute classifier, whose output can be used to predict whether the attribute is present in a new image. The framework requires no attribute labels on the images for training. Intuitively, it is more efficient to collect the category-attribute label proportions than image-level attribute labels. For example, based on statistics, or commonsense, "80% bears are black", "90% Asians are with black hair", and "70% living rooms have a TV".

Our work makes the following contributions. We propose a framework to model attributes based on category-attribute proportions, in which no image-level attribute labels are needed (Section 10.1). Finding the category-attribute proportions is still not a trivial task. To this end, we propose methods for efficiently estimating the category-attribute proportions based on category-attribute relatedness collected from different modalities, such as automatic NLP tools, and manual efforts with minimal human interactions (Section 10.3). The effectiveness of the proportion method is verified by various applications including modeling animal attribute, sentiment attributes, and scene attributes (Section 10.5).

10.2 LLP for Attribute Modeling

Our problem can be viewed as a learning with label proportion setting. Here the bags are defined by the M categories, each containing some corresponding images (instances).

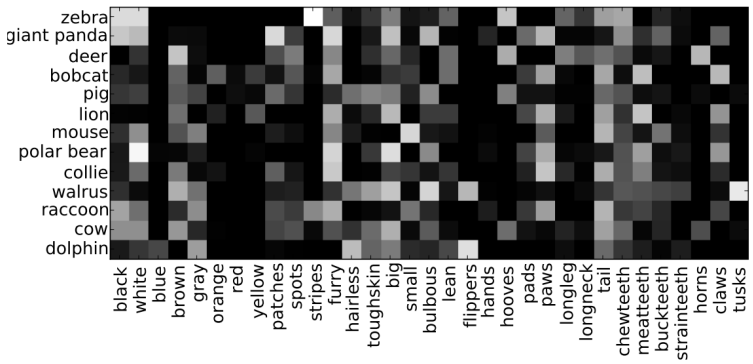


Figure 10.2: Manually defined category-attribute similarity matrix copied from [119]: the rows are the categories, and the columns are the attributes. This matrix is obtained from human judgments on the "relative strength of association" between attributes and animal categories.

The proportions are represented by a category-attribute proportion vector. The task is to model the attribute based on such information. In the case of modeling l attributes, we will need an $M \times l$ category-attribute proportion matrix, where the i-th column characterizes the proportion for the i-th attribute, $i = 1, \cdots, l$. For simplicity, the l attributes are modeled independently in this work. We apply the $\propto$SVM algorithm (Chapter 8) in solving this problem.

10.3 Collecting Category-Attribute Proportions

Collecting the *exact* category-attribute proportion is still a challenging problem. In this section, we propose several ways of estimating the proportions based on the category-attribute relatedness collected from different sources.

10.3.1 Human Knowledge

Perhaps the most straightforward method is to estimate the proportions based on human commonsense. For example, it is easy to know things like "80% bears are black", "100% home theater have TVs". To alleviate the bias of individual user, one can estimate the

Categories	Attributes	Source of Proportions	Evaluation Method
Animals	Animal visual properties	Commonsense by human	User study
Sentiment attributes	Sentiment attributes	Commonsense by ConceptNet	Test on a labeled set
Scenes	Scene properties	Borrowed from another dataset	Application-based (event detection)

Table 10.1: Summary of the three applications explored in our experiments.

proportion based on averaged value of multiple persons. In addition, to make the human interaction task easier, one can also discrete the proportion values, for example, into 0, 1/3, 2/3, 1.

A similar method has been used in modeling the category-level attributes in [119]. Figure 10.2 shows a subset of the category-attribute similarity matrix on the AwA dataset. [119] treats this matrix as a "visual similarity" matrix. When training the attributes, they binarize the matrix and treat the images of all positive categories as positive for the corresponding attribute. Unfortunately, the binarization will lead to huge information loss. Different from the above work, we treat this matrix as a proportion matrix. Based on the animal dataset provided in [119]. we will show by experiments (based on user study) that LLP provides better results than the binarization approach.

10.3.2 NLP Tools

To make the above commonsense based approach more efficient and scalable, we can also use NLP (Neural Language Processing) tools to automatically build such a category-attribute proportion matrix. For example, the ConceptNet[1] can be used to construct a category-attribute similarity matrix. The ConceptNet is a hypergraph that links a large amount of concepts (words) by the knowledge discovered from Wikipedia and WordNet, or provided by community contributors. The concept similarity can be computed by searching the shortest path in this graph. We apply the association function of the Web API of ConceptNet 5 to get the semantic similarity between the categories and attributes. After getting this semantic similarity matrix, we use the normalized similarity as the estimation of the proportions. We demonstrate this NLP tool based approach in the experiment section

[1] `http://conceptnet5.media.mit.edu/`

by modeling visual sentiment attributes.

10.3.3 Transferring Domain Knowledge and Domain Statistics

The proportions can also be borrowed from knowledge of other domains. For example, we can get the "black hair" proportion of different ethnic groups of people based on genetic research. And by combining such statistics with a multi-ethnic group human face dataset, we can model the attribute "black hair". In addition, the proportions can be borrowed from another dataset where the proportion is available. We will show one application of this approach, modeling scene attributes, in the experiment section.

10.4 Discussions

The proposed attribute modeling technique provides an efficient alternative to the classic approaches. However, there are some limitations. First, the estimated category-attribute proportion has to be close to the *exact* proportion of the dataset. This may not be true if the multi-class dataset is biased, or if the number of images in each category is small. Second, enough number of categories are needed. For example, the method cannot be applied to the worst case scenario where only a single category and a single category-attribute proportion value is available.

10.5 Experiments

We demonstrate the power of the proposed framework in three different applications: modeling animal attributes, modeling sentiment attributes, and modeling scene attributes. The three applications are summarized in Table 10.3.1. The parameters of the LLP algorithm are tuned based on cross-validation in terms of the proportion loss. The algorithm we use has the same computational complexity of linear SVM (it scales linearly to the number of images). In practice, the LLP algorithm is several times slower than linear SVM due to the alternating minimization process [231].

10.5.1 Modeling Attributes of Animals

Setting. Our first experiment consists in modeling animal attributes on the AwA dataset [119], which contains 30,475 images of 50 animal categories. Each image is uniquely labeled as one of the 50 categories. Associated with the dataset, there is a category-attribute similarity matrix of 85 attributes based on manual efforts mentioned in Section 10.3.1. A subset of the category-attribute matrix is shown in Figure 10.2. We use the same set of low-level features provided in [119]. In order to model the attributes, [119] first thresholds the matrix to a 0/1 matrix. They then train 85 attribute classifiers, where for each attribute, all the images belonging to the positive categories are treated as positive, and all images belonging to negative categories are treated as negative. The binarization step obviously leads to loss of information.

Method. In this work, we treat the similarity matrix as a category-attribute proportion matrix, and train the attribute models with $\propto$SVM. We use 50% images for training and 50% for testing.

Evaluation. As there are no labeled images for the 85 attributes, it is hard to directly compare our method with the baselines quantitatively. For evaluation, we perform a small-scale user study. For each attribute, 20 images are randomly selected from the top-100 ranked images for each method. Human subjects are then asked to determine which method produces better attribute modeling results. The subjects are 5 graduate students majoring in engineering and business, who are not aware of the underlying learning framework. In this experiment, the users prefer our results over the baseline ones 74% of the time. This verifies the plausibility of the newly proposed framework.

10.5.2 Modeling Sentiment Attributes

Setting. We consider the task of modeling object-based sentiment attributes such as "happy dog", and "crazy car". Such attributes are defined in [23, 34]. In this work we consider three nouns: dog, car, face and the sentiment attributes associated with them. This results in 77 sentiment attributes (or adjective-noun pairs, ANPs) to be modeled. Such ANPs appear widely in social media. We use the data and features provided in [23]: for each ANP, there is a set of images collected by querying that ANP on Flickr. Borth

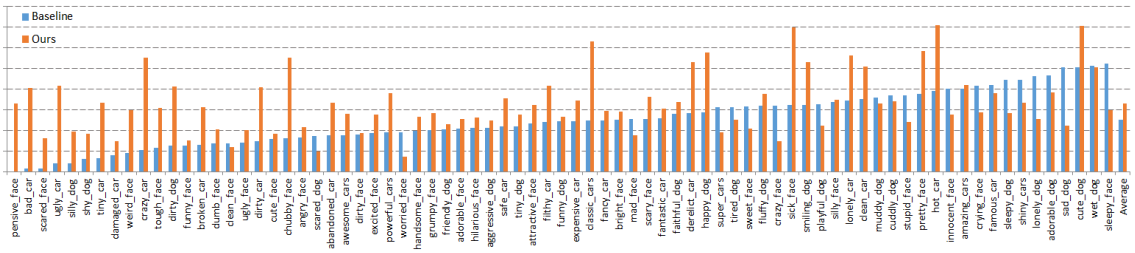


Figure 10.3: Experiment result of sentiment attribute modeling. The figure shows AP@20 of our method and the baseline (binary SVM). The AP is computed based on a labeled evaluation set.

et al. [23] uses such labels to train one-vs-all linear SVMs to model the ANPs. One critical problem for this approach is that many sentiment attributes are intrinsically ambiguous. For example, a "cute dog" can also be a "lovely dog". Therefore, when modeling "lovely dog", some images belonging to "cute dog" should also be positive.

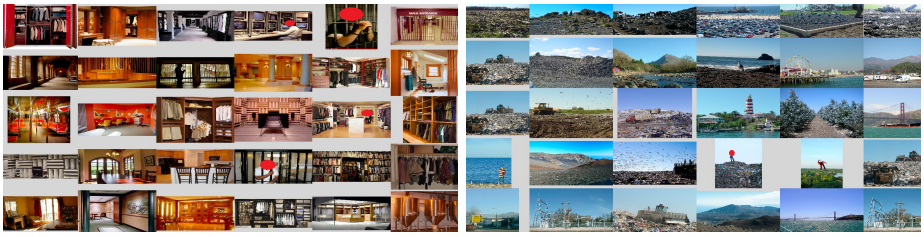
Method. To solve the above problem, we first use the method of Section 10.3.2 to collect the semantic similarity between every pair of ANPs. We then use our framework to model the ANPs. Different from other applications, both the categories and the attributes are ANPs. The proposed framework is used to improve the modeling of existing attributes, rather than learning new ones.

Evaluation. To evaluate the ANP modeling performance, for each ANP, we manually label 40 positive images, and 100 negative images from a separate set. Multiple people are involved in the labeling process, and images with inconsistent labels are discarded. Figure 10.3 compares our ANP modeling performance with [23]. Our approach dramatically outperforms the baseline for most of the sentiment attributes. Our method provides a relative performance gain of 30% in terms of the Mean Average Precision.

10.5.3 Modeling Scene Attributes

Setting. Concept classifiers have been successfully used in video event detection [28, 145]. In such systems, concept classifiers (about scenes, objects, activities *etc.*) are trained based on a set of labeled images visually related to the event detection task. The trained classifiers are used as feature extractors to obtain mid-level semantic representations of

(a) bathing (b) dry



(c) reading (d) rusty

Figure 10.4: Top ranked images of the learned scene attributes classifiers on IMARS.

video frames for event detection. The key to the above event detection paradigm is a set of comprehensive concepts. In order to model one additional concept, traditional approaches require an expensive manual labeling process to label the concepts on the images. The objective of this experiment is to model the 102 scene attributes defined by [163] with the IMARS images [28, 145], without the requirement of manual labeling. Some examples of the scene attributes are "cold", "dry", and "rusty". Existing IMARS concepts do not cover the 102 attributes.

Method. We first compute the empirical category-attribute proportions based on a separate multi-class scene datasets with 717 categories ("SUN attribute dataset") [163], in which each image comes with the attribute labels. Such proportions are then used on the IMARS set to model the attributes. For each of the 717 categories, we can find the corresponding set of images on IMARS based on the concept labels[2]. The supervised information is a 717 $\times$ 102-dimensional category-attribute proportion matrix. We then

[2]Note that one could train the attribute models based on SUN attribute dataset directly. But such attribute models do not lead to satisfactory result on IMARS due to cross-domain issues. Instead, the proportions of the two datasets are empirically very similar, and the proposed method leads to better performance.

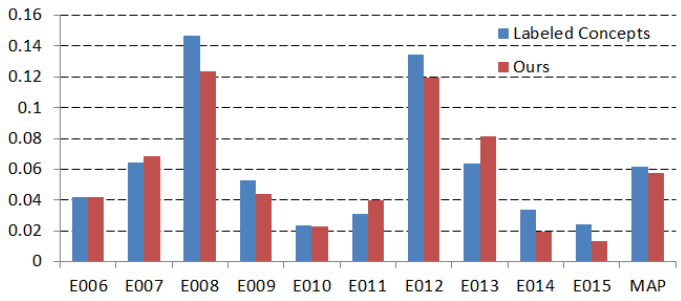


Figure 10.5: Event detection APs based on our attribute models, and the manual concept models. The modeled attributes (without manual labeling process) is competitive with the concepts (with manual labeling process).

train 102 $\propto$SVM classifiers to model the attributes.

Evaluation. Figure 10.4 visualizes a few modeled attributes by the top ranked images of IMARS. We can qualitatively see that the attribute classifiers can successfully capture the corresponding visual properties. We further apply the learned attribute models in the event detection task. The evaluation is based the TRECVID MED 2011 events with the evaluation set and pipeline described in [28, 145]. The baseline method is the concepts trained by manual labels. The attributes modeled by category-attribute proportions are very competitive compared with the manually labeled concepts in terms of the average precision on the event detection task. For certain events, *e.g.*, E007, E013, the performance of the attributes even outperforms the manual concepts. In summary, the proposed technique provides an efficient way of expanding IMARS concept classifiers for event detection.

10.6 Conclusion and Future Works

We proposed a novel framework of modeling attributes based on category-attribute proportions. The framework is based on a machine learning setting called learning from label proportions (LLP). We showed that the category-attribute proportion can be efficiently estimated by various methods. The effectiveness of the proposed scheme has been demonstrated by various applications including modeling animal attributes, sentiment attributes, and scene attributes. Our future direction is to study ways of jointly modeling all the

attributes. It is also worthwhile to study more ways of forming the category-attribute proportions, as well as combining such proportions formed by multiple knowledge sources.

Part III

Scalable Design and Learning of Mid-level Attributes

Chapter 11

Scalable Design and Learning of Mid-Level Attributes

11.1 Introduction

To address the limitation of supervised information, we have studied a weakly supervised learning setting called learning from label proportions (LLP) in Part II. The limitation of supervision can also be alleviated by knowledge transfer *i.e.*, leveraging knowledge from different, yet related tasks, In this part, we provide solutions from this perspective.

One widely used paradigm for knowledge transfer for visual data is attribute based recognition. There is no clear definition of visual attribute – generally speaking, they refer to a set of human nameable properties (*e.g.*, furry, striped, black for describing animals) that are shared across categories [148, 60]. Attributes have also been used to denote shareable properties of objects without concise semantic names (*e.g.*, dogs and cats have it but sharks and whales don't [60]) for improved discrimination. Attributes are sometimes referred to as *visual concepts* in the multimedia community [150, 222].

The idea of attribute-based recognition closely mimics human vision: humans are extremely good at recognizing novel categories with the help of knowledge learned from the past experience. For example, for an unseen animal, human can describe it by comparing it with other animals, and by associating it with visual properties characterizing body color, tail shape *etc.* Therefore, even with very few labeled images, human can build a recognition

model by relating the animal to some existing visual attributes. If the relationship of the visual attributes of the new animal is known, from Wikipedia, for example, a human can also recognize the animal without any training samples.

This is very different from conventional computer vision, where a large number of training samples with labels are given, and the algorithms try to find the sophisticated relationship between the low-level features and the high-level semantics. It is quite obvious that there is a huge gap between the low-level features and the high-level semantic meanings (this is commonly referred to as the "semantic gap"), and being able to describe novel categories by some mid-level attributes can help to bridge the "semantic gap", and greatly alleviate the amount of supervision required. As an intuitive example, recognizing "round" and "green" is helpful in recognizing apple. Therefore attribute-based recognition has received renowned attention in the computer vision community, and it has been applied in recognition with few or zero examples [119] and describing images containing unfamiliar objects [60]. In addition, visual attributes have also been used in applications such as image retrieval with intuitive multiple-attribute queries [190], video event recognition and recounting [28], action recognition [133], image-to-text generation [15], fine-grained visual categorization [55], and classification with humans-in-the-loop [161, 25].

Ironically, although attributes based recognition, by design, is a way to improve the scalability of visual recognition with limited supervision, conventional attribute modeling suffers from scalability issues in itself. On the one hand, designing attributes usually involves manually picking a set of words that are descriptive for the images under consideration, either heuristically [60] or through knowledge bases provided by domain specialists [119]. On the other hand, after deciding the set of attributes, additional human efforts are needed to label the attributes on a set of images to train the attribute classifiers. To make attribute-based recognition applicable to large-scale visual data, it is required to have a high-dimensional attribute space which is sufficiently expressive for the visual world. Unfortunately, the required human efforts in both the attribute design and labeling process hinder scaling up the process to develop a large number of attributes. In addition, a manually defined set of attributes (and the corresponding attribute classifiers) may be intuitive but not discriminative (therefore useful) for the visual recognition task. A challenging question to answer

is how to design a large set of mid-level attributes which is useful for the visual recognition task, as well as easy and cheap to be constructed.

In the thesis, we propose approaches to design and model a large set of useful mid-level visual attributes without human burdens. In particular, we consider designing the attributes for two tasks: image retrieval with multi-attribute queries [229], and attribute-based recognition [230]. For both of the applications, we propose a method which can automatically design and learn a large set of useful attributes which are weakly interpretable. Key to both the proposed methods is to leverage the existing supervised information, as well as considering the data distribution and the task to make the designed attributes discriminative.

11.2 Related Works

11.2.1 Collection of Supervision based on Crowdsourcing

Perhaps the most straightforward way of learning attribute models (and also for solving the weakly supervised learning problem) is by scaling up the labeling process which often requires human involvement. For example, Amazon Mechanic Turk, a crowdsourcing system, has been widely used in the computer vision community to collect supervised information, such an image labels and object bounding boxes [194, 47]. Crowdsourcing has also been used to label attributes on the images [163]. Innovative methods, such as reCAPTCHA [210], was designed to collect human-based recognition on characters when performing the CAPTCHA test. Certain types of interactive games were proposed such that human can help labeling images (for free) while being entertained [209, 17]. Different from the above, we study approaches which require no human labeling process.

11.2.2 Transfer Learning

The method of visual attribute modeling is a widely used paradigm for knowledge transfer in computer vision. In machine learning, there is a group of related methods, which are generally referred to as transfer learning. There are many other names of transfer learning, such as life-long learning, learning to learn, cumulative learning, multi-task learning, and

inductive learning. Similar to visual recognition, the more general transfer learning also mimic the human learning process: knowing how to play piano can certainly help to learn playing violin, and learning to speak and to write multiple languages can reinforce each other. We refer the reader to [158] for a survey on transfer learning. In that, transfer learning methods are categorized into three classes: inductive learning, where the source and the target tasks are different, transductive transfer, where the source and target tasks are the same, but in different domains, and unsupervised transfer, where the target task is unsupervised. Attribute-based recognition can be understood as a mixture of transductive transfer and inductive transfer, depending on the settings and formulation.

11.2.3 Designing Semantic Attributes

Conventionally, the attributes are designed by manually picking a set of words that are descriptive for the images under consideration [60, 119, 136]. Similar way has also been explored for designing "concepts" in multimedia [150, 222] and computer vision [126, 203, 202]. The concepts are sometimes manually or automatically organized into a hierarchical structure (ontology) to characterize different levels of semantics [150, 202]. To alleviate the human burdens, Berg et al. [15] proposed to automatically discover attributes by mining the text and images on the web, and Rohrbach et al. [176] explored the "semantic relatedness" through online knowledge source to relate the attributes to the categories. In order to incorporate discriminativeness for the semantic attributes, Parikh and Grauman [160], Duan et al. [55] proposed to build nameable and discriminative attributes with human-in-the-loop. Compared with the above manually designed semantic attributes, our designed attributes cannot be used to describe images with concise semantic terms, and they may not capture subtle non-discriminative visual patterns of individual images. However, the proposed weak attributes and category-level attributes can be automatically and efficiently designed for discriminative visual recognition and retrieval, leading to effective solutions and even state-of-the-art performance on the tasks that were traditionally achieved with semantic attributes.

11.2.4 Designing Data-Driven Attributes

Non-semantic "data-driven attributes" have been explored to complement semantic attributes with various forms. Kumar et al. [114] combined semantic attributes with "simile classifiers" for face verification. Yang and Shah [226] proposed data-driven "concepts" for event detection. Liu et al. [133] extended a set of manually specified attributes with data-driven attributes for improved action recognition. Sharmanska et al. [188] extended a semantic attribute representation with extra non-interpretable dimensions for enhanced discrimination. Bergamo et al. [16], Gordoa et al. [75], Rastegari et al. [174] used the large-margin framework to model attributes for objective recognition. Wang and Mori [217], Farhadi et al. [61] used attribute-like latent models to improve object recognition. For both of the proposed approaches, we utilize the data distribution to make the attribute more discriminative for the tasks, including multi-attribute image retrieval, and attribute-based recognition.

11.3 Overview of the Proposed Approaches

11.3.1 Weak Attributes for Large-Scale Image Retrieval

The attribute-based query offers an intuitive way of image retrieval, in which users can describe the intended search targets with understandable attributes. In this chapter, we develop a general and powerful framework to solve this problem by leveraging a large pool of weak attributes comprised of automatic classifier scores or other mid-level representations that can be easily acquired with little or no human labor. We extend the existing retrieval model of modeling dependency within query attributes to modeling dependency of query attributes on a large pool of weak attributes, which is more expressive and scalable. To efficiently learn such a large dependency model without overfitting, we further propose a semi-supervised graphical model to map each multi-attribute query to a subset of weak attributes. Through extensive experiments over several attribute benchmarks, we demonstrate consistent and significant performance improvements over the state-of-the-art techniques. The work was originally presented in [229].

11.3.2 Designing Category-Level Attributes for Visual Recognition

Attribute-based representation has shown great promises for visual recognition due to its intuitive interpretation and cross-category generalization property. However, human efforts are usually involved in the attribute designing process, making the representation costly to obtain. In this chapter, we propose a novel formulation to automatically design discriminative "category-level attributes", which can be efficiently encoded by a compact category-attribute matrix. The formulation allows us to achieve intuitive and critical design criteria (category-separability, learnability) in a principled way. The designed attributes can be used for tasks of cross-category knowledge transfer, achieving superior performance over well-known attribute dataset Animals with Attributes (AwA) and a large-scale ILSVRC2010 dataset (1.2M images). This approach also leads to state-of-the-art performance on the zero-shot learning task on AwA. This work was originally introduced in [230].

Chapter 12

Weak Attributes for Large-Sale Image Retrieval

12.1 Introduction

In this chapter, we propose the notion of "weak attributes", under the application of large-scale image retrieval with multi-attribute queries. In such a scenario, the user provides multiple attributes, to describe the facets of the query target. For instance, to retrieve images of a person, one could describe the physical traits of gender, hair color, and presence of mustache *etc.* The task is to retrieve images containing all of the query attributes. We assume only a small portion of the database have the attributes labeled before hand, and our goal is to search the entire large-scale image corpus.

A straightforward solution to the above problem is to build classifiers for the attributes of interest and sum the independent classifier scores to answer such multi-attribute queries *e.g.* [113]. A promising alternative, as shown in [190], is to analyze the dependencies/ correlations among query attributes and to leverage such multi-attribute interdependence to mitigate the noise expected from the imperfect automatic classifiers. An illustrative example of the above dependency model is shown in Figure 12.1. The previous work [190] relied only on the pre-labeled attributes to design the dependency model, limiting its performance and scalability. On the one hand, user labeling is a burdensome process; On the other hand, the number of such pre-labeled attributes is limited: only a small set of words were chosen, for

instance "car", "tree", "road" *etc.* for street scenes and "beard", "Asian", "male" *etc.* for human faces. In particular, there are only 27 attributes for face images considered in [190], and 64 attributes in a-PASCAL benchmark [60]. Such a small amount of attributes are far from sufficient in forming an expressive feature space, especially for searching a large image corpus of diverse content.

In this chapter, a **Weak Attribute** based paradigm is proposed to address the above challenge, and it provides a principled solution for large-scale image retrieval using multi-attribute queries:

Weak Attributes *are a collection of mid-level representations, which could be comprised of automatic classifier scores, distances to certain template instances, or even quantization to certain patterns derived through unsupervised learning, all of which can be easily acquired with very little or no human labor.*

We specifically refer to such attributes as "weak" because they could be easily acquired, unlike the classifiers specially trained for attributes that require dedicated human labeling efforts[1]. For example, hundreds or thousands of visual classifiers such as classemes [203], Columbia374 [222], and automatic attributes [15], have been developed and made available, though they typically do not have direct correspondence with the target query attributes. Different from query attributes, weak attributes may not have clear semantic meanings. Examples are discriminative attributes [60] (A is more like a dog than a cat); relative attributes [159] (A is more natural than B); comparative values of visual features [114] (A is similar to B); and even topic models.

We are interested in the case that the dimensionality of weak attributes (say thousands or more) is much higher than that of the query attributes. The large cardinality ensures the weak attribute space to be sufficiently expressive, based on which a robust retrieval model can be developed. As shown in Figure 12.1, to bridge the gap between the limited query attribute (for user) and the expressive weak attribute space (for the machine), we extend the attribute dependency model from the narrow one *among query attributes only*

[1]Scores of the attribute classifiers, which are trained based on labeled training data existed before hand, can also be treated as weak attributes, for the reason that they are easily acquired without additional human labors.

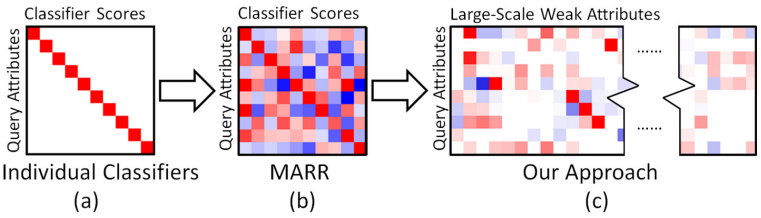


Figure 12.1: "Evolution" of approaches for answering multi-attribute queries: (a) Direct independent matching of query attributes with corresponding classifiers. (b) Modeling dependency/correlation within query attributes (MARR) [190]. (c) Modeling dependency of query attributes on a large pool of weak attributes. Our approach also emphasizes sparsity of the model to avoid overfitting.

to a much more general one that maps *query attributes to weak attributes*, as detailed in Section 12.3.1.

Learning a large-scale dependency model with a small number of training labels is not trivial, due to the complexity of the learning process and the risk of overfitting. To address the above issues, we propose to impose query dependent "sparsity" of the dependency model, so that for both training and prediction, only a limited number of weak attributes are considered for each multi-attribute query (Section 12.3.2). To achieve this goal, we further develop a novel semi-supervised graphical model to incorporate statistics from both the labeled training data and the large amount of unlabeled data (Section 12.3.3). We will demonstrate through extensive experiments that our approach improves significantly over existing methods (Section 12.4), and can largely boost the flexibility of the retrieval model in dealing with cross-dataset variants (Section 12.4.2) and large-scale scenarios (Section 12.4.3). Our work has four unique contributions:

- We propose *weak attributes* that unify various kinds of mid-level image representations which can be easily acquired with no or little human labor.
- We apply weak attributes to image retrieval, by modeling dependency of *query attributes to weak attributes* under the framework of structural learning.
- To achieve efficiency and avoid overfitting, we propose to control query dependent sparsity with a novel *semi-supervised graphical model*. This further enables our approach to be effective for cross-dataset and large-scale scenarios.

- We compile a large multi-attribute image retrieval dataset, named a-TRECVID, including 126 fully labeled query attributes and 6,000 weak attributes of 0.26 million images extracted from videos used in the 2011 TRECVID Semantic Indexing (SIN) track.

12.2 Related Works

Structural Learning. Tsochantaridis et al. [204] introduced structural SVM to address classifier with structural output. Structural SVM has been well advocated in document retrieval [122] (sometimes also referred as "learning to rank" in the information retrieval literature) and multi-label learning [167] *etc.* This is due to its capability to directly incorporate different types of loss distortions, such as precision, F_β score and NDCG, into its optimization objective function. Siddiquie et al. [190] proposed to handle the multi-attribute query based image retrieval problem using a structural learning paradigm [122, 167], by considering interdependency across query attributes.

Multi-Keyword Queries. Besides the utilization of attributes, there have been works on image search based on multi-keywords queries [62, 109], in which images are first retrieved based on textual features and then reranked by classifiers built on visual features. The main limitation of these works is the requirement that every image in the database is tagged by hand or automatic yet imperfect classifiers. This limitation could be mitigated by tag propagation [77, 79]. However, the above works do not take into account the dependencies between query terms, which were shown to largely affect the search performance [190]. Our work is also related to "query expansion" for multi-keyword queries [151]. Different from query expansion, our approach "expands" a query to a sparse subset of weak attributes from a large pool, and the final dependency model is jointly learned across all possible queries under the framework of structural learning.

12.3 Weak Attributes for Image Retrieval

In the weak attribute based retrieval paradigm, we first select a sparse subset of weak attributes for each multi-attribute query, through a novel semi-supervised graphical model (Section 12.3.3). The selection process is optimized under a formulation of maximizing

$\mathcal{Q}$	Set of attribute queries
Q	A multi-attribute query
q	A query attribute
q_i	The i-th query attribute, $i = 1, \cdots, \lvert\mathcal{Q}\rvert$
$\mathcal{X}$	Set of all the images
X	A set of images
$\mathbf{x}$	An image
$\mathcal{Z}$	Set of all weak attributes
z	A weak attribute
z_i	The i-th weak attribute, $i = 1, \cdots, \lvert\mathcal{Z}\rvert$
$z_i(\mathbf{x})$	Score of the i-th weak attribute for image $\mathbf{x}$
$H(\cdot)$	Entropy of a probability distribution
$\mathcal{I}(;)$	Mutual information of two probability distributions

Table 12.1: Key Notations of Chapter 12.

mutual information of query attributes and weak attributes (Section 12.3.2). Then, for each multi-attribute query, only the corresponding subset of weak attributes are considered in the learning and prediction process (Section 12.3.1), ensuring efficiency and avoiding overfitting. We begin this section by first introducing the proposed dependency modeling of query attributes to weak attributes under the structural SVM framework. Table 12.3 shows the key notations of this chapter.

12.3.1 Dependency Modeling

Our dependency modeling is based on an image retrieval scenario. Similar to [190], it can be easily modified for the image ranking scenario, by some minor changes with an appropriate query relevance function.

Prediction. Let $Q \subset \mathcal{Q}$ be a multi-attribute query, where $\mathcal{Q}$ is the set of all possible query attributes. All the possible query attributes are $\{q_i\}_{i=1}^{|\mathcal{Q}|}$. Let $\mathcal{Z} = \{z_i\}_{i=1}^{|\mathcal{Z}|}$ be the set of all weak attributes, and let $\mathcal{X}$ be the set of images. The multi-attribute search is to select

a set of images as the structured response to a given multi-attribute query Q:

$$\underset{X \subset \mathcal{X}}{\operatorname{argmax}} \mathbf{w}^T \psi(Q, X), \tag{12.1}$$

where[2]

$$\mathbf{w}^T \psi(Q, X) = \sum_{i=1}^{|\mathcal{Q}|} \sum_{j=1}^{|\mathcal{Z}|} I(q_i \in Q) w_{ij} \sum_{\mathbf{x} \in X} z_j(\mathbf{x}). \tag{12.2}$$

Here, $I(q_j \in Q)$ is the indicator function which outputs 1 if the j-th attribute is in the query, and 0 otherwise. $z_j(\mathbf{x})$ measures the score of the j-th weak attribute of image $\mathbf{x}$. w_{ij} is the dependency of the i-th query attributes to the j-th weak attribute. Compared with the recent work [190], our key insight is to model the dependency of query attributes on weak attributes characterized by $\mathbf{w}$, as illustrated earlier in Figure 12.1. (12.1) can be solved efficiently in $\mathcal{O}(|\mathcal{X}|)$.

Training. In training, the learner receives N multi-attribute query image set pairs: $\{(Q_i, X_i)\}_{i=1}^N$. We follow the standard max-margin training formulation as follows:

$$\begin{aligned} \underset{\mathbf{w}, \xi}{\operatorname{argmin}} \quad & \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^{N} \xi_i \\ \text{s.t.} \quad & \mathbf{w}^T (\psi(Q_i, X_i) - \psi(Q_i, X)) \geq \triangle(X_i, X) - \xi_i, \quad X \subset \mathcal{X}. \end{aligned} \tag{12.3}$$

$\triangle(X_i, X)$ is the loss function, which can be set as Hamming loss, precision, recall and F_β score *etc.* [190, 167]. (12.3) can be solved by cutting plane method [204]. It iteratively solves (12.3) initially without any constraints, and then at each iteration adds the most violated constraint of the current solution. The most violated constraint at each iteration is generated by:

$$\underset{X \subset \mathcal{X}}{\operatorname{argmax}} \triangle(X_i, X) + \mathbf{w}^T \psi(Q_i, X), \tag{12.4}$$

which can be solved in $\mathcal{O}(|\mathcal{X}|)$ with the Hamming loss, and $\mathcal{O}(|\mathcal{X}|^2)$ with loss such as the F_β score [167].

---

[2]For easier presentation, $\mathbf{w}$ is written as matrix form here: $\mathbf{w} \in \mathbb{R}^{|\mathcal{Q}| \times |\mathcal{Z}|}$, where w_{ij} is the dependency model of the i-th query attributes to the j-th weak attribute.

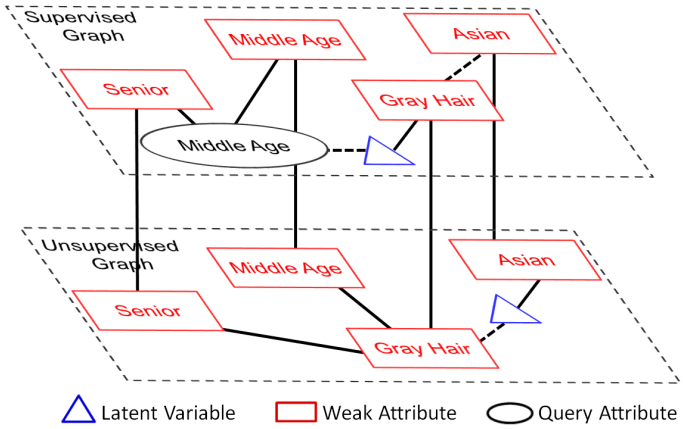


Figure 12.2: Semi-supervised graphical model (the dotted line means connected through other nodes). In this example, the semi-supervised graphical model suggests weak attribute "gray hair" is related to query attribute "middle age". This relation is unclear by considering the supervised graph alone. Note that the latent nodes in the unsupervised graph are also a kind of weak attribute.

12.3.2 Controlling Query Dependent Sparsity

For a large weak attribute pool, the model ($\mathbf{w}$ in (12.1)) contains a large number of parameters: $|Q| \times |\mathcal{Z}|$ (500,000 if there are 100 query attributes and 5,000 weak attributes). It is computationally expensive in learning, and may also cause overfitting.

We solve the above issues by imposing query dependent sparsity on the dependency model. Given a query $Q \subset \mathcal{Q}$, the objective is to get a small set of weak attributes $Z_Q \subset \mathcal{Z}$ relevant to Q, so that for both training (12.3) and testing (12.1), only the corresponding elements of $\mathbf{w}$ are considered:

$$\mathbf{w}^T \psi(Q, X) = \sum_{i=1}^{|\mathcal{Q}|} \sum_{j=1}^{|\mathcal{Z}|} I(q_i \in Q) I(z_j \in Z_Q) w_{ij} \sum_{\mathbf{x} \in X} z_j(\mathbf{x}). \tag{12.5}$$

This idea is important, since, from both intuition and the experiment results which will be presented later, only a small subset of weak attributes in the large weak attribute pool are related to a specific multi-attribute query.

We formulate the above weak attribute selection problem as maximizing mutual infor-

mation, which is a general measurement of relevance:

$$\underset{Z_Q \subset \mathcal{Z}}{\operatorname{argmax}} \mathcal{I}(Q; Z_Q) \quad s.t. \quad |Z_Q| = \vartheta, \tag{12.6}$$

where ϑ is the desired number of selected weak attributes controlling the query dependent sparsity of our model. We call ϑ "sparsity" in this work. General feature selection methods based on entropy criterions can be found in [165]. (12.6) is hard to solve, because there are $\binom{|\mathcal{Z}|}{\vartheta}$ combinations in selecting $Z_Q \subset \mathcal{Z}$. We instead use a greedy algorithm by considering $z \in \mathcal{Z}$ one at a time, and set Z_Q as the top-ϑ z with highest mutual information value:

$$\underset{z \in \mathcal{Z}}{\operatorname{argmax}} \mathcal{I}(Q; z), \tag{12.7}$$

where $I(Q; z) = H(Q) - H(Q|z)$. $H(Q)$ is a constant for a given query Q. $H(Q|z)$ can be expanded as:

$$H(Q|z) = -\sum_{z} \mathbb{P}(z) \sum_{Q} \mathbb{P}(Q|z) \log \mathbb{P}(Q|z), \tag{12.8}$$

where $\mathbb{P}(z)$ is the marginal distribution of weak, attribute z. $\mathbb{P}(Q|z)$ plays a key role in bridging each weak attribute $z \in \mathcal{Z}$ to a specific multi-attribute query Q. Direct modeling of $\mathbb{P}(Q|z)$ based on training data that has query attribute ground truth is easy. However, it will bias the model because the training set can be very small. To address the above issues, we design a novel semi-supervised graphical model, to get $\mathbb{P}(Q|z)$ efficiently with consideration of statistics from both training and an unlabeled auxiliary set. This further enables our approach to be effective for cross-dataset and large-scale retrieval tasks. Due to the utilization of unlabeled set, we term our method "semi-supervised graphical model" in the next section.

12.3.3 Semi-Supervised Graphical Model

The Model. The semi-supervised graphical model (Figure 12.2) is a two-layer probability graphical model[3]. The first layer is called the *supervised graph*, which is constructed based on the training data with the ground truth of query attributes. This graphical model

[3]Before learning this model, weak attributes are binarized in order to use the same discrete format as the query attributes.

Algorithm 6 Alternating Inference

Given a query $Q \subset \mathcal{Q}$, compute $\mathbb{P}(Q|z_i = 1)$, $i = 1, \cdots, |\mathcal{Z}|$, as needed in (12.8). We consider a fixed i, without loss of generality.

while Not convergent (the change of $\mathbb{P}_{up}(z)$ is large) **do**

Inference on the unsupervised graph to get its marginal distribution $\mathbb{P}_{down}(z|z_i = 1), \forall z \in \mathcal{Z}$.

Update the margin of $z \in \mathcal{Z}$ of the supervised graph $\mathbb{P}_{up}(z) \leftarrow \mathbb{P}_{down}(z|z_i = 1)$.

Inference on the supervised graph, to get updated $\mathbb{P}_{up}(z), \forall z \in \mathcal{Z}$.

Update the margin of $z \in \mathcal{Z}$ of the unsupervised graph: $\mathbb{P}_{down}(z) \leftarrow \mathbb{P}_{up}(z)$.

end while

Compute joint distribution $\mathbb{P}_{up}(Q)$ in the supervised graph as output $\mathbb{P}(Q|z_i = 1)$.

is to characterize the joint distribution of query attributes and weak attributes on the training set. The second layer is called the *unsupervised graph*, which is constructed based on all or subset of an unlabeled set, with no query attribute ground truth available. This graphical model is to characterize the joint distribution of weak attributes on the large unlabeled set.

The two layers are connected by weak attributes that appear in both layers. Therefore, the model can leverage information from both the labeled training data and the unlabeled data, and thus we call it "semi-supervised". The graphical model can capture the high-order dependency structure of attributes. It greatly improves the generalization power of our approach, and enables applications of cross-dataset retrieval (Section 12.4.2), and large-scale retrieval with very small amount of training data (Section 12.4.3).

We choose latent tree [38] as our graphical model in each layer. Tree models are a class of tractable graphical models, efficient in inferencing, and widely used in prior works of context modeling and object recognition [217]. The learned latent variables can be treated as additional weak attributes, which summarize higher level information (context) of both weak and query attributes.

Alternating Inference. We now talk about how to get $\mathbb{P}(Q|z), \forall z \in \mathcal{Z}$ of (12.8) from the above semi-supervised graphical model. In other words, we need to model $\mathbb{P}(Q|z_i), i = 1, \cdots, |\mathcal{Z}|$ (recall that $\mathcal{Z} = \{z_i\}_{i=1}^{|\mathcal{Z}|}$). The direct inference is difficult because the graphical

model may not be acyclic and thus intractable. To address the above issue, we have designed Alternating Inference, summarized in Algorithm 6. We use $\mathbb{P}_{up}$ to denote the probability on the supervised graph, and use $\mathbb{P}_{down}$ to denote the probability on the unsupervised graph. The idea is to iteratively inference on the unsupervised and supervised graph. In each iteration, margins of the weak attributes are estimated and passed to the other graph. We set the initial marginal probability of query attributes as uniform. The inference over each layer can be done efficiently by belief propagation.

To compute the joint probability $\mathbb{P}_{up}(Q)$ in the last step of Algorithm 6, we rewrite it as the product of conditional probabilities. For example, if query $Q = \{q_1, q_2, q_3\}$, $\mathbb{P}_{up}(Q) = \mathbb{P}_{up}(q_1|q_2, q_3)\mathbb{P}_{up}(q_2|q_3)\mathbb{P}_{up}(q_3)$. It is easy to show that, in general, we can get $\mathbb{P}_{up}(Q)$ by doing belief propagation $2^{|Q|} - 1$ times, which is small given the fact that $|Q|$ is small. This assumption is valid for the reason that most images are only associated with a limited number of query attributes. For instance, the average number of attributes of LFW and a-PASCAL datasets is 6.95 and 7.1 respectively, while this number for a-TRECVID dataset is only 2.6. In our configuration, we set $|Q| \leq 3$.

We found empirically that the convergence of Algorithm 6 is fast, usually within less than 10 iterations. Therefore, weak attribute selection based on (12.7) can be computed efficiently.

12.4 Experiments

Our implementation of structural SVM is based on [98] with its Matlab wrapper[4], under the 1-slack formulation. We use regCLRG [38] to learn the latent tree graphical model for each layer of the semi-supervised graphical model. This method is found to be effective in terms of both efficiency and performance. Following [190], Hamming loss for binary classification is used as the loss function throughout the experiments:

$$\triangle(X_i, X) = 1 - \frac{|X \cap X_i| + |\bar{X} \cap \bar{X}_i|}{|\mathcal{X}|}. \tag{12.9}$$

Our evaluation is based on mean AUC (Area Under Curve), which is a standard measurement commonly used to evaluate the performance of binary classification tasks, in our case,

[4] `http://www.vlfeat.org/~vedaldi/code/svm-struct-matlab.html`

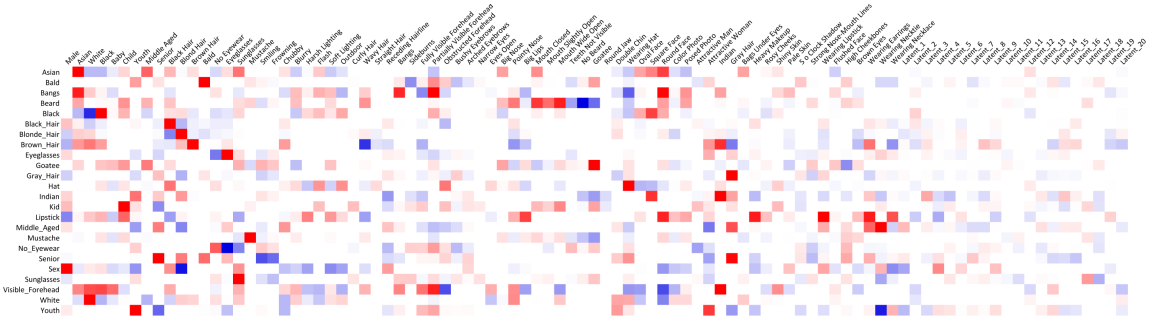


Figure 12.3: Learnt $\mathbf{w}$ for LFW dataset with sparsity $\vartheta = 40$ (best viewed in color). Vertical labels are query attributes, and horizontal labels are weak attributes. Red, blue and white represent positive, negative and no dependency respectively. High intensity means high dependency level. This learned matrix is semantically plausible. For instance, people "wearing lipstick" (query) is unlikely to be "male" (weak), and "kid" (query) is highly related to "child" (weak). Compared with [190], our method avoids dubious artifacts in mappings, *e.g.* "Asian" (query) to "male" (weak) and "black" (query) to "youth" (weak), and also results in faster training/testing and less overfitting.

image retrieval. Note that the AUC measure of a random guess system is 0.5.

12.4.1 Labeled Faces in the Wild (LFW)

Our first evaluation is on the Labeled Faces in the Wild (LFW) dataset [87], which contains 9,992 images with manual annotations of 27 attributes, including "Asian", "beard", "bald", "gray hair", *etc.* Following the setting of [190], we randomly choose 50% of this dataset for training and the rest for testing. For visualization, the weak attributes for this dataset contain only attribute classifier scores from [114] (scores of 73 attribute classifiers designed for human faces) and latent variables from the graphical model[5].

In order to get the baseline results of individual classifiers (direct classifiers corresponding to the query attributes), we have omitted three attributes: "long hair", "short hair" and "no beard" which are not covered by the classifier scores from [114]. Figure 12.3 shows

[5]The value of latent variables are acquired by inferencing on the unsupervised graph, conditioned on the weak attributes. We have transformed the conditional marginal distribution of latent variables back to the real interval $[-1, 1]$ by linear mapping.

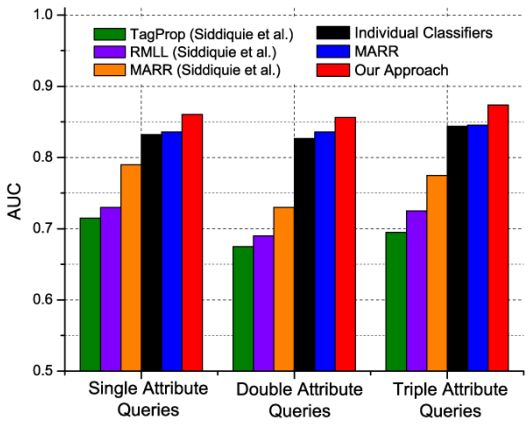


Figure 12.4: Retrieval performance on LFW dataset. The first three results are copied from [190], in which different individual classifiers are used. The last three results are based on our implementation.

the learnt dependency model $\mathbf{w}$ using our proposed method with sparsity $\vartheta = 40$. For visualization, we only show the weak attributes selected by single-attribute queries, *i.e.* each query attribute is uniquely mapped to 40 weak attributes only. Note the sparsity considered in our model is query specific, rather than a fixed sparsity structure across different queries (Section 12.3.2). For example, for a single query "Asian", the selected weak attribute might be "Asian" and "white", while for a double attribute query "Asian"+"woman", the selected weak attributes could be "Asian" and "male". In both learning and prediction processes involving the dependency model, only the selected weak attributes will be considered for each multi-attribute query.

Figure 12.4 shows the comparisons of our method to several existing approaches, including TagProp [79], Reverse Multi-Label Learning (RMLL) [167], Multi-Attribute based Ranking and Retrieval (MARR) [190], individual classifier scores from [114], and our implementation of MARR based on individual classifier scores. Our method outperforms all the other competing methods consistently for all types of queries including single, double and triple attributes. It is interesting to note that in this experiment, the weak attributes are actually not "weak", in the sense that even individual classifiers outperform TagProp, RMLL and MARR reported in [190], in which different individual classifiers are used. The

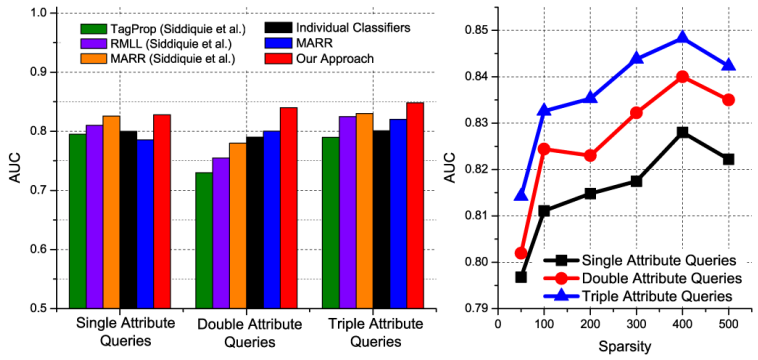


Figure 12.5: Retrieval performance on a-PASCAL dataset. Left: AUC comparison based on optimal sparsity ($\vartheta = 400$). The first three results are copied from [190]. Right: AUC of our approach with varying sparsity.

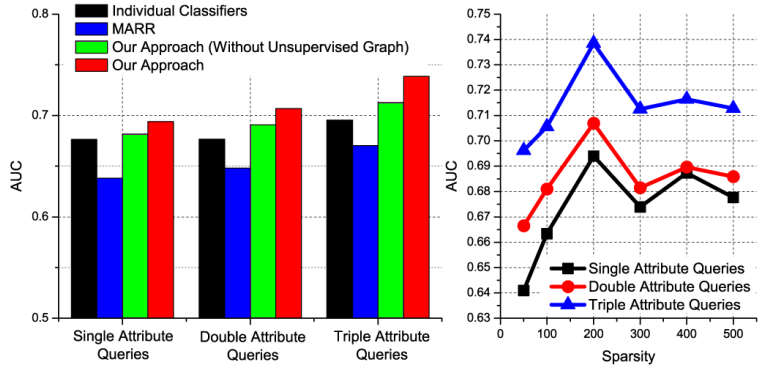


Figure 12.6: Retrieval performance on a-Yahoo dataset. Left: AUC comparison based on optimal sparsity ($\vartheta = 200$). Right: AUC of our approach with varying sparsity.

weak attributes are *weak* in the sense that they are trained from other sources, therefore not directly related to the specific dataset at hand. Closely following [190], our implementation of MARR only slightly outperforms individual classifiers: the stronger baseline reduces the amount of improvement that can be obtained by utilizing dependency information from within query attributes only.

12.4.2 a-PASCAL and a-Yahoo

Another dataset a-PASCAL [60] contains 12,695 images (6,340 for training and 6,355 for testing) collected from the PASCAL VOC 2008 challenge[6]. Each image is assigned one of the 20 object class labels: people, bird, cat, *etc.* Each image also has 64 binary query attribute labels, such as "round", "head", "torso", "label", "feather" *etc.* a-Yahoo is collected for 12 object categories from the Yahoo images search engine. Each image in a-Yahoo is described by the same set of 64 attributes, but with different category labels compared with a-PASCAL, including wolf, zebra, goat, donkey, monkey *etc.*

Following the setting of [60], we use the pre-defined training images of a-PASCAL as the training set and test on pre-defined test images of a-PASCAL and a-Yahoo respectively. We use the feature provided in [60]: 9,751-dimensional features of color, texture, visual words, and edges to train individual classifiers. In addition, weak attributes include:

- Scores from classemes semantic classifiers [203]: 2,659 classifiers trained on images returned by search engines of corresponding query words/phrases;
- Discriminative attributes [60], which are trained using linear SVM by randomly selecting 1-3 categories as positive, and 1-3 categories as negative;
- Random image distances: the distance of each image to some randomly selected images based on the 9,751-dimensional feature vector;
- Latent variables, as detailed in Section 12.4.1.

This finally results in 5,000 weak attributes for each image.

Figure 12.5 shows performance evaluation results using the a-PASCAL benchmark, in comparison with the state-of-the-art approaches in [190, 79, 167]. Our approach outperforms all other methods for all types of queries, especially with large margins for double and triple query scenarios (Figure 12.5 Left). An example of the retrieval result is shown in Figure 12.8. We also evaluate the effect of the sparsity level ϑ, as shown in Figure 12.5 (Right). Our approach reaches the best performance with sparsity $\vartheta = 400$ (only 8% of all the weak attributes). Beyond this point, the performance begins to drop, possibly due to overfitting. This validates the assumption we made earlier that for each query, only a partial set of

[6] `http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2008/`

weak attributes are related. In terms of speed, our implementation requires 10 hours for training, with sparsity $\vartheta = 400$, on a 12-core 2.8GHz workstation. The prediction can be done in real time.

Figure 12.6 shows the performance of our methods on the a-Yahoo benchmark in comparison with individual classifiers and MARR. Image categories of a-Yahoo and a-PASCAL are different, resulting in different data statistics. Therefore, training on a-PASCAL and testing on a-Yahoo can be understood as a "cross-dataset" task, which is ideal for evaluating the power of the proposed semi-supervised graphical model. From Figure 12.6 (Left), the performance of MARR is worse than that of individual classifiers, most likely due to cross-dataset issues. In turn, our method outperforms individual classifiers for all types of queries.

To validate the merit of integrating the supervised graph and unsupervised graph into a semi-supervised model (Section 12.3.3), we have further evaluated the performance of the same model but without the unsupervised layer. As expected, the performance drops compared with the semi-supervised model. This is an evidence validating the contribution of the semi-supervised graphical model in mapping query attributes to a large weak attribute pool.

From Figure 12.6 (Right), the optimal sparsity of a-Yahoo ($\vartheta = 200$) is lower than that of a-PASCAL ($\vartheta = 400$), meaning that for cross-dataset scenario, fewer relationships/dependencies from query attribute to weak attributes are generalizable. Nevertheless, our semi-supervised approach can successfully uncover such dependency structures.

12.4.3 a-TRECVID

To further test the effectiveness of our model, we have compiled a large attribute dataset, named a-TRECVID[7]. It contains 126 uniquely labeled attributes, and 6,000 weak attributes, for 0.26 million images. This dataset is compiled from the TRECVID 2011 Semantic Indexing (SIN) track common annotation set[8] by discarding attributes with too few positive

[7]The images, labeled attributes, and computed weak attributes are described in `http://www.ee.columbia.edu/dvmm/a-TRECVID`

[8]`http://www-nlpir.nist.gov/projects/tv2011/`

Boy	Airplane	Airplane_Flying	Trees	Classroom	Government-Leader	Highway	Politicians	Dark-skinned_People
Car	Bicycles	Daytime_Outdoor	Child	Reporters	Animation_Cartoon	Kitchen	Black_Frame	Domesticated_Animal
Gun	Building	Ground_Vehicles	Lakes	Teenagers	Apartment_Complex	Meeting	Blank_Frame	Eukaryotic_Organism
Face	Cheering	People_Marching	Animal	Carnivore	Female_Human_Face	Outdoor	Wild_Animal	Female_News_Subject
Girl	Mountain	Walking_Running	Beards	Herbivore	Head_And_Shoulder	Running	Anchorperson	Construction_Vehicles
Hand	Suburban	Civilian_Person	Driver	Quadruped	Human_Young_Adult	Singing	Asian_People	Instrumental_Musician
Road	Swimming	Female_Reporter	Indoor	Old_People	Male_News_Subject	Streets	Sitting_Down	Waterscape_Waterfront
City	US_Flags	Hispanic_Person	Person	Scene_Text	Military_Aircraft	Walking	Urban_Scenes	Residential_Buildings
News	Clearing	Male_Human_Face	Forest	Body_Parts	Adult_Female_Human	Glasses	Female_Person	Celebrity_Entertainment
Room	Speaking	Office_Building	Hockey	Caucasians	Man_Wearing_A_Suit	Skating	Overlaid_Text	Male-Face-Closeup
Actor	Standing	House_Of_Worship	Mammal	Junk_Frame	Military_Personnel	Talking	Single_Person	Demonstration
Adult	Bicycling	Press_Conference	Athlete	Urban_Park	Religious_Building	Traffic	Amateur_Video	Studio_Anchorperson
Beach	Boat_Ship	Roadway_Junction	Dancing	Vertebrate	Single_Person_Male	Valleys	Male_Reporter	Female-Face-Closeup
Birds	Cityscape	Adult_Male_Human	Flowers	Male_Person	Speaking_To_Camera	Windows	Man_Made	Text_On_Artificial_Bk

Table 12.2: 126 query attributes of a-TRECVID, selected from a pool of 346 concepts defined in TRECVID 2011 SIN task.

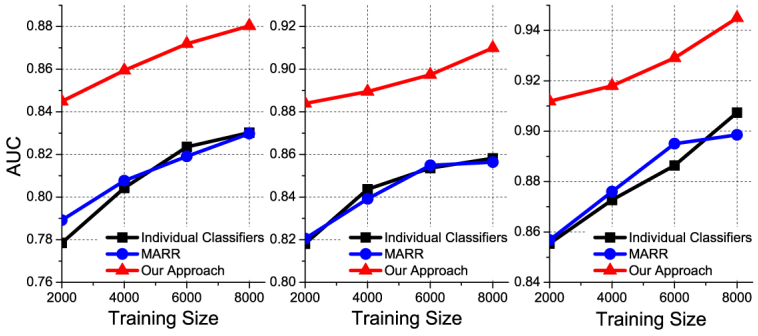


Figure 12.7: Retrieval performance over the a-TRECVID dataset, with the varying training size. From left to right: performance of single, double and triple attribute queries.

images, and images with too few local feature detection regions. The original dataset includes about 0.3 million video frames with 346 fully labeled, unique query attributes [8].

The individual attribute classifiers are trained using bag-of-words SIFT features under the spatial pyramid configuration [121]. Following the setting of Section 12.4.2, weak attributes include individual classifier scores, classemes, discriminative attributes, distance to randomly selected images, and latent variables. Different from a-PASCAL dataset, we treat images from the same video as belonging to the same category. Therefore, the number of categories of a-TRECVID is much larger than that of a-PASCAL, and we have selected 1,000 more discriminative attributes for this dataset. This leads to 6,000 weak attributes per image.

Figure 12.7 shows the performance of our approach comparing to individual classifiers

Figure 12.8: Top-10 results of MARR and our approach based on two query examples of a-PASCAL and a-TRECVID. Images with red frames are false positives. Note that in the third image of "Ours" for a-PASCAL, there is a bird in the background. a-TRECVID is compiled from the dataset used in TRECVID 2011 Semantic Indexing (SIN) track.

and MARR. MARR is only marginally better than individual classifiers, for the reason that the limited feature space is not scalable for the large-scale setting. Our method significantly outperforms both individual classifiers and MARR by 5% – 8.5%. This experiment validates our assumption that the proposed approach can handle the situation when the training size is extremely small. In particular, when using 2,000 images (0.8% of the whole dataset) for training, our method already outperforms both individual classifiers and MARR approaches with 8,000 images for training. An example of retrieval result with 6,000 training images is shown in Figure 12.8.

12.5 Conclusion and Future Works

We introduced *weak attributes* that unify different kinds of mid-level image representations which can be easily acquired with no or little human labor. Based on the large and expressive weak attribute space, robust retrieval model can be developed. Under the framework of structural learning, we extended attribute dependency model originally defined over a small close set of query attributes to a more general and powerful one that maps query to the entire pool of weak attributes. To efficiently learn the dependency model without overfitting, a novel semi-supervised graphical model was proposed to control the model sparsity. It ed enables our approach to be effective for cross-dataset and large-scale

scenarios. We have carried out extensive evaluations on several benchmarks, demonstrating the superiority of the proposed method.

Although the notion of weak attributes was proposed under the framework of multi-attribute based image retrieval, the idea of leveraging attribute classifiers trained from different yet related tasks can also be beneficial to attribute-based recognition. We believe it is also very useful to study methods which can be used to characterize the semantic reliability of the weak attributes.

Chapter 13

Designing Category-Level Attributes for Visual Recognition

13.1 Introduction

To resolve the scalability issue of forming a high-dimensional expressive attribute space, we have proposed the weak attributes in Chapter 12 leveraging attribute models learned from different yet related tasks, under the application of multi-attribute based image retrieval. In this chapter, we propose another approach under the application of attribute-based recognition. The approach is termed as "category-level attributes", a scalable method of automatically designing and learning of attribute models for discriminative visual recognition. Our approach is motivated by [119, 157], in which the attributes are defined by concise semantics, and then manually related to the categories as a *category-attribute matrix* (Figure 13.2). The elements of the matrix characterize each category (row) by the pre-defined attributes (columns). For example, polar bear is non-white, black, non-blue. This matrix is critical for the subsequent process of category-level knowledge transfer.

Similar to characterizing categories as a list of attributes, attributes can also be expressed as how they relate to the known categories. For example, we can say the second attribute of Figure 13.2 characterizes the property that has a high association with polar bear, and a low association with walrus, lion, *etc.* Based on the above intuition, given the images with category labels (a multi-category dataset), we propose to automatically

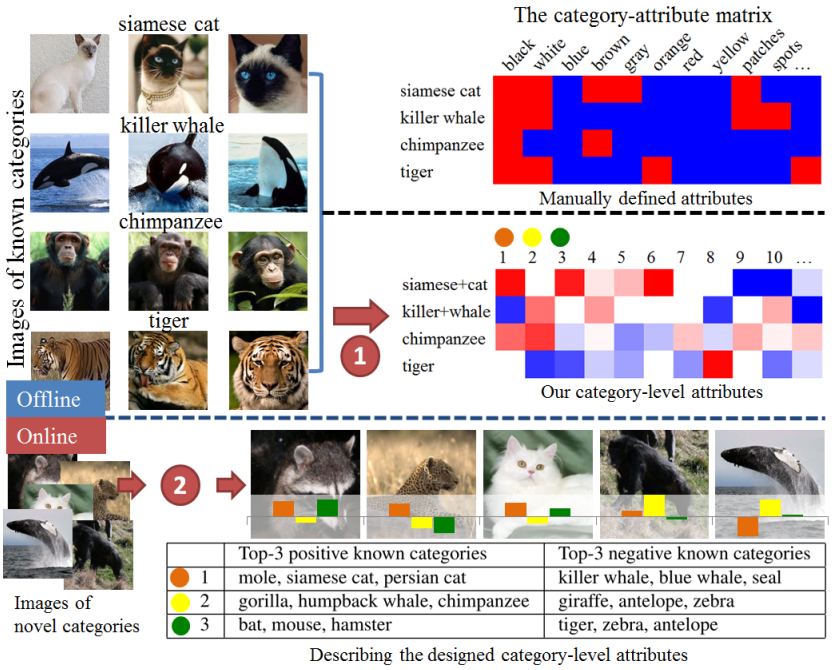


Figure 13.1: Overview of the proposed approach. ①: Designing the category-attribute matrix. ②: Computing the attributes for images of novel categories.

design a category-attribute matrix to *define* the attributes. Such attributes are termed as *category-level attributes*. The designed attributes will not have concise names as the manually specified attributes, but they can be loosely interpreted as relative associations of the known categories. Because multi-category datasets are widely available in the computer vision community, no additional human efforts are needed in the above process.

Figure 13.1 provides an overview of the proposed approach. In the offline phase, given a set of images with labels of pre-defined categories (a multiclass dataset), our approach automatically learns a category-attribute matrix, to *define* the category-level attributes. Then a set of attribute classifiers are learned based on the defined attributes (not shown in the figure). Unlike the previous work [119], in which both the attributes and the category-attribute matrix are pre-defined (as in the "manually defined attributes"), the proposed process is fully automatic. In the online phase, given an image from the novel categories, we can compute the designed category-level attributes. The computed values of three attributes (colored as orange, yellow and green) are shown in Figure 13.1. For example, the

first image (of a raccoon) has positive responses of the orange and green attributes, and negative response of the yellow attribute. Because the category-level attributes are defined based on a category-attribute matrix, they can be interpreted as the relative associations with the pre-defined categories. For example, the orange attribute has positive associations with mole and siamese cat, and negative associations with killer whale and blue whale.

The category-level attributes are more intuitive than mid-level representations defined on low-level features. In fact, our attributes can be seen as soft groupings of categories, with an analogy to the idea of building taxonomy or concept hierarchy in the library science. We will further discuss the semantic aspects of the proposed method in Section 13.5.

Our work in this chapter makes the following unique contributions:

- We propose a principled framework of using category-level attributes for visual recognition (Section 13.3.1).
- We theoretically demonstrate that discriminative category-level attributes should have the properties of category-separability and learnability (Section 13.3.2).
- Based on this analysis, an efficient algorithm is proposed for the scalable design of attributes (Section 13.4).
- We conduct comprehensive experiments (Section 13.6) to demonstrate the effectiveness of our approach in recognizing known and novel categories. Our method achieves the state-of-the-art result on the zero-shot learning task.

13.2 Related Works

Data-Driven Attributes. Traditionally, the semantic attributes are designed by manually picking a set of words that are descriptive for the images under consideration [60, 119, 136]. Non-semantic "data-driven attributes" have been explored to complement semantic attributes with various forms. Kumar et al. [114] combined semantic attributes with "simile classifiers" for face verification. Yang and Shah [226] proposed data-driven "concepts" for event detection. Liu et al. [133] extended a set of manually specified attributes with data-driven attributes for improved action recognition. Sharmanska et al. [188] extended a semantic attribute representation with extra non-interpretable dimensions

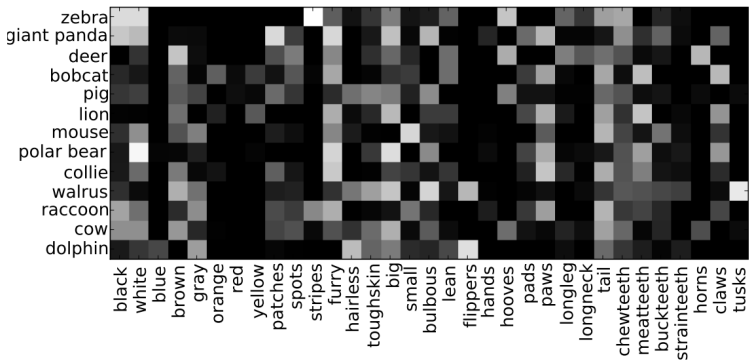


Figure 13.2: Manually defined category-attribute matrix copied from [119]: the rows are the categories, and the columns are the attributes. This matrix is obtained from human judgments on the "relative strength of association" between attributes and animal categories.

for enhanced discrimination. Bergamo et al. [16], Gordoa et al. [75], Rastegari et al. [174] used the large-margin framework to model attributes for objective recognition. Wang and Mori [217], Farhadi et al. [61] used attribute-like latent models to improve object recognition. The highly efficient algorithm and the unique capability of zero-shot learning, differentiate the proposed methodology from the above approaches. The category-level attribute definition can be seen as a generalization of the discriminative attributes used in [60]. Instead of randomly generating the "category split" as in [60], we propose a principled way to *design* the category-level attributes.

Error Correcting Output Code (ECOC). The proposed framework generalizes the Error Correcting Output Code for multi-class classification [4, 42]. Our method becomes ECOC if the elements of the category-attribute matrix are binary. Note that both one-vs-all and one-vs-one multi-class classifications are special cases of ECOC. Different form the conventional ECOC methods, the values of the category-attribute are real-valued, which can capture more subtle difference of the visual categories. More importantly, the category-attribute matrix is learned based on the visual recognition task. In addition, we show how to build the category-attribute matrix to cover categories without any training data in the zero-shot setting.

l	The number of category-level attributes
M	The number of known categories
$\mathbf{A} \in \mathbb{R}^{M \times l}$	The category-attribute matrix
$\mathbf{A}_{\cdot i}$	The i-th column of $\mathbf{A}$
$\mathbf{A}_{i\cdot}$	The i-th row of $\mathbf{A}$
$\mathcal{X}$	Set of all images
$\mathbf{x} \in \mathcal{X}$	Feature of an image
$\mathbf{x}_i \in \mathcal{X}$	Feature of the i-th training sample
$\mathcal{Y} = \{1, \cdots, M\}$	Set of all category labels
$y \in \mathcal{Y}$	A category label
$y_i \in \mathcal{Y}$	Label of the i-th training sample
$\mathbf{f}(\mathbf{x}) \in \mathbb{R}^l$	The l category-level attribute scores of $\mathbf{x}$
ϵ	Average encoding error defined in (13.2)
ρ	Minimum row separation defined in (13.3)
r	Redundancy defined in (13.4)
$\mathbf{S} \in \mathbb{R}^{M \times M}$	Visual similarity matrix
$\mathbf{D} \in \mathbb{R}^{M \times M}$	Visual distance matrix
$\mathbf{L} \in \mathbb{R}^{M \times M}$	Laplacian of $\mathbf{S}$
$\mathbf{w}_i$	Linear model for the i-th attribute

Table 13.1: Key Notations of Chapter 13.

13.3 A Learning Framework of Recognition with Category-Level Attributes

13.3.1 The Framework

The notation of this chapter is shown in Table 13.3.1. We propose a framework of using attributes as mid-level cues for multi-class classification. And the error of such classification scheme is used to measure the "discriminativeness" or "usefulness" of attributes. Note that the framework is defined on recognizing known categories, but the designed attributes are

expected to useful also to the novel and related categories. This is further discussion in Section 13.5 and Section 13.6. Suppose there are M categories, and l attributes. The category-attribute matrix (definition of attributes) is denoted as $\mathbf{A} \in \mathbb{R}^{M \times l}$, in which the columns $\{\mathbf{A}_{\cdot i}\}_{i=1}^{l}$ define l category-level attributes, and the rows $\{\mathbf{A}_{i\cdot}\}_{i=1}^{M}$ correspond to M known categories.

Definition 13.1. *For an input image $\mathbf{x} \in \mathcal{X}$ (as low-level features), we define the following two steps to utilize attributes as mid-level cues to predict its category label $y \in \mathcal{Y}$.*
***Attribute Encoding**: Compute l attributes by attribute classifiers $\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), ..., f_l(\mathbf{x})]^T$ in which $f_i(\mathbf{x}) \in \mathbb{R}$ models the strength of the i-th attribute for $\mathbf{x}$.*
***Category Decoding**: Choose the closest category (row of $\mathbf{A}$) in the attribute space (column space of $\mathbf{A}$):*

$$\underset{i}{\operatorname{argmin}} \parallel \mathbf{A}_{i\cdot} - \mathbf{f}(\mathbf{x})^T \parallel_2 . \tag{13.1}$$

Because $\mathbf{A}$ is real valued, a unique solution for (13.1) can be reached. Figure 13.3 illustrates using two attributes to discriminate cats and dogs.

Definition 13.2. *Designing discriminative category-level attributes is to find a category-attribute matrix $\mathbf{A}$, as well as the attribute classifiers $\mathbf{f}(\cdot)$ to minimize the multi-class classification error.*

Motivation. The above framework is motivated by two previous studies: learning attributes based on category-attribute matrix [119], and Error Correcting Output Code (ECOC) [4, 42]. Multiple previous researches can be unified into the framework by firstly setting $\mathbf{A}$ as a pre-defined matrix, and then modeling $\mathbf{f}(\cdot)$ accordingly. For example, in the previous studies, $\mathbf{A}$ was set as a manually defined matrix [119], a random matrix (discriminative attributes [60]), or a M-dimensional square matrix with diagonal elements as 1 and others as -1. The last case is exactly the one-vs-all approach, in which an attribute is equivalent to a single category. When applied for recognizing novel categories, such attributes are termed as category-level semantic features, or, classemes [203, 222].

Such attributes are discriminative for known categories (Section 13.6.1). However as they capture no properties shared across categories, the computed attributes, known as

category-level semantic features or classemes [203, 222], may not be effective for recognizing novel categories (Section 13.6.2, Section 13.6.3).

Different from the above ad-hoc solutions, we propose to *design* $\mathbf{A}$ and learn $\mathbf{f}(\cdot)$, for discriminative visual recognition. Unlike the manual attributes and classemes, the designed attributes are without concise semantics. However, the category-level attributes are more intuitive than mid-level representations defined on low-level features reviewed in Section 13.2. In fact, our attributes can be seen as *soft* groupings of categories, with analogy to the idea of building taxonomy or concept hierarchy in the library science. We provide a discussion on the semantic aspects of the proposed method in Section 13.5. In addition, by defining attributes based on a set of known categories, we are able to develop a highly efficient algorithm to design the attributes (Section 13.4). It also enables a unique and efficient way for doing zero-shot learning (Section 13.6.3).

Also note that in our model, the matrix $\mathbf{A}$ is real-valued, which is more suitable for vision applications (black bear is totally black; zebra is only partially black), while many studies on ECOC are based on binary values.

13.3.2 Theoretical Analysis

In this section, we show the properties of good attributes in a more explicit form. Specifically, we bound the empirical multi-class classification error in terms of attribute encoding error and a property of the category-attribute matrix, as illustrated in Figure 13.3.

Formally, given training examples $\{\mathbf{x}_i, y_i\}_{i=1}^N$, in which $\mathbf{x}_i \in \mathcal{X}$ is the feature, and $y_i \in \mathcal{Y}$ is the category label associated with $\mathbf{x}_i$:

Definition 13.3. *Define ϵ as the average encoding error of the attribute classifiers $\mathbf{f}(\cdot)$, with respect to the category-attribute matrix $\mathbf{A}$.*

$$\epsilon = \frac{1}{N} \sum_{i=1}^{N} \| \mathbf{A}_{y_i \cdot} - \mathbf{f}(\mathbf{x}_i) \|_2. \tag{13.2}$$

Definition 13.4. *Define ρ as the minimum row separation of the category-attribute matrix* $\mathbf{A}$

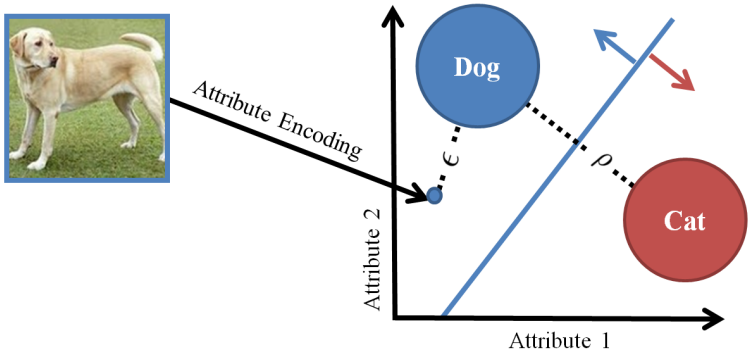


Figure 13.3: Discriminating dogs and cats, with two attributes. Each category (a row of $\mathbf{A}$) is a template vector in the attribute space (a column space of $\mathbf{A}$). ρ is the row separation of the category-attribute matrix. A new image of dog can be represented as an attribute vector through attribute encoding, and ϵ is the encoding error. In order for the image not to be mistakenly categorized as cat, small ϵ and large ρ are desired.

$$\rho = \min_{i \neq j} \| \mathbf{A}_{i\cdot} - \mathbf{A}_{j\cdot} \|_2 \ . \tag{13.3}$$

Proposition 13.1. *The empirical error of multi-class classification is upper bounded by* $2\epsilon/\rho$.

The proof is shown in the Appendix. The message delivered by the bound is very intuitive. It tells us discriminative attributes should have the following properties, illustrated in Figure 13.3:

- **Category-separability.** We want ρ to be large, *i.e.* the categories should be separated in the attribute space.
- **Learnability.** We want ϵ to be small, meaning that the attributes should be learnable. This also implies that attributes should be shared across "similar" categories.

In addition, we also want the attributes to be non-redundant. Otherwise, we may get a large number of identical attributes. In this chapter, the redundancy is measured as

$$r = \frac{1}{l} \| \mathbf{A}^T\mathbf{A} - \mathbf{I} \|_F^2, \tag{13.4}$$

in which $\| \cdot \|_F$ is the Frobenius norm.

13.4 The Attribute Design Algorithm

Based on the above analysis, we propose an efficient and scalable algorithm to design the category-attribute matrix $\mathbf{A}$, and to learn the attribute classifiers $\mathbf{f}(\cdot)$. The algorithm is fully automatic given images with category labels. The proposed solution is a two-step process, where the category-attribute matrix is firstly optimization, and then used in the attribute learning. Note that one may be tempted to optimize the category-attribute matrix, and the attribute classifiers jointly. However, it will lead to a very difficult optimization problem, where some iterative methods (including multiple times of training the attribute classifiers) need to be used. The proposed approach is much more efficient than the joint scheme.

13.4.1 Designing the Category-Attribute Matrix

To optimize the category-attribute matrix $\mathbf{A}$ (definition of attributes), we first consider the objective function in the following form, without the non-redundancy constraint:

$$\underset{\mathbf{A}}{\operatorname{argmax}} J(\mathbf{A}) = J_1(\mathbf{A}) + \lambda J_2(\mathbf{A}), \tag{13.5}$$

in which $J_1(\mathbf{A})$ induces separability (larger ρ), and $J_2(\mathbf{A})$ induces learnability (smaller ϵ). To benefit the algorithm, we set $J_1(\mathbf{A})$ as sum of all distances between every two rows of $\mathbf{A}$, encouraging every two categories to be separable in the attribute space.

$$J_1(\mathbf{A}) = \sum_{i,j} \parallel \mathbf{A}_{i\cdot} - \mathbf{A}_{j\cdot} \parallel_2^2 . \tag{13.6}$$

We set $J_2(\mathbf{A})$ as a proximity preserving regularizer

$$J_2(\mathbf{A}) = -\sum_{i,j} S_{ij} \parallel \mathbf{A}_{i\cdot} - \mathbf{A}_{j\cdot} \parallel_2^2, \tag{13.7}$$

in which S_{ij} measures the category-level visual proximity between the category i and category j. The intuition is that if two categories are visually similar, we expect them to share more attributes. Otherwise, the attribute classifiers will be hard to learn. The construction of the visual proximity matrix $\mathbf{S} \in \mathbb{R}^{M \times M}$ will be presented in Section 13.4.3.

It is easy to show that

$$J(\mathbf{A}) = \operatorname{Tr}(\mathbf{A}^T \mathbf{Q} \mathbf{A}), \quad \mathbf{Q} = \mathbf{P} - \lambda \mathbf{L}, \tag{13.8}$$

Algorithm 7 Designing the category-attribute matrix

Initialize $\mathbf{R} = \mathbf{Q}$, and $\mathbf{A}$ as an empty matrix, solve (13.9) by sequentially learning l additional columns.

for $i = 1 : l$ **do**

Solve (13.10) to get $\mathbf{a}$

Add the new column $\mathbf{A} \leftarrow [\mathbf{A}, \mathbf{a}]$

Update[1] $\mathbf{R} \leftarrow \mathbf{R} - \eta \mathbf{a}\mathbf{a}^T$

end for

in which $\mathbf{P}$ is with diagonal elements being $M - 1$ and all the other elements -1, and $\mathbf{L}$ is the Laplacian of $\mathbf{S}$ [211].

Considering the non-redundant objective, if we force the designed attributes to be strictly orthogonal to each other, *i.e.* $\mathbf{A}^T\mathbf{A} = \mathbf{I}$, the problem can be solved efficiently by a single step, *i.e.* $\mathbf{A}$ combines the top eigenvectors of $\mathbf{Q}$. However, just like PCA, the orthogonal constraint will result in low-quality attributes, because of the fast decay of eigenvalues. So we relax the strict orthogonal constraint, and solve the following problem:

$$\underset{\mathbf{A}}{\text{argmax}} \quad \text{Tr}(\mathbf{A}^T\mathbf{Q}\mathbf{A}) - \beta \parallel \mathbf{A}^T\mathbf{A} - \mathbf{I} \parallel_F^2 . \tag{13.9}$$

Without loss of generality, we require the columns of $\mathbf{A}$ (attributes) to be ℓ_2 normalized. We propose to incrementally learn the columns of $\mathbf{A}$. Specifically, given an initialized $\mathbf{A}$, optimizing an additional column $\mathbf{a}$ is to solve the following optimization.

$$\underset{\mathbf{a}}{\text{argmax}} \quad \mathbf{a}^T\mathbf{R}\mathbf{a} \qquad \text{s.t.} \quad \mathbf{a}^T\mathbf{a} = 1, \tag{13.10}$$

in which $\mathbf{R} = \mathbf{Q} - \eta\mathbf{A}\mathbf{A}^T$, $\eta = 2\beta$. This is a Rayleigh quotient problem, with the optimal $\mathbf{a}$ as the eigenvector of $\mathbf{R}$ with the largest eigenvalue[2]. The overall algorithm is described in Algorithm 7. The algorithm greedily finds additional non-redundant attributes, with desired properties.

[1]Because $\mathbf{A}\mathbf{A}^T = \sum_{i=1}^{l} \mathbf{A}_{\cdot i}\mathbf{A}_{\cdot i}^T$, in each iteration $\mathbf{R}$ can be updated as $\mathbf{R} \leftarrow \mathbf{R} - \eta\mathbf{a}\mathbf{a}^T = \mathbf{Q} - \eta\mathbf{A}\mathbf{A}^T$ for efficiency.

[2]In practice, we also force the optimized $\mathbf{a}$ to be a sparse vector, by manually setting the values close to 0 as 0. This can improve the efficiency in attribute learning, which will be detailed in Section 13.4.2.

13.4.2 Learning the Attribute Classifiers

After getting the real-valued category-attribute matrix $\mathbf{A}$, the next step is to learn the attribute classifiers $\mathbf{f}(\cdot)$. We assume each classifier $\{f_i(\cdot)\}_{i=1}^{l}$ can be learned independently. Specifically, suppose $f_i(\cdot)$ can be represented by a linear model $\mathbf{w}_i$, our solution is to solve a large-margin classification problem with weighted slack variables.

$$\begin{aligned} \underset{\mathbf{w}_i,\xi}{\text{argmin}} \quad & \| \mathbf{w}_i \|_2^2 + C \sum_{j=1}^{N} |A_{y_j,i}| \xi_j \\ \text{s.t.} \quad & \text{sign}(A_{y_j,i}) \mathbf{w}_i^T \mathbf{x}_j \geq 1 - \xi_j, \quad \xi_j \geq 0, \quad j = 1, \cdots, N \end{aligned} \tag{13.11}$$

in which the binarized category-attribute matrix element $\text{sign}(A_{y_j,i})$ defines the presence/non-presence of the i-th attribute for $\mathbf{x}_j$. The idea is to put higher penalties for misclassified instances from categories with a stronger category-attribute association. Generalizing to kernel version is straightforward. Note that if $A_{y_j,i}$ is zero, we do not need to consider the images of category y_j when learning the i-th attribute. Therefore, after (13.10), we also force some values of $\mathbf{a}$ close to 0 to be 0, in order to improve efficiency.

13.4.3 Building the Visual Proximity Matrix

In the proposed algorithm in Section 13.4.1, one important issue is to build the visual proximity matrix $\mathbf{S} \in \mathbb{R}^{M \times M}$ used in (13.7). This matrix is key towards making the attributes learnable, and sharable across categories. Similar to [211], we first build a distance matrix $\mathbf{D} \in \mathbb{R}^{M \times M}$, in which D_{ij} measures the distance between category i and j. $\mathbf{S}$ is modeled as a sparse affinity matrix, with the non-zero elements $S_{ij} = e^{-D_{ij}/\sigma}$.

$\mathbf{D}$ is built dependent on *type* of kernel used for learning the attribute classifiers $\mathbf{f}(\cdot)$ (Section 13.4.2)[3]. When nonlinear kernels are used, SVM margins of $M(M-1)/2$ *one-vs-one* SVMs modeled on low-level features are used as distance measurement for categories; when linear kernels are used (which is usually used for large-scale problems), we simply use the distances of category centers (category mean of the low-level features) as distance

[3]The visual proximity matrix $\mathbf{S}$ is only dependent on the kernel type, not the learned attribute classifiers. Therefore, designing attributes (Section 13.4.1) and learning the attribute classifiers (Section 13.4.2) are two sequential steps, requiring no expensive iterations on the image features.

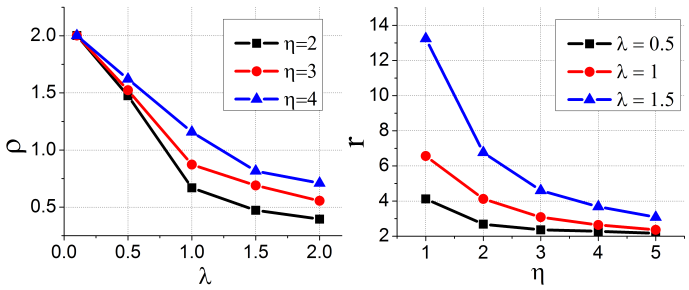


Figure 13.4: The influence of the two parameters. Left: the influence of λ: larger λ means smaller ρ for the category-attribute matrix. Right: the influence of η: larger η means less redundancy r for the designed attributes. The visual proximity matrix $\mathbf{S}$ used for this figure is a 50×50 randomly generated non-negative symmetric matrix.

measurements. Because the category centers can be pre-computed, the latter process is very fast, with computational complexity linear to the number of images, and quadratic to the number of categories.

13.4.4 Parameters of the Algorithm

There are two parameters in the attribute design algorithm, λ and η. Larger λ means smaller ρ for the category-attribute matrix, and larger η means less redundancy r for the designed attributes. Figure 13.4 visualizes the influence of the parameters based on a randomly generated visual proximity matrix $\mathbf{S}$.

13.5 Discussions

Efficiency and Scalability. The attribute design algorithm requires no expensive iterations on the image features. The computational complexity of designing an attribute (a column of $\mathbf{A}$) is as efficient as finding the eigenvector with the largest eigenvalue of matrix $\mathbf{R}$ in (13.10) (quadratic to the number of categories). For example, on a 6-core Intel 2.5 GHz workstation, it just takes about 1 hour to design 2,000 attributes based on 950 categories on the large-scale ILSVRC2010 dataset (Section 13.6.2).

Known Categories *vs.* Novel Categories. Though the above algorithm is for

designing attributes to discriminate known categories, the application of the designed attributes is for recognizing *novel* categories, the categories that are not used in the attribute designing process. In specific, we will show by experiment:

- The designed attributes are discriminative for novel, yet related categories (Section 13.6.2 AwA dataset).
- The designed attributes are discriminative for general novel categories, provided we can design a large number of attributes based on a diverse set of known categories (Section 13.6.2 ILSVRC2010 dataset).
- The attributes are effective for the task of zero-shot learning (Section 13.6.3).

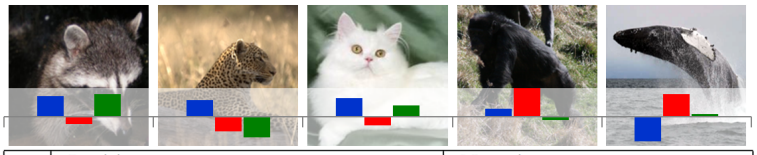
Interpretations of the Category-Level Attributes. One unique advantage of the designed attributes is that they can provide interpretable cues for visualizing the machine reasoning process. In other words, the designed attributes can be used to answer not only "what", but also "why" one image is recognized as a certain category. First, the attributes are designed on category level, the descriptions are readily available through weighted categories names (*e.g.*, the attribute that has high association with polar bear, and low association with walrus, lion). Second, the regularization term $J_2(\mathbf{A})$ in the attribute design formulation can, in fact, lead to human interpretable attributes, by inducing "similar" categories not to be far away in attribute space. Some examples of using the computed attributes to describe the images of novel categories are shown in Figure 13.5.

13.6 Experiments

Datasets. We evaluate the performance of the designed attributes on Animal with Attributes (AwA) [119], and ILSVRC2010 datasets[4]. AwA contains 30,475 images of 50 animal categories. Associated with the images, there is a manually designed category-attribute matrix of 85 attributes shown in Figure 13.2. ILSVRC2010 contains 1.2M images from 1,000 diverse categories. The experiments are performed 10 times, and we report the mean performance.

Baselines. We first demonstrate that our designed category-level attributes are more

[4]http://www.image-net.org/challenges/LSVRC/2010/download-public



	Positive	Negative
	mole, siamese cat, persian cat	killer whale, blue whale, seal
	gorilla, humpback whale, chimpanzee	giraffe, antelope, zebra
	bat, mouse, hamster	tiger, zebra, antelope

Figure 13.5: Using category-level attributes to describe images of novel categories. In the table below, three attributes are described in terms of the corresponding top positive/negative known categories in the category-attribute matrix. Some designed attributes can be further interpreted by concise names: the first two can be described as small land animals *vs.* ocean animals, black *vs.* non/partial-black. Some may not be interpreted concisely: the third one looks like rodent *vs.* tiger and cloven hoof animals. The figure above shows the computed attribute values for images of novel categories.

discriminative than other *category-level representations*. In the task of discriminating known categories (Section 13.6.1), we use the framework proposed in Section 13.3, and compare the performance of the designed attributes with the manual attributes [119] (85 manually defined attributes with a manually specified category-attribute matrix), random category-level attributes [60] (attributes defined as a randomly generated category-attribute matrix), and one-vs-all classifiers (equivalent to attributes defined as a matrix, with diagonal elements as 1 and others as -1). In the task of novel category recognition in Section 13.6.2, we use the extracted attributes as features to perform classification on the images of novel categories. Our approach is compared with the manual attributes, random attributes, classemes [203] (one-vs-all classifiers learned on the known categories), and low-level features (one-vs-all classification scheme based on low-level features of the novel categories). We also test the retrieval and classification performance of our approaches based on the large-scale ILSVRC2010 data. To demonstrate the capability of zero-shot learning of the designed attributes, we compare our approach with the best published results to date in Section 13.6.3.

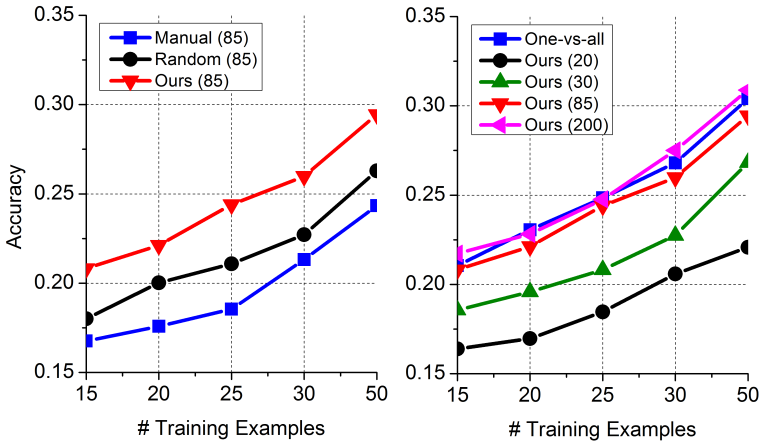


Figure 13.6: Multi-class classification accuracy on known categories. The numbers in parenthesis are the numbers of attributes. The standard deviation is around 1%.

13.6.1 Discriminating Known Categories

In this section, we verify the multi-class classification performance and other properties described in Section 13.3.2 on 40 *known* categories of AwA dataset. The attributes are designed based on 40 training categories defined in [119]. The same low-level features (10,940D), and kernel (χ^2 with bandwidth as 0.2 times median distance) are used. For random attributes, each element of the random category-attribute matrix is generated uniformly from $[-1, 1]$[5].

We select different amount of images per category for training, 25 images per category for testing, and 10 images per category for validation. The parameters are tuned based on the validation set. The margins of $40 \times 39/2 = 780$ one-vs-one classifiers on the training data are used as distance measurements $\mathbf{D}$ of animal categories (the C parameter for one-vs-one SVMs is simply fixed as 10). The visual proximity matrix $\mathbf{S}$ is built as the mutual 10-NN adjacent matrix with bandwidth parameter σ set as 0.5 times the average distance [211]. We first fix the weighed SVM penalty $C = 2$, and tune $\lambda \in \{2, 3, 4, 5\}$, $\eta \in \{6, 8, 10, 12, 14\}$. Then we tune $C \in \{0.02, 0.2, 2, 20\}$.

[5]Other alternatives including binary random matrix, sparse binary random matrix, yield similar performance.

Measurement	Designed	Manual	Random
Encoding error ϵ	**0.03**	0.07	0.04
Minimum row separation ρ	**1.37**	0.57	1.15
Average row separation	**1.42**	1.16	1.41
Redundancy r	**0.55**	2.93	0.73

Table 13.2: Properties of different attributes. The number of attributes is fixed as 85. Encoding error ϵ is defined in (13.2). Minimum row separation ρ is defined in (13.3). Averaged row separation is value of the objective function in (13.6). Redundancy r is defined in (13.4). The category-attribute matrices are column-wise $l2$ normalized in order to be comparable. The measurements are computed on the test set.

Figure 13.6 demonstrates the performance of multi-class classification. Table 13.2 further verifies the properties of the designed attributes.

- The designed attributes perform significantly better than the manual attributes and random category-level attributes (Figure 13.6 left).
- The designed attributes is competitive to, if not better than the low-level features paired with one-vs-all χ^2 classifiers (Figure 13.6 right). The designed attributes significantly outperform the one-vs-all classifiers (known as classemes) for the task of recognizing novel categories (Section 13.6.2), due to the fact that classemes are not shared across categories.
- The designed attributes have smaller encoding error, larger row separation and smaller redundancy. This justifies the theoretical analysis in Section 13.3.2.

One interesting observation is that even the random category-attribute matrix has better properties compared with the manually defined category-attribute matrix (Table 13.2). The random attributes therefore outperform the manual attributes (Figure 13.6 left).

13.6.2 Discriminating Novel Categories

We show that the designed attributes are also discriminative for *novel* categories. Specifically, we use the attributes, and other kinds of category-level representations as *features*, to

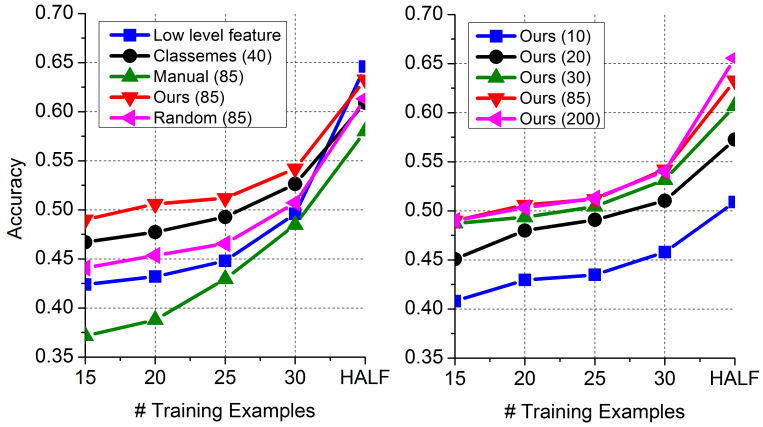


Figure 13.7: Multi-class classification accuracy on novel categories. The 64.6% accuracy with one-vs-all classifier using 50/50 (HALF) split is similar to the performance (65.9%) reported in [119]. The numbers in bracket are the number of attributes. The standard deviation is around 1%.

perform the multi-class classification (AwA, ILSVRC2010) and the category-level retrieval (ILSVRC2010) tasks.

Animals with Attributes. We use the 40 animal categories in Section 13.6.1 to design the attributes. Efficient linear SVM classifiers are trained based on different kinds of attribute features to perform classification on the images of 10 novel categories. The optimally tuned parameters in Section 13.6.1 are used for the task.

Figure 13.7 shows the performance. The designed attributes perform significantly better than other types of representations, especially with few training examples. This means that attributes are discriminative representation for the novel categories, by leveraging knowledge learned from known categories. As the number of training images increases, the performance of low-level features is improved, due to sufficient supervision. Note that the manual attributes and classemes are with fixed dimensions, whereas the dimension of the designed category-level attributes is scalable.

ILSVRC2010. In the previous experiments on AwA, we showed that the designed attributes are discriminative for the novel, yet related categories. We now demonstrate that the designed attributes can be discriminative for general novel categories, provided that we

Method	Low-level feature	Classeme (950)	Ours (500)	Ours (950)	Ours (2,000)
Precision@50	33.40	39.24	39.85	42.16	**43.10**

Table 13.3: Category-level image retrieval result on 50 classes from ILSVRC2010. The numbers in bracket are the numbers of attributes. We closely follow the settings of [72].

	Percentage for training				
Method	1%	5%	10%	50%	100%
Low-level feature	35.55	52.21	57.11	66.21	**69.16**
Classemes (950)	38.54	51.49	56.18	64.31	66.77
Ours (500)	39.01	52.86	56.54	62.38	63.86
Ours (950)	41.60	55.32	59.09	65.15	66.74
Ours (2,000)	**43.39**	**56.51**	**60.36**	**66.91**	68.17

Table 13.4: Image classification accuracy on 50 classes from ILSVRC2010. The training set contains 54,636 images. The number in the bracket is the number of attributes. Standard deviation is around 1%. We closely follow the settings of [72].

can design a large amount of attributes based on a diverse set of known categories. The ILSVRC2010 dataset is used for this experiment. Following the settings in [72], the low-level features are 4,096 dimensional fisher vectors [166]. 950 categories are used as known categories to design attributes. We test the performance of using attribute features for category-level retrieval and classification on the remaining 50 disjoint categories.

The distances of category centers (based on low-level features) are used as distance measurements $\mathbf{D}$ of categories. The visual proximity matrix $\mathbf{S}$ is built as a 30-NN mutual adjacent matrix, with bandwidth parameter σ as 0.5 times the mean distance [211]. The attributes are trained by linear weighted SVM models. All other detailed experiment settings, including data splits, and ways for parameter tuning are identical to [72].

We first test the performance of the designed attributes for category-level image retrieval. 1,000 randomly selected images are used as queries to retrieve the nearest neighbors from the remaining 67,295 images. Table 13.3 shows the performance in terms of precision@50. The

designed attributes outperform low-level features and classemes, even with 500 dimensions. And 2,000-dimensional attributes outperform the baselines by 9.70% and 3.86% respectively.

Next, we use the attribute feature, combined with linear SVM classifiers to classify the images of the 50 novel categories. 80% of the data are used for training (54,636 images), 10% for testing, and 10% for validation. Table 13.4 shows multi-class classification accuracy, using different amount of training images from the training set. Similar to the experiments on AwA dataset, attribute representation outperforms the baselines, especially when training with a small number of examples. It means attributes are effective for leveraging information of the known categories to recognize novel categories. As the number of training images increases, the performance of low-level features goes up, due to sufficient amount of supervision.

A standard eigenvalue solver and a modified Liblinear SVM are used for computing eigenvectors and learning the encoding models, respectively. We report the time for the algorithm on a 4-core 2.5GHz Intel workstation, with all the training data from the 950 categories without sub-sampling. Building the similarity matrix takes less than 1 min; Designing 2,000 attributes takes 42 min; Learning the 2,000 dimensional encoding model takes about 4 hours (this step can be speedup by considering only top positive/negative categories for each attribute); Attribute encoding on low-level features is instant.

13.6.3 Zero-Shot Learning

Building the New Category-Attribute Matrix. Zero-shot learning can be seen as a special case of recognizing novel categories, without training data. In such case, human knowledge [119, 157] is required to build a new category-attribute matrix $\mathbf{A'} \in \mathbb{R}^{M' \times l}$, to relate the M' novel categories to the l designed attributes. After that, we can follow the framework in Section 13.3.1 to recognize the novel categories. However, for each designed attribute in our approach, there is no guarantee that it possesses a coherent human interpretation. For example, while some may say the visual property separating tiger and zebra from cat and dog is "striped", others may say it is the sizes of animals that matter. Therefore, given a new animal, *e.g.* skunk (both striped and small), the humans may come up with different answers.

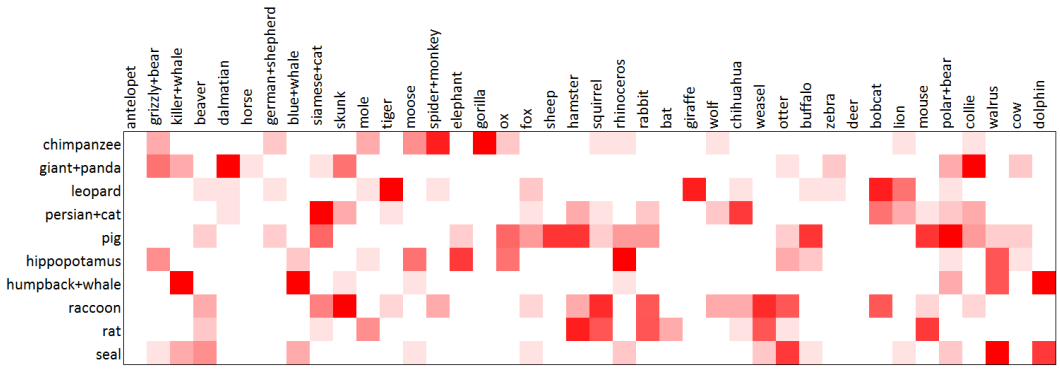


Figure 13.8: The manually built visual similarity matrix. It characterizes the visual similarity of the 10 novel categories and the 40 known categories. This matrix is obtained by averaging the similarity matrices built by 10 different users. Each user is asked to build a visual similarity matrix, by selecting 5 most visually similar known categories for each novel category. The selected elements will be set as 1, and others as 0.

Motivated by the fact that the visual proximity matrix $\mathbf{S}$ in (13.7) is central to the attribute design process, we propose a fairly straightforward solution: similar to [213], given each novel category, and M known categories, we ask the user to find the top-M visually similar categories. The user is free to use any similarity interpretation they wish. We will then have a similarity matrix $\mathbf{S}' \in \{0,1\}^{M' \times M}$, in which $\tilde{S}_{ij}$ is the binary similarity of the i-th novel category and the j-th known category. Figure 13.8 visualizes the averaged similarity matrix based on the results of 10 users.

The novel categories are related to the designed attributes by the simple weighted sum:

$$\mathbf{A}' = \mathbf{S}'\mathbf{A} \tag{13.12}$$

The amount of human interaction is minimal for the above approach, independent on the number of attributes.

Experiment Results. We test the zero-shot learning performance on the AwA dataset, with same settings of [119] (40 animal categories for training and 10 categories for testing). For each novel category, we ask the users to provide up to top-5 similar categories when building the similarity matrix. Empirically, fewer categories cannot fully characterize the visual appearance of the animals, and more categories will lead to more human burdens.

Method	# Attributes	Accuracy
Lampert et al.[119]	85	40.5
Yu and Aloimonos [236]	85	40.0
Rohrbach et al.[176]	-	35.7
Kankuekul et al.[102]	-	32.7
Ours	10	40.52 ± 4.58
Ours	85	42.27 ± 3.02
Ours	200	42.83 ± 2.92
Ours (Fusion)	200	**46.94**
Ours (Adaptive)	200	45.16 ± 2.75
Ours (Fusion + Adaptive)	200	**48.30**

Table 13.5: Zero-shot multi-class classification accuracy with standard deviation on the 10 novel animals categories.

Ten graduate students, who were not aware of the zero-shot learning experiments, were included in the study. When performing the tasks, they were asked to think about visual similarities, rather than similarities otherwise. The time spent for the task ranges from 15 to 25 minutes. Because there is no validation set for zero-shot learning, we empirically set λ, η and SVM penalty C as 3, 15 and 20, throughout the experiments. The performance is not sensitive to the parameters for the range described in Section 13.6.1.

Figure 13.5 shows the experiment results in comparison with various published baselines. Our approach achieves the state-of-the-art performance, even with just 10 attributes. The accuracy and robustness can be improved by using more attributes, and by averaging the multiple binary visual similarity matrices (Fusion). The former helps to fully explore the visual similarities $\mathbf{S}'$, and the later helps to filter out noise from different users. We have achieved accuracy of 46.94%, which significantly outperforms all published results.

Adaptive Attribute Design. In the experiments above, the attributes are designed to be discriminative for the known categorizes. As a refinement for zero-shot learning, we can modify the algorithm to design attributes *adaptively* for discriminating the novel categories.

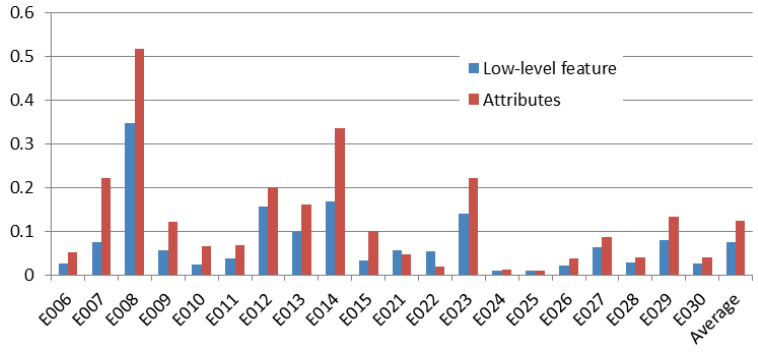


Figure 13.9: Average Precision results base on low-level feature and attributes for the full exemplar task of TRECVID MED 2012. The results are evaluated on the internal threshold split containing 20% of the training data. Linear SVMs are used for event modeling. The same low-level features are used for training attributes.

This can be achieved by changing the first objective $J_1(\mathbf{A})$ (Section 13.4.1) to

$$J_1{}'(\mathbf{A}) = J_1(\mathbf{S}'\mathbf{A}). \tag{13.13}$$

In other words, we want to design a category-attribute matrix $\mathbf{A}$ which is specifically discriminative for the *novel* categories. The modified problem can be solved with minor modifications of the algorithm. The last two rows of Table 13.5 demonstrate the performance of adaptive attribute design. Combined with averaged similarity matrix (Fusion + Adaptive), we have achieved the multi-class classification accuracy of 48.30%, outperforming all published results with a larger margin. The drawback for the adaptive attribute design is that we need to redesign the attributes for different tasks. Because the proposed attribute design algorithm is highly efficient, the drawback can be alleviated.

13.6.4 Designing Attributes for Video Event Modeling

We show one additional application: using attributes for video event classification on the TRECVID 2012 MED task. Traditionally, the semantic features for video event modeling are learned from the taxonomy with the labeled images [28]. The taxonomy is manually defined based on expert knowledge, and a set of images must be labeled by human experts.

Similar to the manually specified attributes, the semantic features suffer from the following problems.

- The human defining and labeling processes are very expensive, especially if we need large-amount of concepts, with enough clean training data.
- Though the taxonomy is semantically plausible, it may not be consistent to the visual feature distributions. Consequently, some dimensions of the semantic feature vector are difficult to be modeled.

Motivated by the above facts, we use the proposed category-level attributes as a data-consistent way of modeling "semantics". Specifically, we design attributes based on 518 leaf nodes of the IMARS taxonomy [28] (as the known categories).

To test the performance of the proposed approach, we have trained and extracted 2,500-dimensional attribute feature for the pre-specified task of TRECVID MED 2012. Following the framework [28], the attributes are used as features which are later plugged into linear SVM for the task of event classification. Figure 13.9 shows the performance of the low-level feature and the proposed attribute feature. Impressively, attributes have achieved relative performance gain over 60%, improving the mAP from 0.075 to 0.123.

13.7 Conclusion and Future Works

We proposed a novel method for designing category-level attributes. Such attributes can be effectively used for tasks of cross-category knowledge transfer.

Not all attributes designed can be semantically interpreted. For the future works, one possible way to enhance the semantics in the attribute designing process is by human interactions, which have have been successfully applied to mobile visual search [228], and complex video event recognition [17]. The solution is to modify the attribute design algorithm with an additional *semantic projection* step: after getting each $\mathbf{a}$ (a column of the category-attribute matrix), make some minimal changes to $\mathbf{a}$ to make it semantically meaningful. Specifically, given an initial pool of pre-defined attributes, together with their manually specified category-attribute matrix, we can define some rules of *what* kinds of category-level attributes are semantically meaningful. For instance, it is intuitive to say

the union (black or white), intersection (black and white), and subset (chimpanzee kinds of black, attributes are often category-dependent) of the manually defined attributes are semantically interpretable. The operations of union, intersection *etc.* can be modeled by operations on the manually specified category-attribute matrix. The designed attributes can then be projected to the nearest semantic candidate. This method can potentially be used to efficiently *expand* the predefined semantic attributes.

Part IV

Conclusions

Chapter 14

Conclusions

14.1 Contributions and Open Issues

In the thesis, we identified challenges facing machine learning methods for large-scale visual data: the large-scale of data, and the limited supervision. We proposed solutions to scale up machine learning for very high-dimensional data, and weak supervision. We highlight our contributions as well as some open issues in this section.

For learning with high-dimensional data, we showed that a type of special structured matrix, the circulant matrix, can be used to improve the efficiency of machine learning for high-dimensional data. We proposed to use the circulant structure in different machine learning models, including binary embedding, neural network, and compact nonlinear maps. Surprisingly, the method reduces the number parameters from $O(d^2)$ to $O(d)$, where d is the feature dimension, without hurting the performance. The success of the method in all the applications means that for high-dimensional data, redundancy exists in the conventional unstructured projection matrices. The circulant matrix provides a way to reduce such redundancy, as well as to reduce the computational cost. To foster a deeper understanding of the matter, as well as to develop more general tools for scalable learning for high-dimensional data, it is important to characterize and find the relationship of the redundancies in the three machine learning models and those in dimensionality reduction.

For learning with weak supervision, we studied a weakly supervised learning setting called learning from label proportions (LLP). It scales learning methods by incorporating

non-conventional types of supervision, which are widely available yet not utilized by other learning methods. The proposed approach has been applied in video event detection and attribute learning with category-attribute proportions. Besides enabling new applications, the feasibility of learning from label proportions also poses concerns in terms of protecting the sensitive information of the individuals. An important future direction is to study the ways of releasing group-level statistics without compromising individual information.

For learning with weak supervision, we also studied attribute modeling, which is widely used in knowledge transfer in computer vision. We provided two ways of scalable design and learning of mid-level visual attributes without the human labeling process. The proposed methods achieved the state-of-the-art results in multi-attribute based retrieval and recognition with few or zero examples. One important open problem is to more accurately characterize the semantic meanings of the designed attributes.

We note that there are many more challenges facing machine learning for large-scale visual data, such as the design and implementation of large-scale parallel computing systems, learning with noisy and incomplete labels, learning with computer-human interactions *etc.* The problems addressed by the thesis only covered a tip of the iceberg. We hope the work can stimulate further research addressing different perspectives of the whole puzzle.

14.2 Applications Beyond Visual Data

Visual data provides a valuable platform to motivate and study machine learning methods. One unique advantage is that such data is relatively easy to collect due to the vast availability of digital cameras, and online photo resources. Most of the methods of the thesis are proposed and evaluated based on applications on visual data, yet they can also be applied to a broader spectrum of applications.

- The circulant projection-based method can be applied to all types of data with high-dimensionality which is common in, for example, finance, biology, and marketing.
- The learning from label proportion approaches are applicable to areas ranging from online advertisement, computational social science to health care. Due to privacy concerns, sensitive information is often released on group level, such as the voting rate of certain

demographic area, the attrition rate of certain department in a company, and the average income of certain ethnic groups. By combining the group statistics with individual-level features (which can be easily obtained from sources such as the social media), the approaches can lead to novel applications, such as forecasting election result based on polling, and predicting income based on census.

- The proposed attribute-based recognition and retrieval methods can be used in modeling general attributes in additional to those in computer vision. For example, they can be used in understanding the unique attributes separating different groups of users, and characterizing new users in applications like content recommendation.

Part V

Bibliography

Bibliography

[1] D. Achlioptas. Database-friendly random projections: Johnson-Lindenstrauss with binary coins. *Journal of Computer and System Sciences*, 2003.

[2] D. Achlioptas and F. McSherry. Fast computation of low rank matrix approximations. In *Proceedings of the ACM Symposium on Theory of Computing*, 2001.

[3] N. Ailon and B. Chazelle. Approximate nearest neighbors and the fast Johnson-Lindenstrauss transform. In *Proceedings of the ACM Symposium on Theory of Computing*, 2006.

[4] E. Allwein, R. Schapire, and Y. Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. *Journal of Machine Learning Research*, 1:113–141, 2001.

[5] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *Advances in Neural Information Processing Systems*, 2002.

[6] M. Anthony and P. L. Bartlett. *Neural network learning: Theoretical foundations*. Cambridge University Press, 2009.

[7] A. Argyriou, R. Hauser, C. A. Micchelli, and M. Pontil. A dc-programming algorithm for kernel selection. In *Proceedings of the International Conference on Machine Learning*, 2006.

[8] S. Ayache and G. Quénot. Video corpus annotation using active learning. In *Proceeding of European Conference on Information Retrieval*, 2008.

[9] B. Babenko, N. Verma, P. Dollár, and S. J. Belongie. Multiple instance learning with manifold bags. In *Proceedings of the International Conference on Machine Learning*, 2011.

[10] F. Bach, G. Lanckriet, and M. Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In *Proceedings of the International Conference on Machine Learning*, 2004.

[11] F. R. Bach and M. I. Jordan. Predictive low-rank decomposition for kernel methods. In *Proceedings of the International Conference on Machine Learning*, 2005.

[12] P. L. Bartlett and S. Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2003.

[13] E. G. Băzăvan, F. Li, and C. Sminchisescu. Fourier kernel learning. In *Proceedings of the European Conference on Computer Vision*, 2012.

[14] K. Bellare, G. Druck, and A. McCallum. Alternating projections for learning with expectation constraints. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2009.

[15] T. Berg, A. Berg, and J. Shih. Automatic attribute discovery and characterization from noisy web data. In *Proceedings of the European Conference on Computer Vision*, 2010.

[16] A. Bergamo, L. Torresani, and A. Fitzgibbon. Picodes: Learning a compact code for novel-category recognition. In *Advances in Neural Information Processing Systems*, 2011.

[17] S. Bhattacharya, F. X. Yu, and S.-F. Chang. Minimally needed evidence for complex event recognition in unconstrained videos. In *Proceedings of the Annual ACM International Conference on Multimedia Retrieval*, 2014.

[18] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven. Photoocr: Reading text in uncontrolled conditions. In *Proceedings of the IEEE International Conference on Computer Vision*, 2013.

[19] A. Blum and S. Chawla. Learning from labeled and unlabeled data using graph mincuts. In *Proceedings of the International Conference on Machine Learning*, 2001.

[20] A. Blum and A. Kalai. A note on learning from multiple-instance examples. *Machine Learning*, 30(1):23–29, 1998.

[21] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 1998.

[22] S. Bochner. *Harmonic analysis and the theory of probability.* Dover Publications, 1955.

[23] D. Borth, R. Ji, T. Chen, T. Breuel, and S.-F. Chang. Large-scale visual sentiment ontology and detectors using adjective noun pairs. In *Proceedings of the ACM Multimedia Conference*, 2013.

[24] L. Bottou. *Large-scale kernel machines.* MIT Press, 2007.

[25] S. Branson, C. Wah, F. Schroff, B. Babenko, P. Welinder, P. Perona, and S. Belongie. Visual recognition with humans in the loop. In *Proceedings of the European Conference on Computer Vision*, 2010.

[26] D. Cai, X. He, and J. Han. Semi-supervised discriminant analysis. In *Proceedings of the IEEE International Conference on Computer Vision*, 2007.

[27] L. Cao, Y. Mu, A. Natsev, S.-F. Chang, G. Hua, and J. R. Smith. Scene aligned pooling for complex video recognition. In *Proceedings of the European Conference on Computer Vision*, 2012.

[28] L. Cao, S.-F. Chang, N. Codella, C. Cotton, D. Ellis, L. Gong, M. Hill, G. Hua, J. Kender, M. Merler, Y. Mu, J. Smith, and F. X. Yu. IBM Research and Columbia University TRECVID-2012 Multimedia Event Detection (MED), Multimedia Event Recounting (MER), and Semantic Indexing (SIN) Systems. In *NIST TRECVID Workshop*, December, 2012.

[29] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[30] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee. Choosing multiple parameters for support vector machines. *Machine Learning*, 46(1-3):131–159, 2002.

[31] O. Chapelle, V. Sindhwani, and S. Keerthi. Optimization techniques for semi-supervised support vector machines. *Journal of Machine Learning Research*, 9:203–233, 2008.

[32] M. S. Charikar. Similarity estimation techniques from rounding algorithms. In *Proceedings of the ACM Symposium on Theory of Computing*, 2002.

[33] B. Chen, L. Chen, R. Ramakrishnan, and D. Musicant. Learning from aggregate views. In *Proceedings of the IEEE International Conference on Data Engineering*, 2006.

[34] T. Chen, F. X. Yu, J. Chen, Y. Cui, Y.-Y. Chen, and S.-F. Chang. Object-based visual sentiment concept analysis and application. In *Proceedings of the ACM Multimedia Conference*, 2014.

[35] Y. Chen and J. Z. Wang. Image categorization by learning and reasoning with regions. *Journal of Machine Learning Research*, 5:913–939, 2004.

[36] Y. Chen, J. Bi, and J. Z. Wang. Miles: Multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):1931–1947, 2006.

[37] Y. Cheng, F. X. Yu, R. Feris, S. Kumar, A. Choudhary, and S.-F. Chang. Fast neural networks with circulant projections. *arXiv preprint arXiv:1502.03436*, 2015.

[38] M. Choi, V. Tan, A. Anandkumar, and A. Willsky. Learning latent tree graphical models. *Journal of Machine Learning Research*, 12:1771–1812, 2011.

[39] A. Collins and P. Kohli. Memory bounded deep convolutional networks. In *arXiv preprint arXiv:1412.1442*, 2014.

[40] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[41] C. Cortes, M. Mohri, and A. Rostamizadeh. Learning non-linear combinations of kernels. In *Advances in Neural Information Processing Systems*, 2009.

[42] K. Crammer and Y. Singer. On the learnability and design of output codes for multiclass problems. *Machine Learning*, 47(2):201–233, 2002.

[43] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[44] A. Dasgupta, R. Kumar, and T. Sarlós. Fast locality-sensitive hashing. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2011.

[45] S. Dasgupta and A. Gupta. An elementary proof of a theorem of johnson and lindenstrauss. *Random Structures & Algorithms*, 22(1):60–65, 2003.

[46] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, Q. Le, M. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang, and A. Ng. Large scale distributed deep networks. In *Advances in Neural Information Processing Systems*, 2012.

[47] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[48] M. Denil, B. Shakibi, L. Dinh, M. Ranzato, and N. Freitas. Predicting Parameters in Deep Learning. In *Advances in Neural Information Processing Systems*, 2013.

[49] E. L. Denton, W. Zaremba, J. Bruna, Y. LeCun, and R. Fergus. Exploiting linear structure within convolutional networks for efficient evaluation. In *Advances in Neural Information Processing Systems*, 2014.

[50] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1):31–71, 1997.

[51] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv preprint arXiv:1310.1531*, 2013.

[52] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52 (4):1289–1306, 2006.

[53] P. Drineas and M. W. Mahoney. On the nyström method for approximating a gram matrix for improved kernel-based learning. *Journal of Machine Learning Research*, 6: 2153–2175, 2005.

[54] D.-Z. Du and F. K. Hwang. *Combinatorial group testing and its applications*. World Scientific, 1993.

[55] K. Duan, D. Parikh, D. Crandall, and K. Grauman. Discovering localized attributes for fine-grained recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[56] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[57] K. Fan. Minimax theorems. *Proceedings of the the National Academy of Sciences of the United States of America*, 39(1):42, 1953.

[58] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[59] C. Farabet, B. Martini, P. Akselrod, S. Talay, Y. LeCun, and E. Culurciello. Hardware accelerated convolutional neural networks for synthetic vision systems. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, 2010.

[60] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[61] A. Farhadi, M. Tabrizi, I. Endres, and D. Forsyth. A latent model of discriminative aspect. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[62] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning object categories using google's image search. In *Proceedings of the IEEE International Conference on Computer Vision*, 2005.

[63] S. Fine and K. Scheinberg. Efficient svm training using low-rank kernel representations. *Journal of Machine Learning Research*, 2:243–264, 2002.

[64] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188, 1936.

[65] P. Frankl and H. Maehara. The johnson-lindenstrauss lemma and the sphericity of some graphs. *Journal of Combinatorial Theory, Series B*, 44(3):355–362, 1988.

[66] P. Gehler and S. Nowozin. Infinite kernel learning. *MPI Technical Report 178*, 2008.

[67] K. Ghiasi-Shirazi, R. Safabakhsh, and M. Shamsi. Learning translation invariant kernels for classification. *Journal of Machine Learning Research*, 11:1353–1390, 2010.

[68] J. Gillenwater, K. Ganchev, J. Graça, F. Pereira, and B. Taskar. Posterior sparsity in unsupervised dependency parsing. *Journal of Machine Learning Research*, 12:455–490, 2011.

[69] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[70] M. Gönen and E. Alpaydın. Multiple kernel learning algorithms. *Journal of Machine Learning Research*, 12:2211–2268, 2011.

[71] Y. Gong, S. Kumar, V. Verma, and S. Lazebnik. Angular quantization-based binary codes for fast similarity search. In *Advances in Neural Information Processing Systems*, 2012.

[72] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1, 2012.

[73] Y. Gong, S. Kumar, H. A. Rowley, and S. Lazebnik. Learning binary codes for high-dimensional data using bilinear projections. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[74] A. Gordo and F. Perronnin. Asymmetric distances for binary embeddings. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[75] A. Gordoa, J. Rodrıguez-Serranoa, F. Perronnina, and E. Valvenyb. Leveraging category-level labels for instance-level image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[76] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised on-line boosting for robust tracking. In *Proceedings of the European Conference on Computer Vision*, 2008.

[77] D. Grangier and S. Bengio. A discriminative kernel-based model to rank images from text queries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30 (8):1371–1384, 2008.

[78] R. M. Gray. *Toeplitz and circulant matrices: A review.* Now Pub, 2006.

[79] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Discriminative metric learning in nearest neighbor models for image auto-annotation. In *Proceedings of the IEEE International Conference on Computer Vision*, 2009.

[80] R. Hamid, Y. Xiao, A. Gittens, and D. Decoste. Compact random feature maps. In *Proceedings of the International Conference on Machine Learning*, 2014.

[81] D. Haussler and P. M. Long. A generalization of Sauer's lemma. *Journal of Combinatorial Theory, Series A*, 71(2):219–240, 1995.

[82] A. Hinrichs and J. Vybíral. Johnson-Lindenstrauss lemma for circulant matrices. *Random Structures & Algorithms*, 39(3):391–398, 2011.

[83] G. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Improving neural networks by preventing coadaptation of feature detectors. In *arXiv preprint arXiv:1207.0580*, 2012.

[84] G. Hinton, O. Vinyals, and J. Dean. Distilling knowledge in a neural network. In *Deep Learning and Representation Learning Workshop, NIPS*, 2014.

[85] C.-J. Hsieh, S. Si, and I. Dhillon. A divide-and-conquer solver for kernel support vector machines. In *Proceedings of the International Conference on Machine Learning*, 2014.

[86] C.-J. Hsieh, S. Si, and I. S. Dhillon. Fast prediction for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, 2014.

[87] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. *University of Massachusetts, Amherst, Technical Report 07-49*, 2007.

[88] P.-S. Huang, H. Avron, T. N. Sainath, V. Sindhwani, and B. Ramabhadran. Kernel methods match deep neural networks on timit. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 2014.

[89] P. Indyk and R. Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the ACM Symposium on Theory of Computing*, 1998.

[90] M. Jaderberg, A. Vedaldi, and A. Zisserman. Deep features for text spotting. In *Proceedings of the European Conference on Computer Vision*, 2014.

[91] M. Jaderberg, A. Vedaldi, and A. Zisserman. Speeding up convolutional neural networks with low rank expansions. In *Proceedings of the British Machine Vision Conference*, 2014.

[92] H. Jegou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2011.

[93] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[94] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui. Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In *Proceedings of the Annual ACM International Conference on Multimedia Retrieval*, 2011.

[95] Y.-G. Jiang, S. Bhattacharya, S.-F. Chang, and M. Shah. High-level event recognition in unconstrained videos. *International Journal of Multimedia Information Retrieval*, pages 1–29, 2012.

[96] T. Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of the International Conference on Machine Learning*, 1999.

[97] T. Joachims. Training linear svms in linear time. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2006.

[98] T. Joachims, T. Finley, and C. Yu. Cutting-plane training of structural svms. *Machine Learning*, 77(1):27–59, 2009.

[99] W. B. Johnson and J. Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. *Contemporary Mathematics*, 26(189-206):1, 1984.

[100] C. Jose, P. Goyal, P. Aggrwal, and M. Varma. Local deep kernel learning for efficient non-linear svm prediction. In *Proceedings of the International Conference on Machine Learning*, 2013.

[101] M. Juneja, A. Vedaldi, C. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[102] P. Kankuekul, A. Kawewong, S. Tangruamsub, and O. Hasegawa. Online incremental attribute-based zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[103] P. Kar and H. Karnick. Random feature maps for dot product kernels. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2012.

[104] S. Keerthi, O. Chapelle, and D. DeCoste. Building support vector machines with reduced classifier complexity. *Journal of Machine Learning Research*, 7:1493–1515, 2006.

[105] S. Keerthi, V. Sindhwani, and O. Chapelle. An efficient method for gradient-based adaptation of hyperparameters in svm models. In *Advances in Neural Information Processing Systems*, 2006.

[106] S. Keerthi, S. Sundararajan, and S. Shevade. Extension of TSVM to multi-class and hierarchical text classification problems with general losses. In *Proceedings of the International Conference on Computational Linguistics*, 2012.

[107] J. Kelley Jr. The cutting-plane method for solving convex programs. *Journal of the Society for Industrial & Applied Mathematics*, 8(4):703–712, 1960.

[108] I. R. Kondor. *Group Theoretical Methods in Machine Learning*. PhD thesis, Department of Computer Science, Columbia University, 2008.

[109] J. Krapac, M. Allan, J. Verbeek, and F. Jurie. Improving web image search results using query-relative classifiers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[110] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012.

[111] H. Kuck and N. de Freitas. Learning about individuals from group statistics. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 2005.

[112] B. Kulis and T. Darrell. Learning to hash with binary reconstructive embeddings. In *Advances in Neural Information Processing Systems*, 2009.

[113] N. Kumar, A. Berg, P. Belhumeur, and S. Nayar. A search engine for large collections of images with faces. In *Proceedings of the European Conference on Computer Vision*, 2008.

[114] N. Kumar, A. Berg, P. Belhumeur, and S. Nayar. Attribute and simile classifiers for face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[115] S. Kumar, M. Mohri, and A. Talwalkar. Sampling techniques for the nystrom method. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2009.

[116] L. Ladicky and P. Torr. Locally linear support vector machines. In *Proceedings of the International Conference on Machine Learning*, 2011.

[117] K.-T. Lai, D. Liu, M.-S. Chen, and S.-F. Chang. Recognizing complex events in videos by learning key static-dynamic evidences. In *Proceedings of the European Conference on Computer Vision*, 2014.

[118] K.-T. Lai, F. X. Yu, M.-S. Chen, and S.-F. Chang. Video event detection by inferring temporal instance labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[119] C. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[120] C. H. Lampert. *Kernel methods in computer vision*. Now Publishers Inc, 2009.

[121] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[122] Q. Le and A. Smola. Direct optimization of ranking measures. *arXiv preprint arXiv:0704.3359*, 2007.

[123] Q. Le, T. Sarlós, and A. Smola. Fastfood – approximating kernel expansions in loglinear time. In *Proceedings of the International Conference on Machine Learning*, 2013.

[124] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the the IEEE*, pages 2278–2324, 1998.

[125] F. Li, C. Ionescu, and C. Sminchisescu. Random fourier approximations for skewed multiplicative histogram kernels. *Pattern Recognition*, pages 262–271, 2010.

[126] L. Li, H. Su, E. Xing, and L. Fei-Fei. Object bank: A high-level image representation for scene classification and semantic feature sparsification. In *Advances in Neural Information Processing Systems*, 2010.

[127] P. Li, A. Shrivastava, J. Moore, and A. C. Konig. Hashing algorithms for large-scale learning. In *Advances in Neural Information Processing Systems*, 2011.

[128] W. Li, Q. Yu, A. Divakaran, and N. Vasconcelos. Dynamic pooling for complex event recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 2013.

[129] Y. Li, J. Kwok, and Z. Zhou. Semi-supervised learning using label mean. In *Proceedings of the International Conference on Machine Learning*, 2009.

[130] Y. Li, I. Tsang, J. Kwok, and Z. Zhou. Tighter and convex maximum margin clustering. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2009.

[131] E. Liberty, N. Ailon, and A. Singer. Dense fast random projections and lean walsh transforms. *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*, pages 512–522, 2008.

[132] M. Lin, Q. Chen, and S. Yan. Network in network. In *Proceedings of the International Conference on Learning Representations*, 2014.

[133] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[134] W. Liu, J. Wang, S. Kumar, and S.-F. Chang. Hashing with graphs. In *Proceedings of the International Conference on Machine Learning*, 2011.

[135] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[136] D. Mahajan, S. Sellamanickam, and V. Nair. A joint learning framework for attribute models and object descriptions. In *Proceedings of the IEEE International Conference on Computer Vision*, 2011.

[137] S. Maji and A. C. Berg. Max-margin additive classifiers for detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 2009.

[138] G. Mann and A. McCallum. Simple, robust, scalable semi-supervised learning via expectation regularization. In *Proceedings of the International Conference on Machine Learning*, 2007.

[139] O. Maron and A. L. Ratan. Multiple-instance learning for natural scene classification. In *Proceedings of the International Conference on Machine Learning*, 1998.

[140] D. K. Maslen and D. N. Rockmore. Generalized FFTs - a survey of some recent results. In *Groups and Computation II*, volume 28, pages 183–287. American Mathematical Soc., 1997.

[141] M. Mathieu, M. Henaff, and Y. LeCun. Fast training of convolutional networks through FFTs. *arXiv preprint arXiv:1312.5851*, 2013.

[142] J. Matoušek. On variants of the Johnson–Lindenstrauss lemma. *Random Structures & Algorithms*, 33(2):142–156, 2008.

[143] M. D. McDonnell and T. Vladusich. Enhanced image classification with a fast-learning shallow convolutional neural network. *arXiv preprint arXiv:1503.04596*, 2015.

[144] M. D. McDonnell, M. D. Tissera, A. van Schaik, and J. Tapson. Fast, simple and accurate handwritten digit classification using extreme learning machines with shaped input-weights. *arXiv preprint arXiv:1412.8307*, 2014.

[145] M. Merler, B. Huang, L. Xie, G. Hua, and A. Natsev. Semantic model vectors for complex video event recognition. *IEEE Transactions on Multimedia*, 14(1):88–101, 2012.

[146] P. Mermelstein. Distance measures for speech recognition, psychological and instrumental. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 116:374–388, 1976.

[147] Y. Mu, J. Shen, and S. Yan. Weakly-supervised hashing in kernel space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[148] G. Murphy. *The big book of concepts*. MIT Press, 2004.

[149] D. Musicant, J. Christensen, and J. Olson. Supervised learning by training on aggregate outputs. In *Proceedings of the IEEE International Conference on Data Mining*, 2007.

[150] M. Naphade, J. Smith, J. Tesic, S.-F. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and C. J. Large-scale concept ontology for multimedia. *IEEE Multimedia*, 2006.

[151] A. Natsev, A. Haubold, J. Tešić, L. Xie, and R. Yan. Semantic concept-based query expansion and re-ranking for multimedia retrieval. In *Proceedings of the ACM Multimedia Conference*, 2007.

[152] M. Norouzi and D. Fleet. Minimal loss hashing for compact binary codes. In *Proceedings of the International Conference on Machine Learning*, 2012.

[153] M. Norouzi, D. Fleet, and R. Salakhutdinov. Hamming distance metric learning. In *Advances in Neural Information Processing Systems*, 2012.

[154] A. V. Oppenheim, R. W. Schafer, J. R. Buck, et al. *Discrete-time signal processing*, volume 5. Prentice Hall Upper Saddle River, 1999.

[155] E. Osuna, R. Freund, and F. Girosi. Support vector machines: Training and applications. *Massachusetts Institute of Technology Technical Report*, 1997.

[156] P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A. F. Smeaton, and G. Quéenot. TRECVID 2013 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *TRECVID 2013*, 2013.

[157] M. Palatucci, D. Pomerleau, G. Hinton, and T. Mitchell. Zero-shot learning with semantic output codes. In *Advances in Neural Information Processing Systems*, 2009.

[158] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[159] D. Parikh and K. Grauman. Relative attributes. In *Proceedings of the IEEE International Conference on Computer Vision*, 2011.

[160] D. Parikh and K. Grauman. Interactively building a discriminative vocabulary of nameable attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[161] A. Parkash and D. Parikh. Attributes for classifier feedback. In *Proceedings of the European Conference on Computer Vision*, 2012.

[162] G. Patrini, R. Nock, T. Caetano, and P. Rivera. (almost) no label no cry. In *Advances in Neural Information Processing Systems*, 2014.

[163] G. Patterson and J. Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[164] D. Pavlov, D. Chudova, and P. Smyth. Towards scalable support vector machines using squashing. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2000.

[165] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.

[166] F. Perronnin, J. Sánchez, and T. Mensink. Improving the Fisher kernel for large-scale image classification. In *Proceedings of the European Conference on Computer Vision*, 2010.

[167] J. Petterson and T. Caetano. Reverse multi-label learning. In *Advances in Neural Information Processing Systems*, 2010.

[168] N. Pham and R. Pagh. Fast and scalable polynomial kernels via explicit feature maps. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2013.

[169] N. Quadrianto, A. Smola, T. Caetano, and Q. Le. Estimating labels from label proportions. *Journal of Machine Learning Research*, 10:2349–2374, 2009.

[170] M. Raginsky and S. Lazebnik. Locality-sensitive binary codes from shift-invariant kernels. In *Advances in Neural Information Processing Systems*, 2009.

[171] A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, 2007.

[172] A. Rakotomamonjy, F. Bach, S. Canu, Y. Grandvalet, et al. Simplemkl. *Journal of Machine Learning Research*, 9:2491–2521, 2008.

[173] C. R. Rao. The utilization of multiple measurements in problems of biological classification. *Journal of the Royal Statistical Society. Series B (Methodological)*, 10(2): 159–203, 1948.

[174] M. Rastegari, A. Farhadi, and D. Forsyth. Attribute discovery via predictable discriminative binary codes. In *Proceedings of the European Conference on Computer Vision*, 2012.

[175] D. Rockmore. Some applications of generalized ffts. In *Proceedings of the 1995 DIMACS Workshop on Groups and Computation*, 1997.

[176] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele. What helps where– and why? semantic relatedness for knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[177] A. Romero, N. Ballas, S. Kahou, A. Chassang, C. Gatta, and Y. Bengio. FitNets: Hints for thin deep nets. In *Proceedings of the International Conference on Learning Representations*, 2015.

[178] C. Rosenberg, M. Hebert, and H. Schneiderman. Semi-supervised self-training of object detection models. In *IEEE Workshop on Applications of Computer Vision / IEEE Workshop on Motion and Video Computing*, 2005.

[179] S. Rüeping. SVM classifier estimation from group probabilities. In *Proceedings of the International Conference on Machine Learning*, 2010.

[180] S. Sabato. *Partial Information and Distribution-Dependence in Supervised Learning Models*. PhD thesis, School of Computer Science and Engineering, Hebrew University of Jerusalem, 2012.

[181] S. Sabato and N. Tishby. Multi-instance learning with any hypothesis class. *Journal of Machine Learning Research*, 13:2999–3039, 2012.

[182] T. Sainath, B. Kingsbury, V. Sindhwani, E. Arisoy, and B. Ramabhadran. Low-Rank Matrix Factorization for Deep Neural Network Training with High-Dimensional Output Targets. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 2013.

[183] J. Sánchez and F. Perronnin. High-dimensional signature compression for large-scale image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[184] N. Sauer. On the density of families of sets. *Journal of Combinatorial Theory, Series A*, 13(1):145–147, 1972.

[185] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *Proceedings of the International Conference on Learning Representations*, 2014.

[186] B. Settles. Active learning literature survey. *University of Wisconsin, Madison*, 52 (55-66):11, 2010.

[187] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter. Pegasos: Primal estimated sub-gradient solver for svm. *Mathematical Programming*, 127(1):3–30, 2011.

[188] V. Sharmanska, N. Quadrianto, and C. Lampert. Augmented attribute representations. In *Proceedings of the European Conference on Computer Vision*, 2010.

[189] A. Shilton, M. Palaniswami, D. Ralph, and A. Tsoi. Incremental training of support vector machines. *IEEE Transactions on Neural Networks*, 16(1):114–131, 2005.

[190] B. Siddiquie, R. Feris, and L. Davis. Image ranking and retrieval based on multi-attribute queries. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[191] V. Sindhwani, S. Keerthi, and O. Chapelle. Deterministic annealing for semi-supervised kernel machines. In *Proceedings of the International Conference on Machine Learning*, 2006.

[192] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, 2003.

[193] H. Soltau, G. Saon, and T. Sainath. Joint training of convoutional and non-convoutional neural networks. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 2014.

[194] A. Sorokin and D. Forsyth. Utility data annotation with amazon mechanical turk. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*, 2008.

[195] V. Sreekanth, A. Vedaldi, A. Zisserman, and C. Jawahar. Generalized RBF feature maps for efficient detection. In *Proceedings of the British Machine Vision Conference*, 2010.

[196] M. Stolpe and K. Morik. Learning from label proportions by optimizing cluster model selection. *Machine Learning and Knowledge Discovery in Databases*, pages 349–364, 2011.

[197] C. Sun and R. Nevatia. Discover: Discovering important segments for classification of video events and recounting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[198] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[199] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *arXiv preprint arXiv:1409.4842*, 2014.

[200] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[201] K. Tang, L. Fei-Fei, and D. Koller. Learning latent temporal structure for complex event detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[202] L. Torresani and A. Bergamo. Meta-class features for large-scale object categorization on a budget. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[203] L. Torresani, M. Szummer, , and A. Fitzgibbon. Efficient object category recognition using classemes. In *Proceedings of the European Conference on Computer Vision*, 2010.

[204] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6(2):1453–1484, 2006.

[205] K. van de Sande, J. Uijlings, T. Gevers, and A. Smeulders. Segmentation as selective search for object recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 2011.

[206] V. Vanhoucke, A. Senior, and M. Z. Mao. Improving the speed of neural networks on cpus. In *Deep Learning and Unsupervised Feature Learning Workshop, NIPS*, 2011.

[207] V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16 (2):264–280, 1971.

[208] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(3):480–492, 2012.

[209] L. Von Ahn and L. Dabbish. Labeling images with a computer game. In *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems*, 2004.

[210] L. Von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum. recaptcha: Human-based character recognition via web security measures. *Science*, 321(5895):1465–1468, 2008.

[211] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4): 395–416, 2007.

[212] J. Vybíral. A variant of the Johnson–Lindenstrauss lemma for circulant matrices. *Journal of Functional Analysis*, 260(4):1096–1105, 2011.

[213] G. Wang, D. Forsyth, and D. Hoiem. Comparative object similarity for improved recognition with few or no examples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[214] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu. Action recognition by dense trajectories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[215] J. Wang, S. Kumar, and S.-F. Chang. Semi-supervised hashing for scalable image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[216] J. Wang, S. Kumar, and S.-F. Chang. Sequential projection learning for hashing with compact codes. In *Proceedings of the International Conference on Machine Learning*, 2010.

[217] Y. Wang and G. Mori. A discriminative latent model of object classes and attributes. In *Proceedings of the European Conference on Computer Vision*, 2010.

[218] Y. Weiss, A. Torralba, and R. Fergus. Spectral hashing. In *Advances in Neural Information Processing Systems*, 2008.

[219] C. Williams and M. Seeger. Using the nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems*, 2001.

[220] E. P. Xing, M. I. Jordan, S. Russell, and A. Y. Ng. Distance metric learning with application to clustering with side-information. In *Advances in Neural Information Processing Systems*, 2002.

[221] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans. Maximum margin clustering. In *Advances in Neural Information Processing Systems*, 2004.

[222] A. Yanagawa, S.-F. Chang, L. Kennedy, and W. Hsu. Columbia University's baseline detectors for 374 LSCOM semantic visual concepts. *Columbia University ADVENT Technical Report # 222-2006-8*, 2007.

[223] J. Yang, V. Sindhwani, H. Avron, and M. Mahoney. Quasi-monte carlo feature maps for shift-invariant kernels. In *Proceedings of the International Conference on Machine Learning*, 2014.

[224] J. Yang, V. Sindhwani, Q. Fan, H. Avron, and M. Mahoney. Random laplace feature maps for semigroup kernels on histograms. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[225] T. Yang, Y.-F. Li, M. Mahdavi, R. Jin, and Z.-H. Zhou. Nyström method vs random fourier features: A theoretical and empirical comparison. In *Advances in Neural Information Processing Systems*, 2012.

[226] Y. Yang and M. Shah. Complex events detection using data-driven concepts. In *Proceedings of the European Conference on Computer Vision*, 2012.

[227] D. Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 1995.

[228] F. X. Yu, R. Ji, and S.-F. Chang. Active query sensing for mobile location search. In *Proceedings of the ACM Multimedia Conference*, 2011.

[229] F. X. Yu, R. Ji, M.-H. Tsai, G. Ye, and S.-F. Chang. Weak attributes for large-scale image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[230] F. X. Yu, L. Cao, R. Feris, J. Smith, and S.-F. Chang. Designing category-level attributes for discriminative visual recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[231] F. X. Yu, D. Liu, S. K., T. Jebara, and S.-F. Chang. $\propto$SVM for learning with label proportions. In *Proceedings of the International Conference on Machine Learning*, 2013.

[232] F. X. Yu, L. Cao, M. Merler, T. Chen, J. Smith, and S.-F. Chang. Modeling attributes from category-attribute proportions. In *Proceedings of the ACM Multimedia Conference*, 2014.

[233] F. X. Yu, K. Choromanski, S. Kumar, T. Jebara, and S.-F. Chang. On learning with label proportions. *arXiv preprint arXiv:1402.5902*, 2014.

[234] F. X. Yu, S. Kumar, Y. Gong, and S.-F. Chang. Circulant binary embedding. In *Proceedings of the International Conference on Machine Learning*, 2014.

[235] F. X. Yu, S. Kumar, H. Rowley, and S.-F. Chang. Compact nonlinear maps and circulant extensions. *arXiv preprint arXiv:1503.03893*, 2015.

[236] X. Yu and Y. Aloimonos. Attribute-based transfer learning for object categorization with zero/one training example. In *Proceedings of the European Conference on Computer Vision*, 2010.

[237] M. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *Proceedings of the European Conference on Computer Vision*, 2014.

[238] D. Zhang, J. He, L. Si, and R. D. Lawrence. Mileage: Multiple instance learning with global embedding. In *Proceedings of the International Conference on Machine Learning*, 2013.

[239] H. Zhang and L. Cheng. New bounds for circulant Johnson-Lindenstrauss embeddings. *arXiv preprint arXiv:1308.6339*, 2013.

[240] Q. Zhang, S. A. Goldman, W. Yu, and J. E. Fritts. Content-based image retrieval using multiple-instance learning. In *Proceedings of the International Conference on Machine Learning*, 2002.

[241] X. Zhu. Semi-supervised learning literature survey. *Technical Report 1530, University of Wisconsin – Madison*, 2008.

Part VI

Appendix

Appendix A

Additional Proofs

A.1 Proof of Lemma 3.1

Proof. For convenience, define $\mathbf{u}^\perp = \mathbf{u} - \Pi\mathbf{u}$, and similarly define $\mathbf{v}^\perp$. Based on Lemma 3.1,

$$\mathbb{E}\left[\left(\frac{1-\operatorname{sign}(\mathbf{r}^T\mathbf{a})\operatorname{sign}(\mathbf{r}^T\mathbf{b})}{2}-\frac{\theta}{\pi}\right)\left(\frac{1-\operatorname{sign}(\mathbf{r}^T\mathbf{u}^\perp)\operatorname{sign}(\mathbf{r}^T\mathbf{v}^\perp)}{2}-\frac{\theta}{\pi}\right)\right]=0. \tag{A.1}$$

Thus the quantity we wish to bound is

$$\mathbb{E}\left[\left(\frac{1-\operatorname{sign}(\mathbf{r}^T\mathbf{a})\operatorname{sign}(\mathbf{r}^T\mathbf{b})}{2}-\frac{\theta}{\pi}\right)\left(\frac{\operatorname{sign}(\mathbf{r}^T\mathbf{u})\operatorname{sign}(\mathbf{r}^T\mathbf{v})-\operatorname{sign}(\mathbf{r}^T\mathbf{u}^\perp)\operatorname{sign}(\mathbf{r}^T\mathbf{v}^\perp)}{2}\right)\right].$$

Now by using the fact that $\mathbb{E}[XY] \le \mathbb{E}[|X||Y|]$, together with the observation that the quantity $|(1-\operatorname{sign}(\mathbf{r}^T\mathbf{a})\operatorname{sign}(\mathbf{r}^T\mathbf{b}))/2-\theta/\pi|$ is at most 2, we can bound the above by

$$\mathbb{E}\left[|\operatorname{sign}(\mathbf{r}^T\mathbf{u})\operatorname{sign}(\mathbf{r}^T\mathbf{v})-\operatorname{sign}(\mathbf{r}^T\mathbf{u}^\perp)\operatorname{sign}(\mathbf{r}^T\mathbf{v}^\perp)|\right].$$

This is equal to

$$2\mathbb{P}[\operatorname{sign}(\mathbf{r}^T\mathbf{u})\operatorname{sign}(\mathbf{r}^T\mathbf{v}) \neq \operatorname{sign}(\mathbf{r}^T\mathbf{u}^\perp)\operatorname{sign}(\mathbf{r}^T\mathbf{v}^\perp)],$$

since the term in the expectation is 2 if the product of signs is different, and 0 otherwise. To bound this, we first observe that for any two unit vectors $\mathbf{x}, \mathbf{y}$ with $\angle(\mathbf{x},\mathbf{y}) \le \epsilon$, we have $\mathbb{P}[\operatorname{sign}(\mathbf{r}^T\mathbf{x}) \neq \operatorname{sign}(\mathbf{r}^T\mathbf{y})] \le \epsilon/\pi$ (it follows from (3.5)). We can use this to say that

$$\mathbb{P}[\operatorname{sign}(\mathbf{r}^T\mathbf{u}) \neq \operatorname{sign}(\mathbf{r}^T\mathbf{u}^\perp)] = \frac{\angle(\mathbf{u},\mathbf{u}^\perp)}{\pi}.$$

This angle can be bounded in our case by $(\pi/2) \cdot \delta$ by basic geometry.[1] Thus by a union bound, we have that

$$\mathbb{P}[\big(\operatorname{sign}(\mathbf{r}^T\mathbf{u}) \neq \operatorname{sign}(\mathbf{r}^T\mathbf{u}^\perp)\big) \vee \big(\operatorname{sign}(\mathbf{r}^T\mathbf{v}) \neq \operatorname{sign}(\mathbf{r}^T\mathbf{v}^\perp)\big)] \leq \delta.$$

This completes the proof. □

A.2 Proof of Lemma 3.3

Proof. We are to show that $\mathbb{P}[s_{\to t}(\mathbf{Dx})^T\mathbf{Dz} \geq \gamma] \leq e^{-\gamma^2/8\rho^2}$.

We have $s_{\to t}(\mathbf{Dx})^T\mathbf{Dz} = \sum_{i=0}^{d-1} z_i x_{i+t}\sigma_i\sigma_{i+t}$ (the subindex is modulo d), let us define

$$f(\sigma_0, \sigma_1, \cdots, \sigma_{d-1}) := \sum_{i=0}^{d-1} z_i x_{i+t}\sigma_i\sigma_{i+t}$$

and

$$Z_i := f(\sigma_0, \sigma_1, \cdots, \sigma_i, 0, \cdots, 0).$$

In this notion we have $s_{\to t}(\mathbf{Dx})^T\mathbf{Dz} = Z_{d-1} - Z_0$. For all i, we have

$$\mathbb{E}(Z_i - Z_{i-1}|Z_0, \cdots, Z_{i-1}) = 0.$$

This is because σ_i is ± 1 with equal probability. Therefore, the sequence $Z_0, Z_1, \cdots$ is a martingale. Further, we have $|Z_i - Z_{i-1}| \leq |z_i x_{i+t}| + |z_{i-t}x_i|$. Based on Azuma's inequality: for any $\gamma > 0$,

$$\begin{aligned}\Pr\left[|Z_{d-1} - Z_0| \geq \gamma\right] &\leq e^{-\frac{\gamma^2}{2\sum_{i=0}^{d-1}(|z_i x_{i+t}| + |z_{i-t}x_i|)^2}} \\ &\leq e^{-\frac{\gamma^2}{8\rho^2}}.\end{aligned}$$

The last step is based on the fact that

[1] $\mathbf{u}$ is a unit vector, and $\mathbf{u}^\perp + \Pi\mathbf{u} = \mathbf{u}$, and $\|\Pi\mathbf{u}\| \leq \delta$, so the angle is at most $\sin^{-1}(\delta)$.

$$
\begin{aligned}
& \sum_{i=0}^{d-1}(|z_i x_{i+t}| + |z_{i-t} x_i|)^2 \\
= & \sum_{i=0}^{d-1} z_i^2 x_{i+t}^2 + x_i^2 z_{i-t}^2 + 2|z_i x_{i+t} z_{i-t} x_i| \\
\leq & \sum_{i=0}^{d-1} z_i^2 x_{i+t}^2 + x_i^2 z_{i-t}^2 + z_i^2 z_{i-t}^2 + x_{i+t}^2 x_i^2 \\
\leq & \ 4\rho^2.
\end{aligned}
$$

□

A.3 Proof of Theorem 7.1

Proof. One important tool used in the proof is the lemma below bounding the covering number of bag proportion hypothesis class $\bar{\mathcal{H}}$ by the covering number of the instance hypothesis class $\mathcal{H}$.

Lemma A.1. *[181, 180] Let $r \in \mathbb{N}$. Let $\gamma > 0$, $p \in [1, \infty]$, and $\mathcal{H} \in \mathbb{R}^{\mathcal{X}}$. For any $M \geq 0$,*

$$\mathcal{N}_p(\gamma, \bar{\mathcal{H}}, M) \leq \mathcal{N}_p(\frac{\gamma}{r^{1/p}}, \mathcal{H}, rM).$$

Covering number [6] can be seen as a complexity measure on real-valued hypothesis class. The larger the covering number, the larger the complexity. Another lemma we use is the uniform convergence for real function class.

Lemma A.2. *[6]. Let $\hat{\mathcal{Y}}, \mathcal{Y} \subseteq \mathbb{R}$, $\mathcal{G} \subseteq \hat{\mathcal{Y}}^{\mathcal{X}}$, and $L : \hat{\mathcal{Y}} \times \mathcal{Y} \to [0, 1]$, such that L is Lipschitz in its first argument with Lipschitz constant $\alpha_L > 0$. Let D be any distribution on $\mathcal{X}$. Let S be a set of M* iid *samples generated from D, then for any $0 < \epsilon < 1$ and $g \in \mathcal{G}$:*

$$\mathbb{P}\left(\sup_{g \in \mathcal{G}} \left| er_D^L(g) - er_S^L(g) \right| \geq \epsilon \right) \leq 4\mathcal{N}_1(\epsilon/(8\alpha_L), \mathcal{G}, 2M) e^{-M\epsilon^2/32},$$

in which $er_S^L(g) = \frac{1}{|S|} \sum_{x \in S} L(g(x), y)$, $er_D^L(g) = \mathbb{E}_{x \sim D} L(g(x), y)$.

Based on the definition of covering number: $\mathcal{N}_1(\epsilon, W, M) \leq \mathcal{N}_\infty(\epsilon, W, M)$. Applying Lemma A.1:

$$\begin{aligned} 4\mathcal{N}_1(\epsilon/(8\alpha_L), \mathcal{H}, 2M)e^{-M\epsilon^2/32} \leq & 4\mathcal{N}_\infty(\epsilon/(8\alpha_L), \mathcal{H}, 2M)e^{-M\epsilon^2/32} \\ \leq & 4\mathcal{N}_\infty(\epsilon/(8\alpha_L), \mathcal{H}, 2rM)e^{-M\epsilon^2/32}. \end{aligned} \tag{A.2}$$

The loss functions L we are considering is 1-Lipschitz. Based on the definition of covering number, for $\mathcal{H} \subseteq \{-1, 1\}^{\mathcal{X}}$, for any $\epsilon < 2$, $\mathcal{N}(\epsilon, \mathcal{H}_{|x_1^M}, d_\infty) = |\mathcal{H}_{|x_1^M}|$. Thus,

$$\mathcal{N}_\infty(\epsilon, \mathcal{H}, M) = \Pi_{\mathcal{H}}(M).$$

Refer to [6] for the definition of *restriction* $\mathcal{H}_{|x_1^M}$ and *growth function* $\Pi_{\mathcal{H}}(M)$. In addition, we have the following lemma to to bound the growth function by VC dimension of the hypothesis class.

Lemma A.3. *[184] Let $\mathcal{G} \subseteq \{-1, 1\}^{\mathcal{X}}$ with $VC(\mathcal{G}) = d \leq \infty$. For all $M \geq d$, $\Pi_{\mathcal{G}}(M) \leq \sum_{i=0}^{d} \binom{M}{i} \leq \left(\frac{eM}{d}\right)^d$.*

Let $d = VC(\mathcal{H})$. By combining the above facts, and $0 < \epsilon < 1$, (A.2) leads to

$$4\Pi_{\mathcal{H}}(2rM)e^{-M\epsilon^2/32} \leq 4\left(\frac{2erM}{d}\right)^d e^{-M\epsilon^2/32}.$$

Therefore, with probability at least $1 - \delta$,

$$\begin{aligned} & \mathrm{er}_D^L(f) \leq \mathrm{er}_S^L(f) + \left(\frac{32}{M}(d\ln(2eMr/d) + \ln(4/\delta))\right)^{1/2} \\ \Leftarrow\ & M \leq \frac{32}{\epsilon^2}(d\ln M + d\ln(2er/d) + \ln(4/\delta)). \end{aligned}$$

Since $\ln x \leq ax - \ln a - 1$ for all $a, x > 0$, we have

$$\begin{aligned} \Leftarrow\ & \frac{32d}{\epsilon^2}\ln M \leq \frac{32d}{\epsilon^2}\left(\frac{\epsilon^2}{64d}M + \ln\left(\frac{64d}{\epsilon^2}\right)\right) \leq \frac{M}{2} + \frac{32d}{\epsilon^2}\ln\left(\frac{64d}{e\epsilon^2}\right). \\ \Leftarrow\ & M \geq \frac{M}{2} + \frac{32}{\epsilon^2}(d\ln(128r/\epsilon^2) + \ln(4/\delta)). \\ \Leftarrow\ & M \geq \frac{64}{\epsilon^2}(2d\ln(12r/\epsilon) + \ln(4/\delta)). \end{aligned}$$

□

A.4 Proof of Proposition 7.2

Proof. Let B be a bag containing $\mathbf{x}_1, \cdots, \mathbf{x}_r$. Assume that $|\bar{h}(B) - \bar{y}(B)| \leq \epsilon$. Let y_i be the ground-truth label for $\mathbf{x}_i$. Let

$$\mathcal{A}_1 = \{i \in \{1, ..., r\} : h(\mathbf{x}_i) = +1 \wedge y(\mathbf{x}_i) = -1\},$$

$$\mathcal{A}_2 = \{i \in \{1, ..., r\} : h(\mathbf{x}_i) = -1 \wedge y(\mathbf{x}_i) = +1\},$$

$$\mathcal{A}_3 = \{i \in \{1, ..., r\} : h(\mathbf{x}_i) = +1 \wedge y(\mathbf{x}_i) = +1\},$$

$$\mathcal{A}_4 = \{i \in \{1, ..., r\} : h(\mathbf{x}_i) = -1 \wedge y(\mathbf{x}_i) = -1\}.$$

We have: $\bar{h}(B) = (|\mathcal{A}_1| + |\mathcal{A}_3|)/r$, $\bar{y}(B) = (|\mathcal{A}_2| + |\mathcal{A}_3|)/r$. Thus, from the assumption: $|\bar{h}(B) - \bar{y}(B)| \leq \epsilon$, we have: $||\mathcal{A}_1| - |\mathcal{A}_2|| \leq \epsilon r$.

Assume that B is $(1-\eta)$-*pure*. Without loss of generality we can assume that $|\mathcal{A}_1| + |\mathcal{A}_4| \leq \eta r$. This implies that $|\mathcal{A}_1| \leq \eta r$. Thus we also have that $|\mathcal{A}_2| \leq (\eta + \epsilon)r$. Hypothesis h correctly classifies $|\mathcal{A}_3| + |\mathcal{A}_4|$ instances of the bag B. From what we have just derived, we conclude that h correctly classifies at least $(1 - 2\eta - \epsilon)r$ instances of the bag B.

So far in the analysis above we assumed that: $|\bar{h}(B) - \bar{y}(B)| \leq \epsilon$ and B is $(1-\eta)$-*pure*. From the statement of the theorem we know that the former happens with probability at least $1-\delta$ and the latter with probability at least $1-\rho$. Thus, by the union bound, we know that with probability at least $1 - \delta - \rho$, h classifies correctly at least $(1 - 2\eta - \epsilon)$ instances of the bag drawn from the distribution $\mathcal{D}$. □

A.5 Proof of Proposition 7.3

Proof. This can be shown by construction. Let B be a bag containing $\mathbf{x}_1, \cdots, \mathbf{x}_r$. Assume that the bag is formed such that

$$y(\mathbf{x}_i) = \begin{cases} 1, & 1 \leq i \leq \eta r. \\ -1, & \text{otherwise.} \end{cases}$$

Assume that the hypothesis h satisfies

$$h(\mathbf{x}_i) = \begin{cases} -y(\mathbf{x}_i), & 1 \leq i \leq 2\eta r. \\ y(\mathbf{x}_i), & \text{otherwise.} \end{cases}$$

Then h predicts the bag proportions with 0 error, yet misclassifies 2η instances. □

A.6 Proof of Proposition 8.1

Proof. We consider the k-th bag in this proof. We first note that the influence of y_i, $\forall i \in \mathcal{B}_k$ to the first term of the objective function, $\sum_{i\in\mathcal{B}_k} L'(y_i, \boldsymbol{w}^T\varphi(\boldsymbol{x}_i)+b)$, is independent. Without loss of generality, we assume $\mathcal{B}_k = \{1, \cdots, |\mathcal{B}_k|\}$. Also without loss of generality, we assume δ_i's are already in sorted order, i.e. $\delta_1 \geq \delta_2 \geq ... \geq \delta_{|\mathcal{B}_k|}$.

Define $\mathcal{B}_k^+ = \{i|y_i = 1, i \in \mathcal{B}_k\}$, and $\mathcal{B}_k^- = \{i|y_i = -1, i \in \mathcal{B}_k\}$. In order to satisfy the label proportion, the number of elements in $\{y_i|i \in \mathcal{B}_k\}$ to be flipped is $\theta|\mathcal{B}_k|$. We are to solve the following optimization problem.

$$\max_{\mathcal{B}_k^+} \quad \sum_{i\in\mathcal{B}_k^+} \delta_i - \sum_{i\in\mathcal{B}_k^-} \delta_i, \quad \text{s.t.} \quad |\mathcal{B}_k^+| = \theta|\mathcal{B}_k|.$$

What we need to prove is that $\mathcal{B}_k^+ = \{1, 2, ..., \theta|\mathcal{B}_k|\}$ is optimal.

Assume, on the contrary, there exists $\mathcal{B}_k^{+*}$, and $\mathcal{B}_k^{-*}$, $|\mathcal{B}_k^{+*}| = \theta|\mathcal{B}_k|$, $\mathcal{B}_k^{+*} \neq \{1, 2, ..., \theta|\mathcal{B}_k|\}$, $\mathcal{B}_k^{+*} \cup \mathcal{B}_k^{-*} = \mathcal{B}_k$, $\mathcal{B}_k^{+*} \cap \mathcal{B}_k^{-*} = \varnothing$, such that

$$\left(\sum_{i\in\mathcal{B}_k^{+*}} \delta_i - \sum_{i\in\mathcal{B}_k^{-*}} \delta_i\right) - \left(\sum_{i=1}^{\theta|\mathcal{B}_k|} \delta_i - \sum_{i=\theta|\mathcal{B}_k|+1}^{|\mathcal{B}_k|} \delta_i\right) > 0.$$

However, $\sum_{i\in\mathcal{B}_k^{+*}} \delta_i - \sum_{i=1}^{\theta|\mathcal{B}_k|} \delta_i \leq 0$, $\quad \sum_{i=\theta|\mathcal{B}_k|+1}^{|\mathcal{B}_k|} \delta_i - \sum_{i\in\mathcal{B}_k^{-*}} \delta_i \leq 0$, a contradiction. □

A.7 Proof of Proportion 13.1

Proof. Given example $(\mathbf{x}, y)$, we assume the example is misclassified as some category $z \neq y$, meaning that

$$\| \mathbf{A}_{y\cdot} - \mathbf{f}(\mathbf{x}) \| > \| \mathbf{A}_{z\cdot} - \mathbf{f}(\mathbf{x}) \| .$$

Then

$$\| \mathbf{A}_{y\cdot} - \mathbf{f}(\mathbf{x}) \| > \frac{\| \mathbf{A}_{y\cdot} - \mathbf{f}(\mathbf{x}) \| + \| \mathbf{A}_{z\cdot} - \mathbf{f}(\mathbf{x}) \|}{2}.$$

From triangle inequality and the definition of ρ:

$$\| \mathbf{A}_{y\cdot} - \mathbf{f}(\mathbf{x}) \| + \| \mathbf{A}_{z\cdot} - \mathbf{f}(\mathbf{x}) \| \geq \| \mathbf{A}_{y\cdot} - \mathbf{A}_{z\cdot} \| \geq \rho.$$

So we know misclassifying $(\mathbf{x}, y)$ implies that

$$\| \mathbf{A}_{y\cdot} - \mathbf{f}(\mathbf{x}) \| > \frac{\rho}{2}.$$

Therefore given N samples $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_N, y_N)$, the number of category recognition mistakes we make is at most

$$\frac{\sum_{i=1}^{N} \| \mathbf{A}_{y_i\cdot} - \mathbf{f}(\mathbf{x}_i) \|}{\rho/2} = \frac{2N\epsilon}{\rho}.$$

Thus the empirical error is upper bounded by $2\epsilon/\rho$. □

Appendix B

Publications

B.1 Circulant Projections

[234] F. X. Yu, S. Kumar, Y. Gong, and S.-F. Chang. Circulant binary embedding. In *Proceedings of the International Conference on Machine Learning*, 2014

[37] Y. Cheng, F. X. Yu, R. Feris, S. Kumar, A. Choudhary, and S.-F. Chang. Fast neural networks with circulant projections. *arXiv preprint arXiv:1502.03436*, 2015

[235] F. X. Yu, S. Kumar, H. Rowley, and S.-F. Chang. Compact nonlinear maps and circulant extensions. *arXiv preprint arXiv:1503.03893*, 2015

B.2 Learning from Label Proportions

[231] F. X. Yu, D. Liu, S. K., T. Jebara, and S.-F. Chang. $\propto$SVM for learning with label proportions. In *Proceedings of the International Conference on Machine Learning*, 2013

[233] F. X. Yu, K. Choromanski, S. Kumar, T. Jebara, and S.-F. Chang. On learning with label proportions. *arXiv preprint arXiv:1402.5902*, 2014

[118] K.-T. Lai, F. X. Yu, M.-S. Chen, and S.-F. Chang. Video event detection by inferring temporal instance labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014

[34] T. Chen, F. X. Yu, J. Chen, Y. Cui, Y.-Y. Chen, and S.-F. Chang. Object-based visual sentiment concept analysis and application. In *Proceedings of the ACM Multimedia*

Conference, 2014

[232] F. X. Yu, L. Cao, M. Merler, T. Chen, J. Smith, and S.-F. Chang. Modeling attributes from category-attribute proportions. In *Proceedings of the ACM Multimedia Conference*, 2014

B.3 Attribute-Based Image Retrieval and Recognition

[229] F. X. Yu, R. Ji, M.-H. Tsai, G. Ye, and S.-F. Chang. Weak attributes for large-scale image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012

[230] F. X. Yu, L. Cao, R. Feris, J. Smith, and S.-F. Chang. Designing category-level attributes for discriminative visual recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013

B.4 Mobile Visual Search

[228] F. X. Yu, R. Ji, and S.-F. Chang. Active query sensing for mobile location search. In *Proceedings of the ACM Multimedia Conference*, 2011

B.5 Video Event Recognition

[17] S. Bhattacharya, F. X. Yu, and S.-F. Chang. Minimally needed evidence for complex event recognition in unconstrained videos. In *Proceedings of the Annual ACM International Conference on Multimedia Retrieval*, 2014