Towards Robust Semantic Attribute Learning in Visual Computing

by

Lin Chen

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved April 2016 by the
Graduate Supervisory Committee:

Baoxin Li, Chair
Pavan Turaga
Yalin Wang
Huan Liu

ARIZONA STATE UNIVERSITY

May 2016

ABSTRACT

The rapid growth of social media in recent years provides a large amount of user-generated visual objects, e.g., images and videos. Advanced semantic understanding approaches on such visual objects are desired to better serve applications such as human-machine interaction, image retrieval, etc. Semantic visual attributes have been proposed and utilized in multiple visual computing tasks to bridge the so-called "semantic gap" between extractable low-level feature representations and high-level semantic understanding of the visual objects.

Despite years of research, there are still some unsolved problems on semantic attribute learning. First, real-world applications usually involve hundreds of attributes which requires great effort to acquire sufficient amount of labeled data for model learning. Second, existing attribute learning work for visual objects focuses primarily on images, with semantic analysis on videos left largely unexplored.

In this dissertation I conduct innovative research and propose novel approaches to tackling the aforementioned problems. In particular, I propose robust and accurate learning frameworks on both attribute ranking and prediction by exploring the correlation among multiple attributes and utilizing various types of label information. Furthermore, I propose a video-based skill coaching framework by extending attribute learning to the video domain for robust motion skill analysis. Experiments on various types of applications and datasets and comparisons with multiple state-of-the-art baseline approaches confirm that my proposed approaches can achieve significant performance improvements for the general attribute learning problem.

I dedicate my dissertation work to my parents, Ping Chen and Kailan Tang. Their love is the source of my power moving me forward!

ACKNOWLEDGMENTS

This dissertation is impossible without the help from my advisor Dr. Baoxin Li. I really appreciate his patience and excellent advising skills during guiding my Ph.D. research. His passion and responsibility on research always inspire and encourage me to discern and analyze the essentials of challenge research problems. Dr. Li is more of a mentor and friend for life than an advisor for research. He gives me a lot of valuable suggestions on my career and my life in United States. I really enjoy sharing my personal thoughts with him. I can not thank him more for making my Ph.D. experience becoming an exciting journey!

I would like to thank my committee members, Dr. Yalin Wang, Dr. Huan Liu and Dr. Pavan Turaga, for helpful suggestions and insightful comments. I attended Shape Analysis for Computer Vision and Graphics class with Dr. Yalin Wang and data mining class with Dr. Huan Liu, which lies a solid basis for my research. Dr. Yalin Wang's insightful mathematical analysis on Computer Vision greatly inspires me on my research to tackle challenge problems. I am impressed by Dr. Huan Liu's ability to make complicated data mining techniques extremely easy to understand and applicable, which makes applying those techniques on my research really enjoyable. As an excellent researcher on Computer Vision, Dr. Pavan Turage's research and comments broaden my perspectives on my research.

I was lucky to work as an intern in Vision group of Nokia Technologies. I really appreciate the opportunity working with my amazing mentors and colleagues: Timo Ahonen, Vaquero Daniel, Basavaraja Vandrotti, Hui Zhou, Hoseok Chang and Muninder Valdandi. I will never forget the joyful summer working in this diversity and friendly group and accomplishing an exciting research project. Their guidance guarantees me my first patent in my lift as a main inventor.

During my Ph.D. study, my friends and colleagues provided me consistent support and encouragement and they deserve a special thank. I am thankful to my colleagues at Visual Representation and Processing Group: Qiang Zhang, Gazi Islam, Hima Bindu, Nan Li, Peng Zhang, Jun Cao, Parag Chandakkar, Devi Archana Paladugu, Qiongjie Tian, Ragav Venkatesan, Yilin Wang, Xu Zhou, Yikang Li, Yuzhen Ding, Jaya Gattupalli, Pak Lun Ding, Jashmi Lagisetty, Sheng-Hung Hu, Zhigang Tu. Sharing ideas and experience not only on research but also on life make our experience in ASU full of joy. I am also lucky to have my friends in my life Jan Lows Clay, Karen Rice, Dick and Suzanne Bither, Joyce Lemons, Bob Thompson and Gary Boyd who treat me as their family members and offer me great help when I am suffering from illness.

Finally, My heartfelt appreciations go to my dear mother Kailan Tang and father Ping Chen for their love and strong support during my graduate study. All of my achievements honors in my life belong to them.

TABLE OF CONTENTS

Page

APPENDIX

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

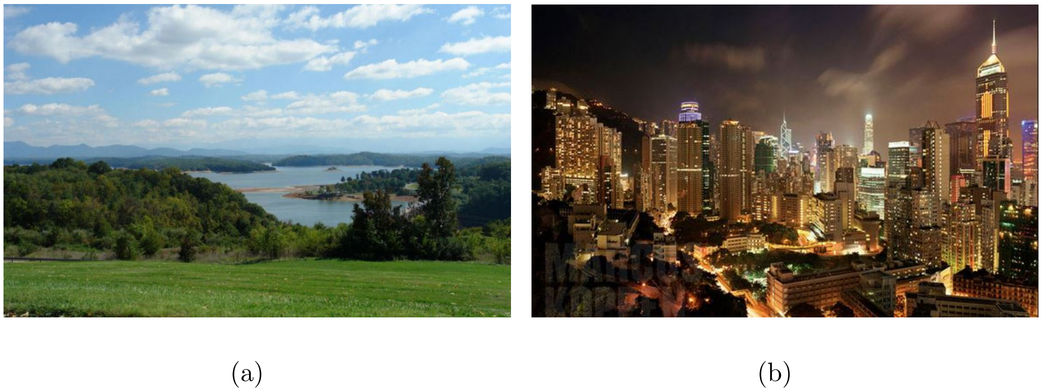
Figure Page

Chapter 1

INTRODUCTION

The social media era has witnessed phenomenal growth of user-generated visual objects, e.g., images and videos, on the Internet. The ever-growing number of visual objects has brought about new challenges for efficient semantic understanding, and in turn, for applications that rely on semantic understanding. The semantic gap, which refers to the discrepancy between extractable low-level feature representation and high-level semantic understanding of visual objects, still exists in recent visual computing applications. In recent years, towards bridging the semantic gap, methods exploiting semantic attributes of visual objects have attracted significant attention. Instead of using low-level features, these approaches describe images by high-level, human-nameable visual attributes including both holistic descriptors, such as "natural" scenes, "fluffy" dogs, as well as localized parts, such as car "has-wheels", bird "has-wings". Such semantic visual attributes exist across object category boundaries, thus can be utilized as human-nameable features [51] or general descriptors [31] to support learnt knowledge transferring/generalization [54] in many visual computing tasks including object recognition [31, 30], face verification [85] and image search [79].

Recent research on semantic attributes mainly focuses on two tracks. The first one is relative attribute ranking. Given pairs of object attributes describing the relative strength (e.g., in Figure 1.1, Figure 1.1(a) is more natural than Figure 1.1(b)), relative attribute ranking aims to learn a ranking function to rank the visual objects according to the strength of the attributes. The other one is attribute prediction. Given a set of visual objects with labels identifying whether the attribute is associated with the object (e.g., Figure 1.1(a) is natural and Figure 1.1(b) is not natural), attribute



(a) (b)

Figure 1.1: An example of semantic attributes. In binary attribute prediction, (a) is labeled as "natural" and (b) is labeled as "man-made". In relative attribute ranking, (a) is more "natural" than (b).

prediction learns a binary predictor that aims to indicate whether or not a visual property is present in the visual object.

Despite of years of research efforts on this regard, there are still some difficult problems in need of better solutions. For example, the availability of very limited attribute labels becomes a bottleneck for learning a well-generalized model. To make the matter worse, recent applications usually involve a large number of attributes. For example, the **Caltech-UCSD Birds-200-2011** dataset [91] involves 312 attributes per image. Many attributes, e.g., "has-hooked-bill-shape" or "has-spotted-tail", are very sparsely labeled. Such scarcely-labeled data make it very difficult to learn an accurate attribute model either for prediction or for ranking. Also, existing research on visual semantic attributes mainly focuses on the image domain. It remains to be explored how the success of semantic attribute learning in the image domain may be extended to challenging video analysis tasks such as motion skill analysis, where semantics may be more difficult to model.

In this dissertation, I present several new solutions to tackle the aforementioned problems in attribute learning. First, I investigate both the attribute ranking and prediction problems in order to learn a reliable and accurate semantic attribute model, given very limited training samples, for various types of data. Then I conduct the research on how to extend the semantic learning approaches to the video domain for robust motion skill analysis. Specifically, my contribution can be summarized in the following several aspects.

Many practical applications involve multiple attributes. For a given problem, such multiple attributes usually exhibit correlation among themselves. For example, as shows in Figure 1.1, a “natural” scene (Figure 1.1(a)) is usually observed as “open” as well. Capturing such correlation would be benefitial for effective attribute learning especially when the available training data is very limited. To exploit the potential correlation among multiple attributes, I propose several new frameworks, for both relative attribute ranking and binary attribute prediction, employing multi-task learning (MTL) to capture such attribute correlation for improved attribute learning.

The problem of relative attribute ranking and binary attribute prediction involve different types of data for learning. Relative attribute ranking takes pairwise labels (e.g., Figure 1.1(a) is more “natural” than Figure 1.1(b)), while binary attribute prediction takes pointwise labels (e.g., Figure 1.1(a) is “natural” and Figure 1.1(b) is not “natural (man-made)”). Pairwise labels and pointwise labels have different advantages and limitations in terms of data availability, labeling complexity and representational capability. To alleviate the problem of lacking labelled training samples, I propose a hybrid framework fusing pointwise and pairwise labels to maximize the utilization of all available training data for improved attribute learning.

Video-based coaching systems have seen increasing adoption in various applications including dance, sports, and surgery training. However, how to automatically analyze human motion skills and provide semantically-meaningful feedback is still a practical challenge. Inspired by the success of image understanding through semantic attributes, I introduce semantic attribute learning into the video domain for high-level skill understanding. By integrating a suite of vision-based techniques, I present a video-based skill coaching system for simulation-based surgical training by exploring a newly-proposed problem of instructive video retrieval. The proposed framework provides automatic skill analysis feedback, as well as recommends an instructive video for self-improvement.

The remainder of this dissertation is organized as follows. In Chapter 2, I introduce some basic concepts and technical fundamentals related to my research, e.g., learning-to-rank, multi-task learning, etc. In Chapter 3, I present how to utilize the correlation among attributes for improved relative attribute ranking. In Chapter 4, I further investigate the specific correlation structures existent among attributes and propose a advanced framework for binary attribute prediction. In Chapter 5, I present a framework maximizing the utilization of all available data by fusing both pointwise and pairwise labels for robust attribute learning. In Chapter 6, I extend the attribute learning on video domain and propose a video-based skill coaching system through semantic skill attribute analysis. I conclude the dissertation and point out potential feature research directions in Chapter 7.

Chapter 2

FOUNDATIONS AND PRELIMINARIES

In this chapter I mainly discuss some technical foundations and preliminaries involved in this dissertation, i.e., multi-task learning and relative learning.

2.1 Multi-task Learning

Multi-task learning aims to improve generalization performance by training several tasks together to capture their intrinsic correlation. Various types of multi-task learning approaches and applications have been proposed. Neural network approaches [8, 17, 84] utilized a hidden layer with a few nodes and a set of network weights shared by all tasks. Hierarchical Bayes approach [7, 94, 95, 97] enforced task relatedness through a common prior probability distribution on the tasks' parameters.

In recent years, more attention has been paid to regularization-based multi-task learning, which is what I mainly considered in this work. The general form of regularization-based multi-task learning is:

$$\min_{W} \Big(\sum_{t=1}^{t'} \sum_{i=1}^{n} (y_{ti} - W_t^T x_{ti})^2 + \lambda\Omega(W) \Big) \tag{2.1}$$

where t denotes the t-th task and i denotes the i-th sample in task t. Much work has been proposed, often introducing different cost functions and regularization terms.

Evgeniou and Pontil [63] assumed that the projection vectors of all tasks are close to each other and proposed the regularization term using a shared mean vector $\boldsymbol{w}_0$ and a small perturbation vector $\boldsymbol{v}_t$ to represent the projection vector of the t-th task $\boldsymbol{w}_t = \boldsymbol{w}_0 + \boldsymbol{v}_t$. This idea is intuitive and easy to implement, but the assumption is too strong to hold in real applications. [3] proposed Alternating Structure Optimization

(ASO) based on a similar assumption that the projection model is the sum of a task specific component and a shared low dimensional subspace.

Processing high-dimensional feature datasets attracts a lot of research interests. Considering the task relatedness that different task models share a common set of features, [40] and [60] introduced ℓ_1/ℓ_q-norm group lasso penalty as regularization to obtain a sparse projection matrix for feature selection. Ji and Ye [44] introduced trace norm as regularization and obtained a low-rank structure projection matrix to capture task relatedness. These approaches make the strong assumption that all tasks are related.

Considering the existence of outlier tasks, Jalali [42] and [35] introduced an extra ℓ_1 and ℓ_1/ℓ_q-norm regularization term individually into feature selection; [18] introduced an extra ℓ_1/ℓ_q-norm regularization term into low-rank subspace learning. These approaches learn a projection matrix as well as detect the outlier tasks.

Other multi-task learning approaches include assumptions that tasks have some special structure. For example, in [5, 98], tasks in the same group are closer to each other than tasks in a different group; in [47], tasks from the same node are closer to each other and relatedness among the nodes depends on the depth in a tree; in [20], task relatedness depends on the edge weight between the two tasks in a graph representation.

Recent multi-task learning approaches in literature are in general for classification, and there is little work on extending them for ranking applications.

2.2 Attribute Prediction

A visual attribute learner is a binary predictor that aims to indicate whether or not a visual property is present. The standard approaches learn the attribute predictor independently per attribute. [32] presented a probabilistic generative model which learns attributes by distinguishing unary property of single segment or patterns of alternating segments. [54] considered zero-shot learning where the test set consists entirely previously unseen object categories and the information is transferred from the training set to the test phase entirely through the attribute labels. [31] described unfamiliar objects and new categories by visual attribute of object parts, e.g., “has head”, or appearance adjectives, e.g., “spotty”. [30] first learned part and category detectors of objects and then described objects by spacial arrangement of the attributes and their interactions. [51] and [79] used attributes to facilitate human-machine interaction for image search by which the user is able to specify precise semantic queries.

While most methods learn attributes independently, some initial steps have been taken towards modeling attribute relationships. [93] treated attributes as latent variables and capture the correlations among attributes using an undirected graphical model built from training data. [85] proposed a method to model the attribute relationship for face verification based on a discriminative distributed-representation for attribute description. [83] proposed retrieval approach where correlations of attributes are considered as multi-attributes query in the vocabulary.

Considering utilizing multi-task learning framework for attribute learning, [19] proposed a ranking framework which learns a common feature space among all attributes while detect outliers; [43] aimed to select appropriate subset of features for different attributes by manually dividing attributes into different semantic groups and encourage intra group feature sharing and inter group feature competition.

These approaches either make the strong assumption that all tasks are correlated or require human intervention effort to specify appropriate semantic groups. In contrast, my approach can automatically detect semantic groups and learns an effective subset of features representing the attributes.

2.3 Relative Attributes Learning

Relative attribute learning is a fairly recent concept, which has drawn increasing attention. Relative attributes were first used by Parikh and Grauman [67] to learn a ranking function for each human-nameable attribute of an image. The relative “strength” of an attribute is measured by some distance metrics learned through SVM-like optimization using (relatively) labeled pairs. Relative attribute learning is applicable to zero-shot learning (detecting ‘unseen’ category) and image description in relative terms.

Parkash and Parikh [69] incorporated attribute feedback into the classification process. Employing attributes as the communication “language” between the human supervisor and the machine learner, their work allows supervisors to provide feedback to the learner for improved learning. [51] presented a feedback scheme for image search. Based on pre-trained relative attribute ranking functions, their system demonstrates an initial set of queried results and asks the user to provide relative attribute feedback. The system then updates the training set based on the feedback and provides new queried images utilizing newly trained relative attribute ranking functions.

Most of current relative attribute learning approaches only consider ranking attributes independently. The proposed work attempts to explicitly model potential correlation among the attributes of interest so as to achieve better ranking performance, especially when limited training data are available (and thus each individual attribute may have even fewer labeled pairs of training samples).

2.4 Notations

In this dissertation, I represent scalars, vectors, matrices and sets as lower case letters x, bold face lower case letters $\boldsymbol{x}$, capital letters X and calligraphic capital letters $\mathcal{O}$ respectively. $\boldsymbol{x}_i$ denotes the i-th column of the matrix X. $\|\cdot\|$ and $\|\cdot\|_F$ represent Euclidean and Frobenius norms respectively. $\|X\|_{p,q}$ is defined as the $\ell_{p,q}$ norm $(\sum_i((\sum_j x_{ij}^q)^{\frac{1}{q}})^p)^{\frac{1}{p}}$. $\mathbf{Tr}(X)$ represents the trace of X and $\|X\|_* = \sum_{i=1}^r \sigma_i(X)$ is the trace norm, with $r = rank(X)$ and $\sigma_i(X)$ the i-th non-zero singular value in non-increasing order.

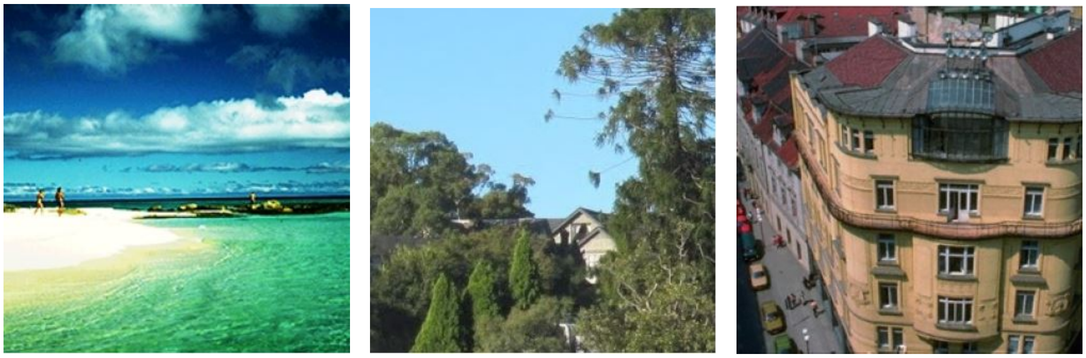
Chapter 3

ATTRIBUTE RANKING BASED ON PAIRWISE LABEL

In this chapter, I mainly discuss the work proposed that utilizes the relatedness among different attributes for attribute ranking.

3.1 Introduction

Recent literature has witnessed fast development of the new methodology of relative attribute learning, whose goal is to overcome the limitation of traditional learning approaches based only on binary labels. In general, a traditional learning approach using binary labels can only map low-level features to one of the two labels, without capturing the "relativeness" of the concepts that the labels are supposed to represent. For example, in Figure 3.1, we may see that 3.1(a) is "natural" and 3.1(c) is "man-made", but we may be less certain on assigning either of the labels to 3.1(b). Unlike learning with binary labels, relative attributes learning is to capture the strength of



(a) (b) (c)

Figure 3.1: An example of relative attributes. Considering binary label learning, (a) is labeled as "natural" and (c) is labeled as "man-made", however, it is hard to lable (b) as "natural" or "man-made." In relative attributes, (a) is more "natural" and "open" than (b) and (b) is more "natural" and "open" than (c).

the attributes under consideration. For example, this would allow us to say 3.1(b) is less “natural” but more “man-made” than 3.1(a) while being more “natural” but less “man-made” than 3.1(c).

Many practical applications involve multiple attributes (like the two concepts, “natural” and “man-made”, in the above image labeling example). Current relative attributes learning approaches train separate ranking functions independently for each of the attributes under consideration. For a given problem, if multiple attributes are involved, they usually exhibit correlation among them. For example, the more “natural” a scene image is, the more “open” it may be, where two attributes “being natural” and “being open” often have positive correlation. In addition, even if two attributes are disjointed in the high-level semantic space, in a practical algorithm they may be dependent of some common low-level features, and thus are made to be related to each other in some sense. Both factors suggest that the correlation among different attributes of the same problem should be dealt with in a principled way for effective relative attributes learning.

To exploit potential correlation among multiple attributes for learning better ranking functions, in this chapter, I employ multi-task learning in relative attributes learning and propose a new multi-attribute relative learning framework. Multi-task learning learns several tasks simultaneously for potential performance gain through utilizing “relatedness” among different tasks, which provides a principled way for us to model correlation among attributes, if we view the attributes as tasks. In the proposed framework, a new cost function is defined to capture the joint effect of the individual objective functions in original relative attribute learning. Further, assuming different types of correlation exist among attributes, different regularizations are introduced to model the potential correlation among the attributes. As a result, the proposed framework could learn the relative strength of the attributes simultaneously while utilizing the correlation among the attributes/tasks. Under this framework, I develop corresponding optimization algorithms employing Block Coordinate Descent principles. Our algorithm solves the learning problem through alternating optimization steps dealing with capturing the relative ranking information and the attribute correlation information iteratively.

The key contribution of the work in this Chapter lies in a novel formulation of relative attributes learning that handles multiple attributes jointly to capture the potential correlation among them for improved learning performance. Additionally, an algorithm is developed to find a solution under the formulation. As demonstrated by our experiments, the proposed method is able to deliver good performance even with a small number of training pairs, owing to its ability to exploit correlation among the attributes.

In the remaining of this Chapter, I first introduce a joint feature selection framework for multi-attribute relative learning, based on the assumption that the attributes are correlated by sharing the same subset of features, in Section 3.2. The I discuss another framework for learning a low-rank latent space, in which the related attributes are linear correlated, for relative attribute ranking.

3.2 Joint Feature Selection

Assuming the correlation among different attributes exist as related attributes share the same subset of feature representation, I first propose a joint feature selection framework for multiple attribute ranking.

3.2.1 Proposed Approach

With the reasonable assumption that multiple attributes describing the same object should be related in some way and that only relatively-labelled data pairs are given, I propose to jointly learn multi-attribute ranking functions in the following general formulation of an optimization problem:

$$\begin{aligned}
&\min_{W,\rho_1,\rho_2,\lambda} \sum_{t=1}^{t'} (\frac{1}{2}\left\|\boldsymbol{w}_t\right\|^2 + \rho_1 \sum_{i,j\in\mathcal{O}} \xi_{ijt} + \rho_2 \sum_{i,j\in\mathcal{S}_t} \gamma_{ijt}) + \mu\Omega(W) \\
&s.t. \;\; \boldsymbol{w}_t^\top(\boldsymbol{x}_i - \boldsymbol{x}_j) \geq 1 - \xi_{ijt}; \;\; \forall(i,j)\in\mathcal{O}_t; \\
&\qquad \left|\boldsymbol{w}_t^\top(\boldsymbol{x}_i - \boldsymbol{x}_j)\right| \leq \gamma_{ijt}; \;\; \forall(i,j)\in\mathcal{S}_t; \\
&\qquad \xi_{ijt} \geq 0; \;\; \gamma_{ijt} \geq 0; \;\; t = 1,2,...,t'.
\end{aligned} \tag{3.1}$$

In this formulation, W is the projection matrix with the t-th column $\boldsymbol{w}_t$ as the projection vector for the t-th attribute (task), $\Omega(W)$ is a regularization term, $\boldsymbol{x}_i$ is the feature vector of the i-th sample, $\mathcal{O}_t = \{(i,j)\}$ is the set of ordered pairs (i,j) satisfying $\boldsymbol{w}_i^\top\boldsymbol{x}_i > \boldsymbol{w}_j^\top\boldsymbol{x}_j$, $\mathcal{S}_t$ is the set of similar pairs (i,j) satisfying $\boldsymbol{w}_i^\top\boldsymbol{x}_i \approx \boldsymbol{w}_j^\top\boldsymbol{x}_j$, ρ_1, ρ_2 and μ are trade-off constants, ξ_{ijt} and γ_{ijt} are slack variables measuring the error of the distance of prior and similar pairs. By applying appropriate regularization terms, the attribute projection model W is learned simultaneously.

In this study, we adopted the same regularization scheme as in [35] which effectively achieves joint feature learning based on the assumption that the same set of essential features may be shared across different attributes with existence of outlier tasks. This results in the following specialized problem

$$\begin{aligned}
&\min_{W,\rho_1,\rho_2,\lambda} \sum_{t=1}^{t'} (\frac{1}{2}\left\|\boldsymbol{w}_t\right\|^2 + \rho_1 \sum_{i,j\in\mathcal{O}} \xi_{ijt} + \rho_2 \sum_{i,j\in\mathcal{S}_t} \gamma_{ijt}) + \mu_1\|P\|_{1,2} + \mu_2\left\|Q^\top\right\|_{1,2} \\
&s.t. \;\; \boldsymbol{w}_t^\top(\boldsymbol{x}_i - \boldsymbol{x}_j) \geq 1 - \xi_{ijt}; \;\; \forall(i,j)\in\mathcal{O}_t; \\
&\qquad \left|\boldsymbol{w}_t^\top(\boldsymbol{x}_i - \boldsymbol{x}_j)\right| \leq \gamma_{ijt}; \;\; \forall(i,j)\in\mathcal{S}_t; \\
&\qquad W = P + Q; \;\; \xi_{ijt} \geq 0; \;\; \gamma_{ijt} \geq 0; \;\; t = 1,2,...,t'.
\end{aligned} \tag{3.2}$$

where the first regularization term enforces a group Lasso penalty on row groups of P in order to capture the shared features among the attributes. The second term enforces the same group Lasso penalty, but on column groups of Q to discover the outlier tasks.

3.2.2 Algorithm

I now turn to the problem of finding an solution under the proposed formulation. Without loss of generality, our following discussion is in terms of a general regularization term $\Omega(W)$. In general, solving the constrained optimization problem of Equ. 3.2 is difficult especially since common multi-task regularization terms are typically non-differentiable. We propose an algorithm based on Block Coordinate Descent (BCD) principles. In this approach, we introduce a slack variable $\tilde{W}$ which is similar to W so that the original problem may be solved by two alternating processes, focusing on a new cost function and the regularization term respectively. That is, we first convert the original problem into

$$\min_{W,\tilde{W},\rho_1,\rho_2,\lambda} \sum_{t=1}^{t'} (\frac{1}{2} \|\boldsymbol{w}_t\|^2 + \rho_1 \sum_{i,j \in \mathcal{O}} \xi_{ijt} + \rho_2 \sum_{i,j \in \mathcal{S}_t} \gamma_{ijt}) + \lambda \left\| W - \tilde{W} \right\| + \mu \Omega(\tilde{W}) \tag{3.3}$$

in which the norm $\left\| W - \tilde{W} \right\|$ enforces a similar solution of W and $\tilde{W}$.

The above optimization problem can be solved by iteratively updating W and $\tilde{W}$ in the following two separate problems:

Optimization of W For a fixed $\tilde{W}$, the optimal W can be obtained via solving:

$$\begin{aligned} \min_{W,\rho_1,\rho_2,\lambda} & \sum_{t=1}^{t'} (\frac{1}{2} \|\boldsymbol{w}_t\|^2 + \rho_1 \sum_{i,j \in \mathcal{O}_t} \xi_{ijt} + \rho_2 \sum_{i,j \in \mathcal{S}_t} \gamma_{ijt}) + \frac{\lambda}{2} \left\| W - \tilde{W} \right\|_F^2 \\ s.t. \ \ & \boldsymbol{w}_t^T (\boldsymbol{x}_i - \boldsymbol{x}_j) \geq 1 - \xi_{ijt}; \ \ \forall (i,j) \in \mathcal{O}_t; \\ & \left| \boldsymbol{w}_t^T (\boldsymbol{x}_i - \boldsymbol{x}_j) \right| \leq \gamma_{ijt}; \ \ \forall (i,j) \in \mathcal{S}_t; \\ & \xi_{ijt} \geq 0; \ \ \gamma_{ijt} \geq 0; \ \ t = 1, 2, ... t'. \end{aligned} \tag{3.4}$$

where we used the Frobenius norm on $\left\| W - \tilde{W} \right\|$ for facilitating the solution. This problem focuses on learning all attributes together by encoding multi-attribute in-

Algorithm 1 Alternating Optimization

```
Input:
    Data feature set $X$, training ranking pairs set $E$ (prior) and $F$ (similar), parameters $\rho_1$, $\rho_2$, $\lambda$, $\tilde{\lambda}$, $\mu$;
Output:
    Projection matrix $W$;
Initiate $\tilde{W}$ as random matrix, $W$ as zero matrix, $\lambda = 0.05\tilde{\lambda}$;;
while $\left\|W - \tilde{W}\right\|_F^2 > 10^{-10}$ do
    Optimize function (3.4), update matrix $W$;
    Optimize function (3.5), update matrix $\tilde{W}$;
    Set $\lambda = \lambda + 0.05\tilde{\lambda}$;
end while
return $W$;
```

formation into one quadratic optimization process. The second term enforces the projection weight matrix W to be close to the given "multi-task" weight matrix $\tilde{W}$.

Optimization of $\tilde{W}$ For a fixed W, the optimal $\tilde{W}$ can be obtained via solving:

$$\min_{\tilde{W}} \left\|(\tilde{W}^\top - W^\top)X\right\| + \mu\Omega(\tilde{W}) \tag{3.5}$$

This problem enforces a joint learning regularization constraints $\Omega(\tilde{W})$ to the projection weight matrix to capture the correlation among the attributes. The first term penalizes the difference to make sure the learned "multi-task" weight matrix $\tilde{W}$ is close to the given projection weight W. The overall optimization algorithm is summarized in Algorithm 1.

In implementation, the first problem given in (3.4) can by solved by first converting it to its dual form problem, which is a typical quadratic optimization problem. While interested readers may find the derivation in the supplemental material, we list the dual form below for completeness:

$$\begin{aligned}
&\min_{a_{st},d_{st}} \left(\frac{1}{2}\boldsymbol{x}^T Y^T Y \boldsymbol{x} + (\lambda Y^\top \tilde{\boldsymbol{w}} - (1+\lambda)\boldsymbol{e})^T \boldsymbol{x}\right) \\
&s.t. \quad A\boldsymbol{x} \le \boldsymbol{b} \\
&with \quad \boldsymbol{x} = (\boldsymbol{x}_1; \boldsymbol{x}_2; \cdots; \boldsymbol{x}_t)^\top; \\
&\qquad Y = \Sigma(Y_t); \\
&\qquad \tilde{\boldsymbol{w}} = [\tilde{\boldsymbol{w}}_1; \tilde{\boldsymbol{w}}_2; \cdots; \tilde{\boldsymbol{w}}_t]; \\
&\qquad \boldsymbol{e} = [\boldsymbol{e}_1; \boldsymbol{e}_2; \cdots; \boldsymbol{e}_t]; \\
&\qquad A = [E_{|Y|\times|Y|}; -E_{|Y|\times|Y|}]; \\
&\qquad \boldsymbol{b}_t = [\underbrace{\rho_1, \rho_1, \cdots, \rho_1}_{|\mathcal{O}_t|}, \underbrace{\rho_2, \rho_2, \cdots, \rho_2}_{|\mathcal{S}_t|}]^\top. \\
&\qquad \tilde{\boldsymbol{b}}_t = [\underbrace{0, 0, \cdots, 0}_{|\mathcal{O}_t|}, \underbrace{\rho_2, \rho_2, \cdots, \rho_2}_{|\mathcal{S}_t|}]^\top. \\
&\qquad \boldsymbol{b} = [\boldsymbol{b}_1, \boldsymbol{b}_2, \cdots, \boldsymbol{b}_t, \tilde{\boldsymbol{b}}_1, \tilde{\boldsymbol{b}}_2, \cdots, \tilde{\boldsymbol{b}}_t]; \\
&\qquad t = 1, 2, \cdots, t'.
\end{aligned} \tag{3.6}$$

In essence, the problem of (3.4) is similar to regular relative attribute learning, and the problem of (3.5) is similar to multi-task learning, and thus there convergence behavior is well-understood. In our implementation, to facilitate convergence, we set a small value for λ in Equation (3.4) at the beginning. Then in each iteration afterwards, we increase λ gradually until it reaches a specified large threshold. Therefore, the weight of the second term becomes larger and larger which ensures the cost $\left\|W - \tilde{W}\right\|_F^2$ would decrease after each iteration. The algorithm terminates when $W \approx \tilde{W}$ is reached.

The selection of parameters ρ_1, ρ_2, λ and μ is made problem-dependent through cross-validation. Specifically, we first find a suitable parameter search space by binary search or subgradient approach. For example, μ can be searched in a space ranging from achieving a desired minimal sparsity to a maximal sparsity. Then we adjust the parameters one by one while fixing the other parameters according to the performance of cross-validation.

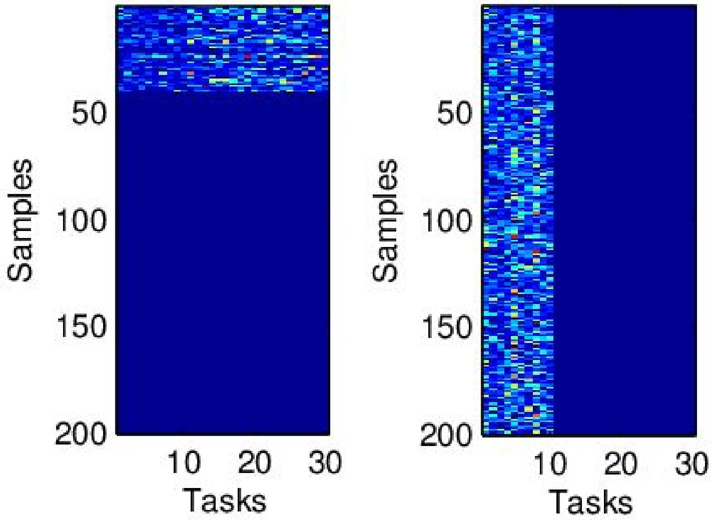
3.2.3 Experiments

In this section, we first experimented on synthetic dataset to show how well the correlation among the attributes are captured in our new proposed attribute learning framework. Then we test the framework on two real datasets on both the ranking accuracy of learned ranking function and classification accuracy of zero-shot learning.

Experiments with Synthetic Data

In order to test whether our framework can capture the relatedness among the attributes, we construct the synthetic datasets in the following way. The total attribute (task) number is $t = 30$. For the i-th attribute, we generate the data set $X_i \in \mathcal{R}^{d\times n}$ containing $n = 200$ samples and d dimensions for each sample. Each entry of X_i is drawn from the normal distribution $N(0, 25)$. The groundtruth projection matrices $P \in \mathcal{R}^{d\times n}$ and $Q \in \mathcal{R}^{d\times n}$ are drawn from $N(0, 64)$. We set the first 10 columns of Q non-zero and they indicate outlier tasks. We also draw a noise vector $\boldsymbol{\delta}_i \in R^n$ from $N(0, 1)$. Thus, the final ranking score for data set X_i is computed as $\boldsymbol{y}_i = X_i^T(P + Q) + \boldsymbol{\delta}_i$.

I run the experiments 4 rounds with the feature dimension d increasing from 50 to 200 with step size 50. In the first round, all 50 dimensions are set as shared intrisic features, which means all 50 rows of P are set non-zeros. Then 50 more zero rows are added into Q in each round afterwards till d reaches to 200. In this setup, the first 50 dimensions of feature (first 50 rows of P) represent the selected joint features among the attributes.



(a) P (b) Q

Figure 3.2: Projection matrices P (a) and Q (b) learned by our framework on synthetic data. Blue color represents zero entry while other colors represent non-zero entry. Results show the first 50 rows of P are selected as selected shared features and the first 10 columns of Q are detected as outlier tasks.

Through cross validation, during each round of our experiment the best ranking performance is always achieved while the first 50 dimensions are selected as joint features (the first 50 rows of learned projection matrix P are non-zeros) and the first 10 attributes are detected as outliers (the first 10 columns of learned projection matrix Q are non-zeros). Figure 3.2 demonstrates the learned projection matrices P and Q when d reaches 200 as the parameters are set as $\mu_1 = 9.3$, $\mu_2 = 20.7$, $\rho_1 = \rho_2 = 300$, and $\tilde{\lambda} = 500$. The result shows that when $d = 200$, the first 50 rows of P are selected as the joint features and the first 10 columns of Q are detected as outlier attributes, which are all non-zeros. This result matches the groundtruth we have constructed previously, which suggests that our approach is able to capture the inherent relatedness of the projection model.

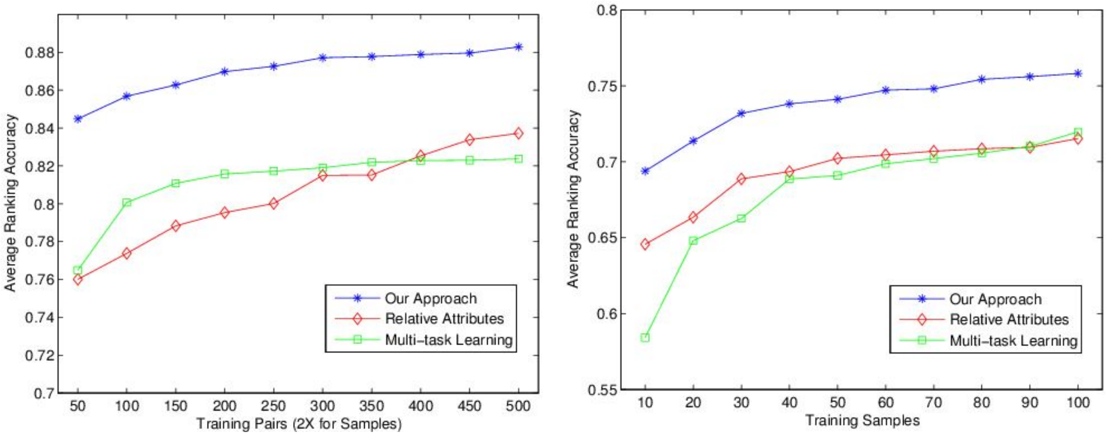
Experiments with Real Data

I compare our framework with the baseline methods on two datasets: OSR dataset includes 2688 color outdoor scene images from 8 categories with 6 attributes [67]. Shoes dataset includes 14765 images collected from like.com containing 10 categories of shoes with 10 attributes [51].

I compare our framework with two alternative approaches. The first approach is relative attribute [68] which learns a ranking function for each attribute independently. The second approach is based on multi-task learning work [35, 18], by which we trained classifiers and used the classification score to rank the attributes. We tested both the ranking accuracy of learned ranking function and classification accuracy of zero-shot learning in the experiments.

The average ranking accuracy is reported by running 5 rounds of each implemented approach. By cross validation, parameters of our framework are set as $\mu_1 = 60$, $\mu_2 = 20$, $\rho_1 = \rho_2 = 300$, $\tilde{\lambda} = 400$ on OSR during which the projection matrix is learned after 17 iterations. On Shoes, parameters are set as $\mu_1 = 3$, $\mu_2 = 50$, $\rho_1 = \rho_2 = 300$, $\tilde{\lambda} = 500$ and the projection model is got through 15 iterations. For the baseline relative attributes approach, we adopted the same parameter setup which is reported in [68] as the optimal parameters.

I first experimented the approaches on OSR dataset. Labeled training pairs are randomly left out for each attribute. The number of training pairs of each attribute increased from 50 to 500 with step size 50. For the baseline multi-task classification approach, we left 100 to 1000 training samples out for comparison. Since n training pairs would select at most $2n$ training samples, the training set left for multi-task classification gains no less information than the other two ranking approaches. Figure 3.3(a) illustrates the average ranking accuracies as a function of increased number of



(a) OSR

(b) Shoes

Figure 3.3: Average ranking accuracy of OSR and Shoes datasets as the increased number of training pairs and samples. Our framework (blue) outperforms the compared approaches by more than 5% (450 pairs) to 11% (50 pairs) on OSR and by more than 4% (100 samples) to 5% (10 samples) on Shoes.

training pairs. The result show that the accuracies of all three approaches increase with growing size of training data. The accuracy achieved by our framework (blue curve) outperforms the baseline results by 5%∼11%. The best performance gain is achieved when the number of training pairs gets to 50. Table 3.1 details the ranking accuracies of all 6 attributes on OSR when the number of training pairs is 50. According to the result, other than "Depth-cloth", accuracies of all attributes achieved by our framework are obviously higher than the competing results and the best performance gain is 18% in attribute "natural".

The implemented approaches are then tested on Shoes in which a different training sets selection scheme is applied. Instead of leaving training pairs out, we left some training samples out (ranging from 10 to 100 in number), and the training pairs are selected merely from the left training set. Figure 3.3(b) depicts the average ranking accuracies as a function of the size of training data. This experiment shows similarly that our proposed framework (blue curve) outperforms the other approaches by 4%∼5%. The highest performance gain is got when the number of training samples

Table 3.1: Ranking accuracies of each attribute on OSR when the number of training pairs are 50 of each attribute for our approach and relative attributes, 100 samples of each attribute for multi-task learning.

Attribute Name	Our Approach	Relative Attributes	Multi-task Learning
Natural	**90.42**%	72.90%	85.33%
Open	**88.62**%	83.18%	79.44%
Perspective	**83.64**%	77.67%	78.17%
Size-large	**80.15**%	71.96%	61.05%
Diagonal-plane	**84.08**%	74.21%	71.73%
Depth-cloth	82.65%	76.70%	**83.21**%
Average	**84.93**%	76.10%	76.49%

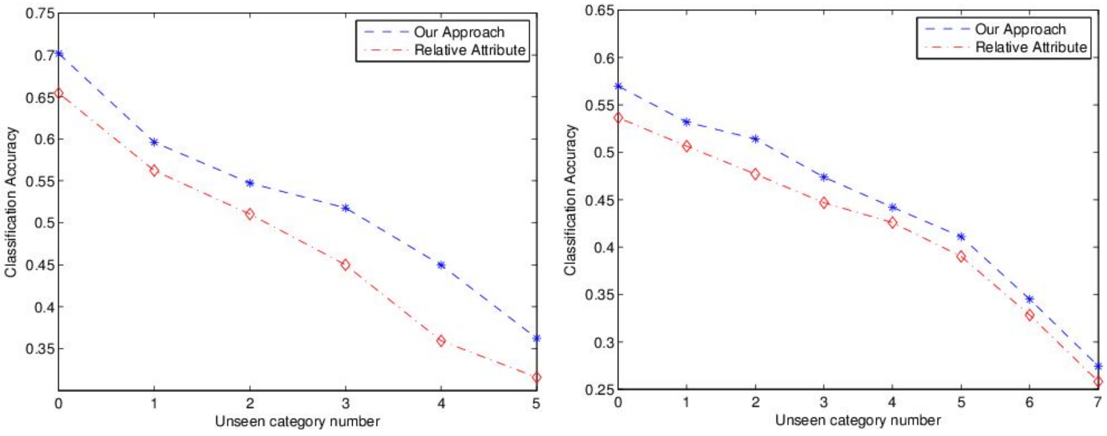
is 10. Table 3.2 describes the ranking accuracies on Shoes in all 10 attributes when the 60 samples are left out for training. In all of the attributes, better ranking accuracies are achieved by our proposed framework. The best performance gain is 8.5% in attribute “Pointy at the front”.

Both of these two experiments show that the more limited size of training dataset it is, the more benefits our proposed framework can gain from the relatedness among the attributes.

Finally, to show that the learned multi-attribute predictor captures intrinsically useful information for the underlying problem, we apply it to the task of zero-shot learning. Given training data from some ‘seen’ categories and some ‘unseen’ categories without any training data, zero-shot learning tries to learn a classifier to predict the category label of a new sample. We choose relative attribute as the comparing approach which has been shown to be the state-of-the-art work in [68]. I compute the average classification accuracies by running the experiment 5 rounds and in each round we randomly selected 400 training pairs for each seen categories to learn the projection

Table 3.2: Ranking accuracies of each attribute on Shoes when the 60 training samples of each attributes are left for training for each approach. Training pairs are generated from these 60 samples for our approach and relative attributes.

Attribute Name	Our Approach	Relative Attributes	Multi-task Learning
Pointy at the front	**82.90%**	74.52%	72.10%
Open	**76.41%**	72.52%	65.33%
Bright in color	**56.55%**	55.24%	53.17%
Covered with ornaments	**67.66%**	65.72%	51.15%
Shiny	**78.59%**	75.03%	72.71%
High at the heel	**76.23%**	70.67%	70.87%
Long on the leg	**74.53%**	71.91%	64.60%
Formal	**73.59%**	70.03%	60.61%
Sporty	**79.88%**	72.39%	69.30%
Feminine	**81.51%**	76.29%	68.45%
Average	**74.79%**	70.43%	64.83%



(a) OSR

(b) Shoes

Figure 3.4: Classification accuracies of zero-shot learning on OSR and Shoes. The number of unseen categories increases from 0 to 5 for OSR and from 0 to 7 for Shoes. My framework (blue) outperforms the competing approach (green) by 4% to 9% on OSR and by 2% to 4% on Shoes.

model. Same as in [68], we also assumed the data follows Gaussian distribution model and estimated the mean μ and the covariance matrix Σ through maximum likelihood estimation. Given a test image i and its corresponding ranking score vector $\tilde{x}_i$, we assigned the category label according to the maximum likelihood.

For the estimation of μ and Σ for unseen categories, we also adopted the similar schemes but added one more rule which we believe can better estimate the model: let $a_i^{(t)}$ and $a_j^{(t)}$ represent the t-th attribute value from the unseen category i and seen category j, we set $\mu_i^{(t)} = \frac{1}{n}\sum_{j=1}^{n} \mu_j^{(t)}$ and $\Sigma_i^{(t)} = \frac{1}{n}\sum_{j=1}^{n} \Sigma_j^{(t)}$.

Figure 3.4 shows the classification accuracies of zero-shot learning on OSR and Shoes. For OSR, the number of unseen categories increases from 0 to 5 while the total category number is 8 and the parameters of seen categories are estimated by randomly selected 30 samples; for Shoes, the number of unseen categories increases from 0 to 7 while the total category number is 10 and the parameters of seen categories are estimated by randomly selected 100 samples. The unseen categories are also randomly selected during each test round for both datasets. The result shows that the classification accuracies decrease as the number of unseen category increasing for both two datasets. On OSR, the accuracy of our framework outperforms the competing approaches by 4%∼9% and best performance gain got as the unseen category number is 4. On Shoes, our classification accuracy is 2%∼4% better than the results from the competing approach and the best performance gain is achieved when the unseen category number gets to 2.

3.3 Low-Rank Subspace Learning

Assuming that attributes are correlated as they are the linear combination of the same set bases, in this section we propose a multiple attribute relative learning framework which aims to learn a low-rank latent subspace which is able to expands the correlated attributes.

3.3.1 Proposed Approach

The proposed method is capable of learning a set of attributes from only relative rankings. Given the ranking information $\mathcal{O}_t$ and $\mathcal{S}_t$, where $\mathcal{O}_t$ is the set of pairs (i, j) that Data i is better than Data j for Attribute t, and $\mathcal{S}_t$ for the set of pairs being similar for Attribute t, we want to learn a classifier $\boldsymbol{w}_t$, such that,

$$\boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt}) \geq 1; \quad \forall(i,j) \in \mathcal{O}_t \tag{3.7}$$

$$|\boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt})| \approx 0; \quad \forall(i,j) \in \mathcal{S}_t \tag{3.8}$$

where $\boldsymbol{x}_{it}$ is the representation of Data i for Attribute t.

In many scenarios, e.g., image classification, we may need to learn multiple attributes and those attributes are likely to be correlated, as illustrated in the examples in Fig. 3.1. Conventional attribute learning approaches learn these attributes independently, and thus their intrinsic relatedness is not utilized. We propose to learn the attributes simultaneously under the multi-task learning framework. One popular multi-task learning model assumes that the classifiers of different tasks are similar and their differences to their mean are small. By combining this idea with relative learning, we obtain a baseline approach termed Multi-Task Relative Learning (MTRL), which is formally defined as

$$\begin{aligned}
\min{}_{W,\epsilon,\gamma} \quad & \sum_t^T \frac{1}{2}|\boldsymbol{w}_t|_2^2 + \frac{\lambda}{2}|\boldsymbol{w}_t - \frac{1}{T}\sum_\tau \boldsymbol{w}_\tau|_2^2 + \rho_1 \sum_{(i,j)\in\mathcal{O}_t} \epsilon_{ij}^t + \rho_2 \sum_{(i,j)\in\mathcal{S}_t} \gamma_{ij}^t \qquad (3.9)\\
\text{s.t.} \quad & \boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt}) + \epsilon_{ij}^t \geq 1 \\
& -\gamma_{ij}^t \leq \boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt}) \leq \gamma_{ij}^t \\
& \epsilon_{ij}^t \geq 0; \gamma_{ij}^t \geq 0
\end{aligned}$$

where $\boldsymbol{x}_{it}$ is the representation of i_{th} data for Task t, $\boldsymbol{w}_t$ is the t_{th} column of W (i.e., classifier of Task t) and $|\boldsymbol{w}_t|_2^2$ is related to the margin of the classifier for Tasks t. This problem can be solved by quadratic programming in its dual form.

In the above baseline approach, the usage of a common component has limited the form of correlation that the formulation could model (e.g., when the two tasks are negatively correlated). To this end, we model the correlation among the tasks by linear dependence, which is more flexible than MTRL. If we put the classifiers into the columns of a matrix, the resultant matrix would be low-rank, i.e., its nuclear norm would be small. Thus, we can formulate this new solution as

$$\begin{aligned}
\min_{W,\epsilon,\gamma} \quad & \sum_{t}^{T} \frac{1}{2}|\boldsymbol{w}_t|_2^2 + \lambda|W|_* + \rho_1 \sum_{(i,j)\in\mathcal{O}_t} \epsilon_{ij}^t + \rho_2 \sum_{(i,j)\in\mathcal{S}_t} \gamma_{ij}^t \\
\text{s.t.} \quad & \boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt}) + \epsilon_{ij}^t \geq 1 \\
& -\gamma_{ij}^t \leq \boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt}) \leq \gamma_{ij}^t \\
& \epsilon_{ij}^t \geq 0; \gamma_{ij}^t \geq 0
\end{aligned} \tag{3.10}$$

where $|\cdot|_*$ is the nuclear norm or the sum of singular values of the matrix for casting the low-rank constraint. We refer the proposed solution in Eqn. 3.11 as Max-Margin Multi-Attribute Learning with Low-Rank Constraint.

This problem is equivalent to the following problem by introducing a slack variable Z, which separates the low-rank constraint from the others:

$$\begin{aligned}
\min_{W,\epsilon,\gamma} \quad & \sum_{t}^{T} \frac{1}{2}|\boldsymbol{w}_t|_2^2 + \lambda|Z|_* + \rho_1 \sum_{(i,j)\in\mathcal{O}_t} \epsilon_{ij}^t + \rho_2 \sum_{(i,j)\in\mathcal{S}_t} \gamma_{ij}^t \\
\text{s.t.} \quad & \boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt}) + \epsilon_{ij}^t \geq 1 \\
& -\gamma_{ij}^t \leq \boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt}) \leq \gamma_{ij}^t \\
& \epsilon_{ij}^t \geq 0; \gamma_{ij}^t \geq 0 \\
& W = Z
\end{aligned} \tag{3.11}$$

By applying the Augmented Lagrange Multiplier (ALM) method to the equality constraint $W = Z$, we have:

$$\begin{aligned} \min_{W,\epsilon,\gamma} \quad & L(Z, W, \mathbf{b}, \gamma, \epsilon, \mu, Y) \\ \text{s.t.} \quad & \boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt}) + \epsilon_{ij}^t \geq 1 \\ & -\gamma_{ij}^t \leq \boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt}) \leq \gamma_{ij}^t \\ & \epsilon_{ij}^t \geq 0; \gamma_{ij}^t \geq 0 \end{aligned} \tag{3.12}$$

with

$$\begin{aligned} & L(Z, W, \mathbf{b}, \epsilon, \gamma, \mu, Y) = \lambda|Z|_* + \langle Y, W - Z\rangle + \\ & \frac{\mu}{2}|W - Z|_F^2 + \frac{1}{2}\sum_t |\boldsymbol{w}_t|_2^2 + \rho_1 \sum \epsilon_{ij}^t + \rho_2 \sum \gamma_{ij}^t \end{aligned} \tag{3.13}$$

where Y is the Lagrange multiplier, $\langle\cdot,\cdot\rangle$ is the inner product and μ is related to the Lipschitz constant of the primal problem

$$f(Z, W, \mathbf{b}, \epsilon) = \lambda|Z|_* + \frac{1}{2}\sum_t |\boldsymbol{w}_t|_2^2 + \rho_1 \sum \epsilon_{ij}^t + \rho_2 \sum \gamma_{ij}^t \tag{3.14}$$

3.3.2 Algorithm

The problem in Eqn. 3.12 can be solved via block coordinate descent, by considering the following two sub-problems:

Low-rank problem: Fix W, $\mathbf{b}$, ϵ, γ, μ and Y to solve Z, i.e.,

$$\min_Z \lambda|Z|_* + \langle Y, W - Z\rangle + \frac{\mu}{2}|W - Z|_F^2 \tag{3.15}$$

Ranking problem: Fix Z, μ and Y to solve W, $\mathbf{b}$, ϵ and γ, i.e.,

$$\begin{aligned} \min_{W,\epsilon,\gamma} \quad & \frac{\mu}{2}|W - Z|_F^2 + \langle Y, W - Z\rangle + \sum_t^\top \frac{1}{2}|\boldsymbol{w}_t|_2^2 + \rho_1 \sum_{(i,j)\in\mathcal{O}_t} \epsilon_{ij}^t + \rho_2 \sum_{(i,j)\in\mathcal{S}_t} \gamma_{ij}^t \\ \text{s.t.} \quad & \boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt}) + \epsilon_{ij}^t \geq 1 \\ & -\gamma_{ij}^t \leq \boldsymbol{w}_t^\top(\boldsymbol{x}_{it} - \boldsymbol{x}_{jt}) \leq \gamma_{ij}^t \\ & \epsilon_{ij}^t \geq 0; \gamma_{ij}^t \geq 0 \end{aligned} \tag{3.16}$$

In the following subsections, we will present specific methods for solving the two sub-problems of Eqn. 3.15 and 3.17, and then analyze the overall algorithm. The convergence analysis of the algorithm is included in Appendix A, where we show the proposed problem is convex and the proposed algorithm will converge to its global optimum.

Solving the Low-rank Problem For the low-rank problem, we want to find the optimal Z for Eqn. 3.15, which is a convex problem. It has been shown in [14] that the optimal solution to the problem $\min_X \lambda|X|_* + \frac{1}{2}|X - W|_F^2$ can be computed via a singular value thresholding algorithm, i.e., $U\mathcal{S}_\lambda(\Sigma)V^\top$, where $U\Sigma V^\top \leftarrow \text{svd}(W)$ is the singular value decomposition and $\mathcal{S}_\cdot(\cdot)$ is the thresholding operator:

$$\mathcal{S}_a(b) = \begin{cases} b - a & b \geq a \\ 0 & a \geq b \geq -a \\ b + a & \text{otherwise} \end{cases} \tag{3.17}$$

Thus the optimal solution to Eqn. 3.15 is $Z^* = U\mathcal{S}_{\frac{\lambda}{\mu}}(\Sigma)V^\top$, where $U\Sigma V^\top \leftarrow \text{svd}(W + \frac{1}{\mu}Y)$

Solving the Ranking Problem By recognizing $|W - Z|_F^2 = \sum_t |\boldsymbol{w}_t - Z_t|_2^2$ and $\langle Y, W - Z\rangle = \sum_t \langle Y_t, \boldsymbol{w}_t - Z_t\rangle$, the problem in Eqn. 3.17 can be decomposed into T independent smaller problems, where each smaller problem is associated with only one attribute/task:

$$\begin{aligned} \min_{W,\epsilon,\gamma} \quad & \frac{\mu}{2}|\boldsymbol{w}_t - Z_t|_F^2 + \langle Y_t, \boldsymbol{w}_t - Z_t\rangle + \frac{1}{2}|\boldsymbol{w}_t|_2^2 + \rho_1 \sum_k \epsilon_{kt} + \rho_2 \sum_l \gamma_{lt} \\ \text{s.t.} \quad & \boldsymbol{w}_t^\top E_{kt} + \epsilon_{kt} \geq 1 \\ & -\gamma_{lt} \leq \boldsymbol{w}_t^\top F_{lt} \leq \gamma_{lt} \\ & \epsilon_{kt} \geq 0; \gamma_{lt} \geq 0 \end{aligned} \tag{3.18}$$

where we use $E_{kt} = \boldsymbol{x}_{it} - \boldsymbol{x}_{jt} \forall (i,j) \in \mathcal{O}_t$, $F_{lt} = \boldsymbol{x}_{it} - \boldsymbol{x}_{jt} \forall (i,j) \in \mathcal{S}_t$, k, l to re-index $(i,j) \in \mathcal{O}_t$ and $(i,j) \in \mathcal{S}_t$. By applying the Lagrange multipliers, Eqn. 3.18 is equal to:

$$\begin{aligned}
&\max_{\alpha,\beta,\delta,\eta,\zeta} \min_{\mathbf{w},\epsilon,\gamma} \frac{\mu}{2}|\boldsymbol{w}_t - Z_t|_F^2 + \langle Y_t, \boldsymbol{w}_t - Z_t \rangle + \\
&\frac{1}{2}|\boldsymbol{w}_t|_2^2 + \sum_k \rho_1 \epsilon_k + \alpha_k(1 - \epsilon_k - \boldsymbol{w}_t^\top Y_k) - \eta_k \alpha_k + \\
&\sum_l \rho_2 \gamma_l + \beta_l(\boldsymbol{w}_t^\top Z_l - \gamma_l) + \delta_l(-\boldsymbol{w}_t^\top Z_l - \gamma_l) - \zeta_l \gamma_l \\
&\text{s.t. } \alpha, \beta, \delta, \eta, \zeta \geq 0 \qquad (3.19)
\end{aligned}$$

By checking the gradients, We have:

$$\begin{aligned}
\boldsymbol{w}_t &= \frac{\mu Z_t - Y_t + \sum_k \alpha_{kt} E_{kt} + \sum_l (\delta_{lt} - \beta_{lt}) F_{lt}}{1 + \mu} && (3.20) \\
0 &\leq \alpha_{kt} \leq \rho_1 && (3.21) \\
0 &\leq \beta_{lt} + \delta_{lt} \leq \rho_2
\end{aligned}$$

Accordingly, We have the dual form for the problem in Eqn. 3.19, which is a quadratic programming problem:

$$\begin{aligned}
&\min_{\mathbf{u}_t} \frac{1}{2} \mathbf{u}_t^\top K_t \mathbf{u}_t + \mathbf{f}_t^\top \mathbf{u}_t \qquad (3.22) \\
\text{s.t.} \quad &\mathbf{lb}_t \leq \mathbf{u}_t \leq \mathbf{ub}_t \\
&A_t^\top \mathbf{u}_t = 0
\end{aligned}$$

with

$$
\begin{aligned}
\mathbf{u}_t &= [\alpha^\top, -\beta^\top, \delta^\top]^\top \\
K_t &= \begin{bmatrix} E_t^\top E_t & E_t^\top F_t & E_t^\top F_t \\ F_t^\top E_t & F_t^\top F_t & F_t^\top F_t \\ F_t^\top E_t & F_t^\top F_t & F_t^\top F_t \end{bmatrix} \\
\mathbf{f}_t &= [E_t^\top(Y_t - \mu Z_t) - 1, F_t^\top(Y_t - \mu Z_t), \\
&\quad\; F_t^\top(Y_t - \mu Z_t)]^\top \\
\mathbf{lb}_t &= [0\mathbf{e}^\top_{|\mathcal{O}_t|}, -\rho_2\mathbf{e}^\top_{|\mathcal{S}_t|}, 0\mathbf{e}^\top_{|\mathcal{S}_t|}]^\top \\
\mathbf{ub}_t &= [\rho_1\mathbf{e}^\top_{|\mathcal{O}_t|}, 0\mathbf{e}^\top_{|\mathcal{S}_t|}, \rho_2\mathbf{e}^\top_{|\mathcal{S}_t|}]^\top \\
A_t &= [0_{|\mathcal{S}_t|\times|\mathcal{O}_t|}, -I_{|\mathcal{S}_t|\times|\mathcal{S}_t|}, I_{|\mathcal{S}_t|\times|\mathcal{S}_t|}] \\
\mathbf{b}_t &= \rho_2\mathbf{e}_{|\mathcal{S}_t|}
\end{aligned}
$$

where $\mathbf{e}_n \in \mathbb{R}^{n\times 1}$ is a all-1 vector, $0_{m\times n} \in \mathbb{R}^{m\times n}$ is all-0 matrix, $I_{n\times n} \in \mathbb{R}^{n\times n}$ is the identity matrix. Thus the dimension of the dual form of the ranking problem is $|\mathcal{O}_t| + 2|\mathcal{S}_t|$. After we solve the problem in Eqn. 3.22, we can compute the classifier according to Eqn. 3.22.

The overall algorithm for solving the problem of Eqn. 3.12 is summarized as Alg. 2. The proposed algorithm involves two major sub-problems. For the low-rank problem, the most time consuming step is the singular value decomposition for a matrix of dimension $D \times T$ (D is the input dimension), where the typical complexity for an exact decomposition is $O(\min{(TD^2, T^2D)})$. However, we may not be interested in a full/exact decomposition, but only the singular vectors whose singular value are sufficiently large (e.g., PROPACK[55]). For the classification problem, we are solving T quadratic programming problems of dimension n_t, with n_t the number of data points for the $t-th$ task.

The proposed problem in Eqn. 3.12 is convex and the proposed algorithm will

Algorithm 2 Alternating Optimization

```
Input:
    $X$,$\mathbf{L}$,$\lambda$,$\mu$,$\rho_1$,$\rho_2$,$\sigma$;
Output:
    Projection matrix $W$;
Initialize $W$ by solving $T$ tasks independently and $Y = \frac{W}{|W|_2}$;
while NOT converged do
    Solve the low-rank problem (Eqn. 3.15);
    Solve the ranking problem (Eqn. 3.17);
    Update $Y = Y + \mu(W - Z)$ and $\mu = \mu \times \sigma$;
    Check convergence;
end while
return $W$;
```

converge to its global optimum. The proof is given in Appendix A. For the stopping criterion, we compute $\frac{|W-Z|_F^2}{|W|_F^2}$. If this value is sufficiently small (e.g., 10^{-6}), we will terminate the optimization. In our experiments, we observed the convergence was reached within 100 iterations.

There are three parameters required for the proposed algorithm: λ (controlling the weight of the nuclear norm term), μ (controlling the weight of the term $|W-Z|_F^2$) and σ (controlling the increasing speed of μ). The selection of λ depends on the correlation among the tasks: if high correlation among the tasks is expected, we should use a large λ (i.e., $|W|_*$ should be small); otherwise, we should set λ to a small value. When $\lambda = 0$, the proposed method is equivalent to the relative attribute learning method, where each task is solved independently. For μ, we utilizes the analysis in [57] and set it to $\frac{1.25\lambda}{|W|_2}$. For ρ, we use $\rho = 1.2$.

More details of the algorithm and its convergence is attached in Appendix A.

3.3.3 Experiments

we evaluated the proposed approach on both synthetic data and real image/video data sets. The proposed method is compared with the relative attribute method of [67], where each attribute is learned independently, and with the multi-task relative learning method, where the classifying/ranking functions of the attributes are assumed to share a common component. Since no validation set is available for the real datasets (and they are too small to support creation of a validation set), we did not rely on cross-validation for parameter tuning. Instead, in the experiments we used the following fixed parameters for the proposed method and the multi-task relative attribute learning method: $\lambda = 10000$, $\rho_1 = 100$ and $\rho_2 = 100$. Default parameters were used for relative attribute learning.

Simulated Experiment

In this section, we evaluate the proposed method on synthetic data. We generate $T = 10$ tasks and the feature dimension of each tasks is $D = 1000$. The ground truth classification function (or ground truth ranking function) for Task t is $\boldsymbol{w}_t$, where $\boldsymbol{w}_t$ is the $t - th$ column of W. W is generated as:

$$W_0 = \text{rand}(D, T) - 0.5 \tag{3.23}$$

$$\text{svd}(W_0) \rightarrow U\Sigma V^\top \tag{3.24}$$

$$W = U(:, 1:r)\Sigma(1:r, 1:r)V^\top(:, 1:r) \tag{3.25}$$

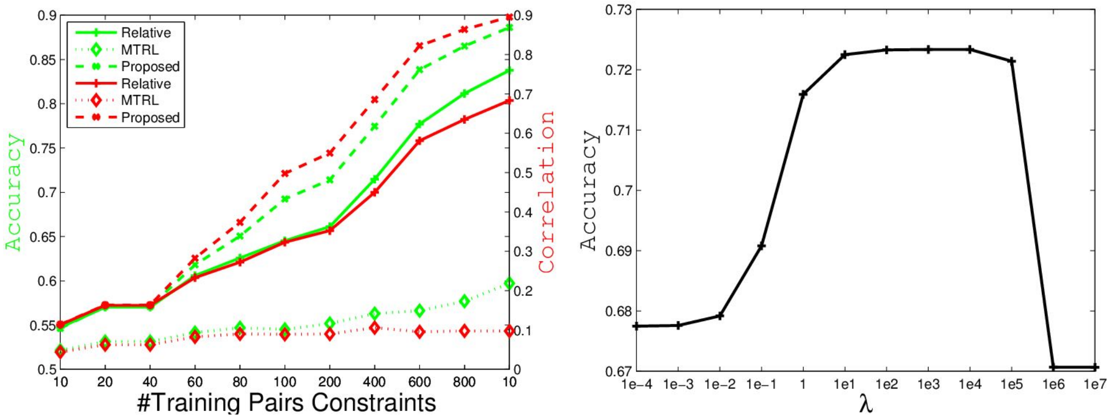
where $r = 2$ is the desired rank of W. Note that by generating the ground truth classification function in this way, the classifiers are not necessarily similar or to share a common component. I uniformly draw the data X for each task, and each set of data contains 1000 data points. For Task t, we randomly select P pairs as the training pairs, i.e., $(i, j) \in \mathbb{E}$ if $\boldsymbol{w}_t^\top X_i - \boldsymbol{w}_t^\top X_j \geq \tau$; or $(j, i) \in \mathbb{E}$ if $\boldsymbol{w}_t^\top X_i - \boldsymbol{w}_t^\top X_j \leq -\tau$; otherwise

$(i, j) \in \mathbb{F}$, where τ is the predefined margin. The proposed algorithm is applied to the training pairs to learn the ranking function for the tasks, with comparison with the relative attribute (refer as "Relative") method and also the baseline (i.e., Multi-Task Relative Learning or MTRL) method. We also test different combinations of λ (from 10^{-4}, i.e., low requirement of the low-rank constraint, to 10^{7}, i.e., high requirement on the low-rank constraint) and P (from 10 to 1000), where the results are shown in Fig. 3.5.

From Fig. 3.5 (a), we can observe that, although the accuracy and the correlation increase with more training pairs, i.e., larger P, the proposed method consistently performs better than the other two competitors. Especially, when $P = 1000$, the correlation between the ranking functions learned by the proposed method and the ground truth ones is about 0.9, which is significantly better than 0.68 achieved by the relative attribute method. The results indicate that the proposed method is more likely to recover the ground truth ranking functions than the relative attribute method, when given the same number of training pairs. The performance of MTRL is significantly lower. This could be explained by the assumption made by its formulation: the classification functions of the tasks should be similar (or share a common component), which is not always true in the generation of the data (e.g., the classification functions can be negatively correlated).

Fig. 3.5 (b) illustrates the performance of the proposed approach with different settings for the parameter λ, which controls the contribution of the low-rank constraint. From the plot, we can observe that the performance is stable for a wide range of $\lambda (\lambda \in [10, 10^{4}])$ and the best result is obtained when $\lambda = 10^{4}$.

I also performed simulations using data whose ground truth ranking functions are not correlated, i.e., the functions are linear independent by setting $r = 10$. The results are shown in Fig. 3.6, from which we can find that, the proposed method



(a) (b)

Figure 3.5: The result for simulation experiments with varying P (a) ($\lambda = 10^4$) and λ (b) ($P = 100$), where the dashed curves correspond to the result of the proposed method, dot curve for the MTRL and solid curve for the relative attributes method. We compute accuracy (y axis of red curves) of the learned ranking functions and the correlation (y axis of green curves) between the learned ranking functions and ground truth ones. The X axis are the p (a) and λ (b) accordingly.

(dashed curve) obtained similar results as the relative attribute learning method (solid curve) in both accuracy and correlation. However, the MTRL (dotted curve) obtained obviously worse performances in both accuracy and correlation. This demonstrates that the proposed method is robust to different correlation levels of the tasks, and its performance is still comparable to that of the relative attribute learning method even when the tasks are totally linear independent. The performance of MTRL method, however, degrades dramatically when the assumption about the relatedness of the tasks does not hold.

For understanding the computational efficiency of the proposed method, we note that its formulation as well as solutions bear similarity to MTRL, which is well-understood to have a polynomial complexity over the number of constraints. Hence the proposed method is expected to have the same order of complexity over the number of training pairs. To empirically verify this, we use Fig. 3.7 to depict the running times of the proposed approach under different numbers of training pairs,

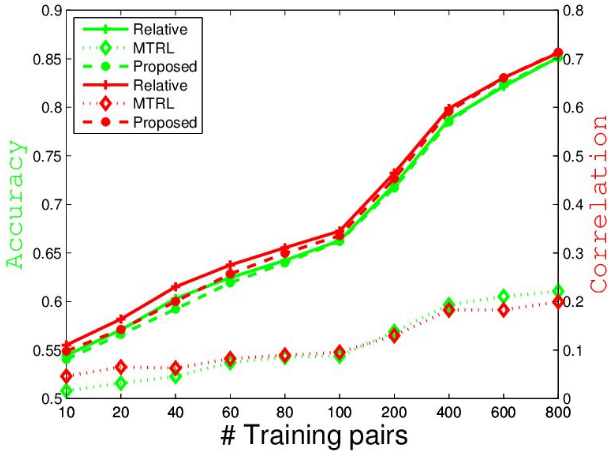


Figure 3.6: The result for simulation experiment when the ground truth ranking functions of the tasks are linear independent. The matrix consisted of ground truth ranking functions as its columns has maximal singular value 0.7 and minimal singular value 0.6.

with the comparison to the relative attribute learning method and the MTRL method. It can be observed that the proposed method, while being more expensive than the basic relative attribute learning method, is indeed in par with the MTRL method in terms of asymptotic time complexity. Note that both axes of Fig. 4 are with logarithm for better illustration.

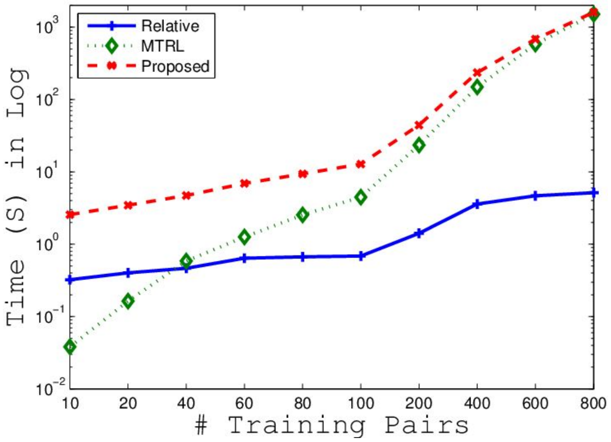


Figure 3.7: The computation time of the proposed approach given different number of training pairs, with the comparison to the relative attribute learning methods and MTRL method. For the time axis (y-axis), we use logarithm.

Learning Attributes for Images

To evaluate the performances of the proposed algorithm on real data, we utilize two datasets, (1) Outdoor Scene Recognition (OSR) Dataset [66] containing 2688 images from 8 categories; (2) A subset of the Public Figure Face Database (PubFig) [52] containing 800 images from 8 random identities (100 images each). We directly used the processed data from [67] and the same experiment settings. To demonstrate that the attributes in these datasets indeed exhibit correlation, we first computed the histogram of the pairwise correlation coefficients among the tasks for each of the datasets, and the results are shown in Fig. 3.8. It is evident from these plots that the tasks are correlated. For example, one can observe that there is a non-trivial mass covering beyond the interval [-0.5, 0.5] in either of the plots. Note that, both the rank of classifier matrix (W) and the correlation coefficients between the tasks are some measurements of the dependency. For ideal case (perfectly dependent), the rank should be 1 and the correlation coefficient should be $+/-1$ cross different tasks.

Next we report the ranking accuracy of the proposed method and compare with the relative attribute method ("Relative" in short) and the multi-task relative attribute learning method ($MTRL$ in short). We randomly picked only 5% of those training pairs for evaluating. For the proposed method, we fixed λ to 10000. For the relative attribute, we employ another baseline approach reported as "Relative*" where over $20,000$ training pairs are involved for learning.

All the results on the two datasets are summarized in Tab. 3.3 and 3.4. From the result we can observe that the proposed method outperforms the other methods in both cases except that the Relative* row of (A). [67] has an insignificant gain over our method, even with much more training pairs (see also the caption of the Table). Additionally, we can observe that the performance gain of our method over Relative or

$MTRL$ (when all trained under the same protocol with only 5% of the training pairs used in [67]) varies. This could be explained by possible varying degree of correlation among the tasks in the two datasets, as alluded by Fig. 5. However, we note that the correlation coefficient used in Fig. 5 measures only linear dependency and thus it is not proper to draw any quantitative conclusion. Additionally, the low-rank constraint would generally work better when there are many tasks considered jointly (comparing with the feature dimension) [44]. This is consistent with the results in the Table (e.g., better gain by the proposed in (B) than in (A)). The low-rank constraint used in the proposed method is more flexible than forcing the tasks to share common components in capturing the intrinsic relatedness of the attributes, which explains the gain of the proposed over $MTRL$.

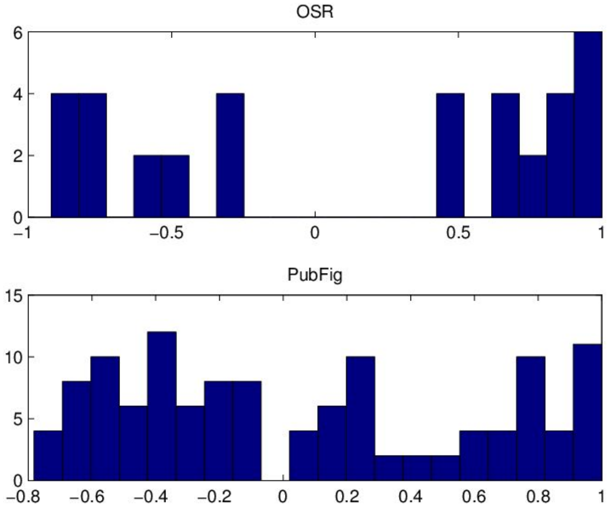


Figure 3.8: The histogram of Pearson's correlation coefficients among the tasks for both two datasets. From these histograms, we can observe that the attributes are correlated, as there are non-trivial mass covering the regions towards -1 or 1. Note that, Pearson's correlation coefficient measures only linear dependency, and thus even if it is low, the tasks could still be highly dependent.

Table 3.3: Ranking Accuracy of OSR Data

Attribute Name	Proposed	MTRL	Relative	Relative*
Natural	**94.81**%	90.50%	93.69%	94.63%
Open	90.67%	84.97%	91.05%	**91.28**%
Perspective	**86.81**%	82.04%	85.75%	86.17%
Size-large	86.69%	80.78%	86.78%	**86.90**%
Diagonal-plane	**88.63**%	83.15%	87.62%	87.96%
Depth-cloth	88.26%	78.83%	87.82%	**89.07**%
Average	89.28%	83.38%	88.78%	**89.33**%

Table 3.4: Ranking Accuracy of Shoes Data

Attribute Name	Proposed	MTRL	Relative	Relative*
Masculine-looking	**84.08**%	74.52%	81.32%	83.26%
White	**81.18**%	80.30%	76.99%	79.91%
Young	**85.14**%	84.08%	81.86%	83.54%
Smiling	**84.13**%	80.99%	80.75%	83.04%
Chubby	**81.07**%	79.45%	77.54%	79.46%
Visible-forehead	89.08%	85.96%	87.36%	**89.65**%
Bushy-eyebrows	**84.24**%	83.82%	79.51%	82.19%
Narrow-eyes	**83.11**%	81.83%	81.66%	82.93%
Pointy-nose	**82.63**%	82.21%	75.57%	78.79%
Big-lips	**84.46**%	83.47%	78.41%	81.09%
Round-face	**86.03**%	83.53%	81.59%	83.16%
Average	**84.25**%	82.70%	80.23%	82.46%

3.4 Conclusions

In this Chapter, we proposed a framework for relative multi-attribute prediction through multiple task learning. By employing a multi-task learning framework for learning multiple attributes with only relative labels, our proposed framework is able to capture the intrinsic relatedness among the different attributes. The proposed method was evaluated on two public datasets OSR and Shoes with the comparison with the baseline approaches of relative attribute and multi-task learning. Through the experiments on image ranking and zero shot learning, we demonstrated that our method obviously outperforms the baseline methods in both ranking and classification capacities.

Chapter 4

ATTRIBUTE PREDICTION BASED ON POINTWISE LABEL

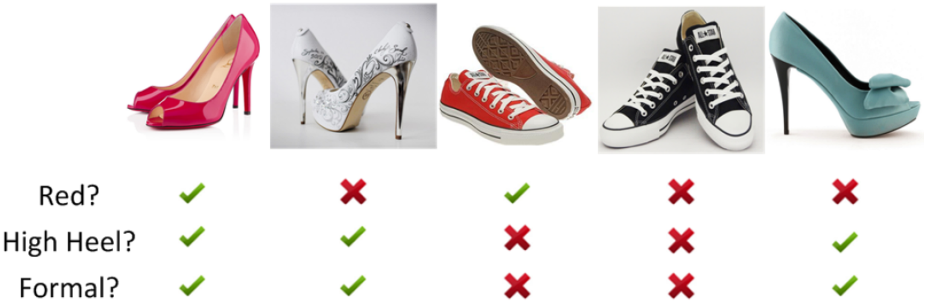
In this chapter, I discuss my proposed approaches on how to utilize the special structure of attribute relatedness for attribute prediction.

4.1 Introduction

Recent literature has witnessed fast development of representations using semantic attributes, whose goal is to bridge the semantic gap between low-level feature representation and high-level semantic understanding of visual objects. Attributes refer to visual properties that help describe visual objects or scenes such as "natural" scenes, "fluffy" dogs, or "formal" shoes. Visual attributes exist across object category boundaries and many methods have been employed in applications including object recognition [30], face verification [85] and image search [51, 79].

Good representations of semantic attributes are often built on top of high-dimensional, low-level features. Attribute learning directly based on such raw, high-dimensional features may suffer from the problem of dimensionality curse. Further, often it is reasonable to assume that not all the low-level features would have equal contribution to all the attributes. Feature selection, selecting a subset of most relevant features for a compact and accurate presentation, is proven to be an effective and efficient way to handle high-dimensional data [86].

Considering exploring the correlation among attributes, multi-task joint feature selection has been introduced by [19] for attribute ranking. However, this work assumes that all attributes are correlated by sharing the same subset of features, which is not always accurate. For example, as shown in Figure 4.1, a "high-heel" shoe is usu-



Red?	✓	✗	✓	✗	✗
High Heel?	✓	✓	✗	✗	✓
Formal?	✓	✓	✗	✗	✓

Figure 4.1: Illustration of Shoe images with three corresponding attributes “High Heel”, “Formal” and “Red” where “High heel” are highly related with “Formal” but weakly related with “Red”.

ally considered as a “formal” shoe as well. It is reasonable to assume these attributes share the same subset of features, e.g., shape-related descriptors. However, it is hard to identify whether “high heel” or “formal” shoes are in red, which suggests the attribute “color” may not share the same subset of features with the other attributes but is determined by, e.g., color-related descriptors. In other words, attributes are usually related in clustering structures. [43] first explores such clustered relatedness on attribute prediction. However, their approach requires manually specified group structure as prior. To my knowledge, there is still lack of a feature selection approach being able to identify grouping/clusering structures among attributes for improved attribute prediction.

4.2 Clustering based Joint Feature Selection for Semantic Attribute Prediction

In this section, I propose a regularization-based multi-task feature selection approach that aims at automatically partitioning the attributes into groups while simultaneously utilizing such group information for attribute-dependent feature selection. I employ a clustering regularizer for attribute partition, where strong attribute relatedness is assumed to exist within each cluster. Besides, a group-sparsity regularizer is imposed on the objective function to encourage intra-cluster feature sharing and

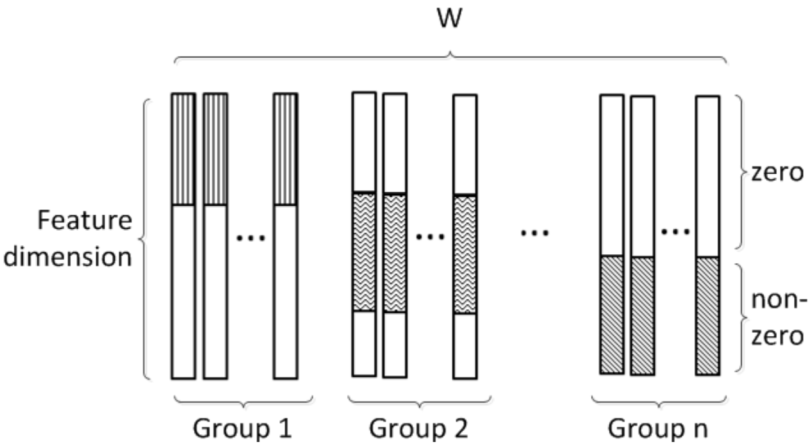


Figure 4.2: Demonstration of feature selection based on group structure. Tasks in the same group are strong related with each other and share the same subset dimension of features; tasks in different group are weakly correlated and featured with different non-zero dimensions.

inter-cluster feature competition. Under this formulation, I propose an alternating structure optimization algorithm, which efficiently solves the relaxed form of the proposed formulation. I verify the effectiveness and generalization capability of our approach on both synthetic and real-world benchmark datasets. The results show that our approach outperforms the state-of-the-art approaches on feature selection, attribute prediction and zero-shot learning.

Suppose that we are given a multi-task learning problem with m tasks (attributes); each task i is associated with a set of training data of d dimension and n samples: $(\boldsymbol{x}_1^i, y_1^i), \ldots, (\boldsymbol{x}_n^i, y_n^i) \subset \mathbb{R}^d \times \mathbb{R}$, we denote $W = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m] \in \mathbb{R}^{d \times m}$ as the projection weight matrix to be estimated where each column $\boldsymbol{w}_i$ is the weight vector of the i-th task. Tasks may exhibit grouping structures, as illustrated in Figure 4.2. The tasks in the same group are highly correlated and thus sharing the same subset dimension of features. The tasks in different groups are weakly correlated and have different subset of non-zero dimension of features. Our goal is to design an approach which can automatically detect such group structure and utilize such group correlation information for feature selection.

4.2.1 Proposed Framework

Let $\mathcal{F} = \{f_1, f_2, \ldots, f_d\}$ be the set of d features and then we can represent a set of n instances by the feature set $\mathcal{F}$ as $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{d\times n}$. Let $\mathcal{C} = \{c_1, c_2, \ldots, c_m\}$ be the set of m attribute labels and $Y = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n] \in \{0,1\}^{m\times n}$ denotes the label matrix where $\boldsymbol{y}_i \in \mathbb{R}^m (i = 1, 2, \ldots, n)$ is the label vector of the i-th instance. We aim to select $K(K\leq d)$ most relevant features from $\mathcal{F}$ by leveraging X, Y and the attribute correlation in $\mathcal{C}$. Let $\mathbf{s} = \pi(\overbrace{0, \ldots, 0}^{d-K}, \overbrace{1, \ldots, 1}^{K})$, where $\pi(\cdot)$ is the permutation function and K is the number of features to select where $\mathbf{s}_i = 1$ indicates that the i-th feature is selected. The original data can be represented as diag($\mathbf{s}$)$\mathbf{X}$ with K selected features where diag($\mathbf{s}$) is a diagonal matrix. We assume that a linear projection matrix $W = [\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_m] \in \mathbb{R}^{d\times m}$ maps the data X to its label matrix Y where $\boldsymbol{w}_i \in \mathbb{R}^d$ is the projection vector for the i-th class c_i. If we do not consider attribute correlation, we can select K features via solving the following optimization problem:

$$\begin{aligned} \min_{W,\mathbf{s}} \quad & L(W^\top \text{diag}(\mathbf{s}) X, Y) \\ s.t., \quad & \mathbf{s} \in \{0,1\}^n,\ \mathbf{s}^T \mathbf{1}_n = K \end{aligned} \tag{4.1}$$

where $L(\cdot)$ is the loss function and typical choices of loss functions include least square and logistic regression.

Modeling Label Correlation

Based on the assumption that correlated attributes would share the same features, I propose to model attribute correlation via learning the clustering structures. Let E be a permutation partition matrix, then a partition of the projection matrix W into

k clusters can be formed as:

$$WE = [W_1, W_2, \dots, W_k], \quad W_i = [\boldsymbol{w}_1^{(i)}, \boldsymbol{w}_2^{(i)}, \dots, \boldsymbol{w}_{n_i}^{(i)}]; \tag{4.2}$$

where $W_i \in \mathbb{R}^{d \times n_i} (i = 1, 2, \dots, k)$ is the i-th partitioned group includes n_i projection vectors (or attribute labels). The associated sum-of-squares cost function for the partition can be formulated as

$$\sum_{i=1}^{k} \sum_{j=1}^{n_i} \|\boldsymbol{w}_j^{(i)} - \boldsymbol{m}_i\|^2, \boldsymbol{m}_i = \sum_{j=1}^{n_i} \boldsymbol{w}_j^{(i)} / n_i \tag{4.3}$$

where $\boldsymbol{m}_i$ denotes the mean vector of the i-th cluster. Let $e_i = [1, 1, \dots, 1]^\top \in \mathbb{R}^{n_i \times 1}$, then Eqn. (4.3) can be derived as

$$\begin{aligned} \sum_{i=1}^{k} \sum_{j=1}^{n_i} \|\boldsymbol{w}_j^{(i)} - \boldsymbol{m}_i\|^2 &= \sum_{i=1}^{k} \|W_i(I_{n_i} - \frac{e_i {e_i}^\top}{n_i})\|_F^2 \\ = \sum_{i=1}^{k} \mathbf{Tr}(W_i^\top W_i) - (\frac{{e_i}^\top}{\sqrt{n_i}}) W_i^\top W_i (\frac{e_i}{\sqrt{n_i}}) \end{aligned} \tag{4.4}$$

Let $F = \text{diag}(\frac{e_1}{\sqrt{n_1}}, \frac{e_2}{\sqrt{n_2}}, \dots, \frac{e_k}{\sqrt{n_k}}) \in \mathbb{R}^{m \times k}$ be an orthonormal matrix, then Eqn. (4.4) can be rewritten as

$$\mathbf{Tr}(W^\top W) - \mathbf{Tr}(F^\top W^\top W F) \tag{4.5}$$

To make the problem tractable, we ignore the special structure of F and let it be an arbitrary orthonormal matrix. By adding a global penalty $\mathbf{Tr}(W^\top W)$ measuring how large the weight vectors are, capturing label correlation is to partition W into k clusters, which can be achieved by solving the following optimization problem:

$$\min_{F^\top F = I_k} \mathbf{Tr}(W^\top W) - \mathbf{Tr}(F^\top W^\top W F) + \gamma \mathbf{Tr}(W^\top W) \tag{4.6}$$

Feature Selection

With the model component to capture attribute correlation in Eqn. (4.6), the proposed feature selection framework is to solve the following optimization problem:

$$\begin{aligned}\min_{W,F,\mathbf{s}} \; & L(W^\top \mathrm{diag}(\mathbf{s})X, Y) + \gamma \mathbf{Tr}(W^\top W) \\ & + \beta(\mathbf{Tr}(W^\top W) - \mathbf{Tr}(F^\top W^\top W F)) \\ s.t. \quad & F^\top F = I_k,\ \mathbf{s} \in \{0,1\}^n,\ \mathbf{s}^\top \mathbf{1}_n = K \end{aligned} \tag{4.7}$$

where β controls the contribution from modeling label correlation and γ controls the generalization performance.

The constraint on $\mathbf{s}$ makes Eqn. (4.7) a mixed integer programming problem, which is difficult to solve. We observe that diag($\mathbf{s}$) and W are in the form of $W^T\mathrm{diag}(\mathbf{s})$. Since $\mathbf{s}$ is a binary vector and $d - K$ rows of the diag($\mathbf{s}$) are all zeros, $W^T\mathrm{diag}(\mathbf{s})$ is a matrix where the elements of many rows are all zeros. This motivates us to absorb diag($\mathbf{s}$) into W as $W = W^T\mathrm{diag}(\mathbf{s})$, and add $\ell_{2,1}$-norm on each grouped W_i to encourage sparse-based group-wise joint feature selection. With this relaxation, Eqn. (4.7) can be rewritten as:

$$\begin{aligned}\min_{W,F;F^\top F = I_k} \; & L(W^\top X, Y) + \alpha \sum_{i=1}^{k} \|W_i\|_{2,1} + \gamma \mathbf{Tr}(W^\top W) \\ & + \beta(\mathbf{Tr}(W^\top W) - \mathbf{Tr}(F^\top W^\top W F)) \end{aligned} \tag{4.8}$$

where α controls the sparsity of W. The key idea lying here is that we use the clustering regularizer to partition the tasks into groups where strong correlation exists among tasks in the same group; and feature selection based on such group structures would make sure appropriate feature subsets are selected to represent the respective semantic attributes.

4.2.2 Algorithm

In this section, we first introduce an optimization algorithm to seek an optimal solution (summarized in Algorithm 3) for Eqn. (4.8). Then we propose an approach to estimate the attribute assignment (summarized in Algorithm 4).

Optimization

The optimization problem in Eqn. (4.8) is non-convex non-smooth, which makes the formulation difficult to solve in its original form. Thus we adopt several relaxations to make it solvable.

The attribute correlation regularization in Eqn. (4.6) can be rewritten as:

$$\beta \mathbf{Tr}(W((1+\eta)I - FF^{\top})W^{\top}) \tag{4.9}$$

where $\eta = \gamma/\beta > 0$. Let $M = FF^{\top}$, according to [98] the previous regularizer can be relaxed into the following convex form:

$$\begin{aligned} &\beta\eta(1+\eta)\mathbf{Tr}(W(\eta I + M)^{-1}W^{\top}) \\ &s.t. \quad tr(M) = k,\ M \preceq I,\ M \in \mathbb{S}_{+}^{m} \end{aligned} \tag{4.10}$$

where $\mathbb{S}_{+}^{m}$ is the set of $m \times m$ positive semidefinite matrices.

Following a similar idea in [6], we reformulate Eqn. (4.8) by squaring the $\ell_{2,1}$ norm. Since the $\ell_{2,1}$ norm is positive, the squaring represents a smooth monotonic mapping. Without loss of the generality, we adopt the traditional least square loss for demonstration in this work. Then we get the following jointly convex smooth objective function regarding to W and M.

$$\begin{aligned} \arg\min_{W,M} & \|W^\top X - Y\|_F^2 + \alpha \sum_{i=1}^{k} (\|W_i\|_{2,1})^2 \\ & - \beta\eta(1+\eta)\mathbf{Tr}(W(\eta I + M)^{-1}W^\top) \\ & s.t. \quad tr(M) = k,\ M \preceq I,\ M \in \mathbb{S}_+^m \end{aligned} \tag{4.11}$$

Since it is difficult to optimize the linear projection matrix W and attribute correlation matrix M simultaneously, we employ Alternating Structure Optimization (ASO), which has been shown to be effective in many practical applications [10, 72] and is guaranteed to converge to a global optimal solution.

Optimizing M when fixing W

Given a fixed W, the optimization problem is decoupled into the following optimization problem:

$$\begin{aligned} & \min_{M}\ \mathbf{Tr}(W(\eta I + M)^{-1}W^\top) \\ & s.t. \quad tr(M) = k,\ M \preceq I,\ M \in \mathbb{S}_+^m \end{aligned} \tag{4.12}$$

I solve the problem based on the following Lemma due to [98]:

Lemma 1. *For the optimization problem in Eqn. (4.12), let $W = U\Sigma V$ be the singular value decomposition of W where $\Sigma = diag([\sigma_1, \sigma_2, \ldots, \sigma_m])$, $M = Q\Lambda Q^\top$ be the Eigen decomposition of M where $\Lambda = diag([\lambda_1, \lambda_2, \ldots, \lambda_q])$ and q be the rank of Σ. Then the optimal Q^* is given by $Q^* = V$ and the optimal Λ^* is given by solving the following optimization problem:*

$$\begin{aligned} \Lambda^* = \arg\min_{\Lambda} & \sum_{i=1}^{q} \frac{\sigma_i^2}{\eta + \lambda_i} \\ s.t. \quad & \sum_{i=1}^{q} \lambda_i = k, 0 \leq \lambda_i \leq 1 \end{aligned} \tag{4.13}$$

```
Algorithm 3 Feature Selection Optimization
Input:
    1. Multiple attribute data $\{X, Y\}$;
    2. Parameters $\alpha$, $\beta$, $k$(optional) and the number of selected features $K$;
    3. The initial projection matrix $W_0$;
Output:
Set $W = W_0$;
repeat
    Update $M$ according to Eqn. (4.12);
    Update $\boldsymbol{r}$ according to Alg. 4;
    Update $\boldsymbol{\delta}$ according to Eqn. (4.15);
    Update $W$ according to Eqn. (4.16);
until Converges
Sort each feature according to $\|\mathbf{w}^i\|_2$ in descending order of each group;
return The group-wise top-$K$ ranked features;
```

Eqn. (4.13) can be solved using the similar technology in [41].

Optimizing W When Fixing M

The squared group-wise $\ell_{2,1}$ norm in Eqn. (4.11) is still difficult to derive directly. To alleviate that, we introduce some positive dummy variables $\delta_{ij} \in \mathbb{R}^+$ which satisfies $\sum_i \sum_j \delta_{ij} = 1$. [4] proves an upper bound of the squared $\ell_{2,1}$ norm in terms of the positive dummy variables

$$\sum_{i=1}^{k} (\|W_i\|_{2,1})^2 = (\sum_{i=1}^{k} \sum_{j=1}^{d} \|\boldsymbol{w}_{i,j}\|_2)^2 \leq \sum_{i=1}^{k} \sum_{i=1}^{d} \frac{(\|\boldsymbol{w}_{i,j}\|_2)^2}{\delta_{ij}} \tag{4.14}$$

where $\boldsymbol{w}_{i,j} \in \mathbb{R}^{1 \times m}$ is the row vector of W_i. Thus δ_{ij} can be updated by holding the

```
Algorithm 4 Cluster Assignment Estimation
Input: M;
Output:
1: Approximate $F$ by top-ranked eigenvector of $Q$;
2: Calculate $R_{11}, R_{12}$ by applying QR decomposition with column pivoting on $F$ by Eqn. (4.18);
3: Calculate $\hat{R}$ by Eqn. (4.19);
4: calculate $r$ by Eqn. (4.20) for each attribute;
5: return Cluster assignment vector $\boldsymbol{r}$;
```

equality:

$$\delta_{ij} = \|\boldsymbol{w}_{i,j}\|_2 / \sum_{j=1}^{d} \|\boldsymbol{w}_{i,j}\|_2. \tag{4.15}$$

Given a fixed M, each projection vector $\boldsymbol{w}$ can then be updated by optimize the following problem

$$\begin{aligned} \arg\min_{W} \|W^T X - Y\|_F^2 + \alpha \sum_{i=1}^{k} \sum_{i=1}^{d} \frac{(\|\boldsymbol{w}_{i,j}\|_2)^2}{\delta_{ij}} \\ - \beta\eta(1+\eta)\mathbf{Tr}(W(\eta I + M)^{-1} W^\top) \end{aligned} \tag{4.16}$$

which can be solved by gradient-type approach.

Estimating Attribute Assignment

The group-wise feature selection is conducted by the clustering structure of the attribute. However, given the M optimized by the previous algorithm, it is not readily possible to observe the cluster assignment of the attributes because M is spectrally relaxed. In this subsection, we propose an approach to acquire the cluster structure.

I first need to obtain a good approximation of the cluster indicator matrix F. Given M, we first apply Eigen decomposition $M = Q\Lambda Q^\top$ where each column of Q is the eigenvector and each diagonal element of Λ is the eigenvalue. Then we rank the columns of Q in decreasing order according to its corresponding eigenvalues, and the top-ranked k columns give an approximation of the cluster assignment matrix F. The number of the cluster k can be either manually specified or automatically explored by setting a threshold ($10e-8$ in our experiment) regarding to the absolute value of the eigenvalue.

After obtaining F, without loss of generality, we assume the optimized $W = [W_1, W_2, \cdots, W_k]^T$ where the submatrix W_i includes all attributes belonging to the i-th cluster. Let $\boldsymbol{t}_i = [t_{i1}, t_{i2}, \ldots, t_{in_i}]^T$ denote the largest eigenvector of ${W_i}^T W_i$, [96] showed that F can be reformulated as

$$F^T = [\underbrace{t_{11}\boldsymbol{v}_1, \cdots, t_{1s_1}\boldsymbol{v}_1}_{cluster1}, \cdots, \underbrace{t_{k1}\boldsymbol{v}_k, \cdots, t_{ks_1}\boldsymbol{v}_k}_{clusterk}] \tag{4.17}$$

where $V^T = [\boldsymbol{v_1}, \boldsymbol{v_2}, \cdots, \boldsymbol{v_k}] \in \mathbb{R}^{k \times k}$ is an orthogonal matrix.

Since $\boldsymbol{v_i}$ is orthogonal to each other, the cluster structure can be acquired by picking up a column of F which has the largest norm as the first cluster, and orthogonalizing the other columns against this column. Then the same process is executed on the rest of columns until all clusters are identified. This process is identical to a QR decomposition with column pivoting on F

$$F^T = Q[R_{11}, R_{12}]P^T \tag{4.18}$$

where $Q \in \mathbb{R}^{k \times k}$ is an orthogonal matrix, $R_{11} \in \mathbb{R}^{k \times k}$ is an upper triangular matrix and $P \in \mathbb{R}^{m \times m}$ is a permutation matrix. Then we calculate the cluster assignment matrix $\hat{R} \in \mathbb{R}^{k \times m}$ by

$$\hat{R} = [I_k, R_{11}^{-1} R_{12}]P^T \tag{4.19}$$

where $I_k \in \mathbb{R}^{k \times k}$ is an identity matrix. The cluster assignment information can then be inferred from $\hat{R}$. The cluster membership of each attribute (column) is determined by the row index of the largest element (in absolute value) of the corresponding column in $\hat{R}$. Denote $\boldsymbol{r} \in \mathbb{R}^m$ as the cluster identification vector where r_i records which cluster the i-th class belongs to, then $\boldsymbol{r}$ can be calculated by

$$r_i = \arg\max_j \hat{r}_{ij} \tag{4.20}$$

where $\hat{r}_{ij}$ is the (i, j)-th entry of $\hat{R}$.

The analysis of the algorithm is attached in Appendix B.

4.2.3 Experiments

In this section, we first verify the effectiveness of our proposed approach on one synthetic dataset. Since the proposed approach can be generalized to general multi-label problem, we evaluate the feature selection capability on various benchmark datasets. At last we evaluate the attribute prediction and zero-shot learning capabilities on image benchmark datasets. All the datasets are standardized to zero-mean and normalized by the standard deviation. For all approaches, the super parameters are selected via cross-validation. We cannot get the number of cluster k without any prior knowledge for real-world, thus we also select k by the prediction accuracy on a small subset of datasets.

Simulation Study

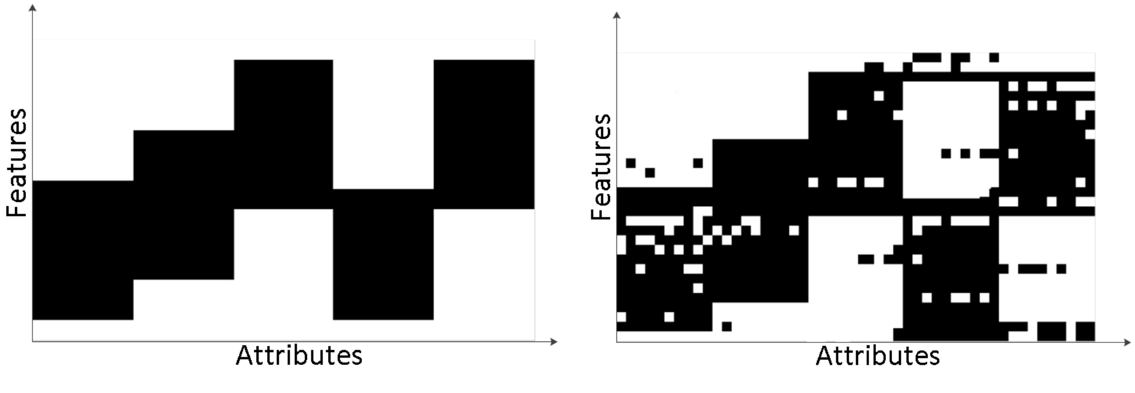
Since it is difficult to obtain the groundtruth cluster structure for real applications, we first verify the effectiveness of the proposed approach in obtaining the cluster structures on simulated dataset. Following [41, 98], we construct the synthetic data containing 5 clusters with 10 learning tasks in each cluster, generating a total number of 50 tasks. For the i-th task, a dataset $X_i \in \mathbb{R}^{d \times n}$ is randomly drawn from a normal distribution $N(0, 1)$ for learning, with the dimension $d = 30$ and the sample size

$n = 60$.

The projection model is constructed as follows. For the i-th cluster, we generate a cluster weight vector $\boldsymbol{w}_i^c \in \mathbb{R}^d$ drawn from the normal distribution $N(0, 900)$. Then 15 dimensions of $\boldsymbol{w}_i^c$ are randomly but carefully selected and assigned to zeros, to ensure all $\boldsymbol{w}^c$ are orthogonal to each other. Similarly, for the j-th task belonging to cluster i, we generate a task-specific weight vector $\boldsymbol{w}_j^s \in \mathbb{R}^d$ drawn from the normal distribution $N(0, 16)$ with the same dimensions of $\boldsymbol{w}_i^c$ assigned to zeros. Thus, the ultimate weight vector of the j-th task is the linear combination of the cluster and task-specific weight vector $\boldsymbol{w}_j = \boldsymbol{w}_i^c + \boldsymbol{w}_j^s$.

The corresponding response $\boldsymbol{y}_i$ of the i-th samples $\boldsymbol{x}_i$ of task j is then obtained by $\boldsymbol{y}_i = \boldsymbol{w}_j^T \boldsymbol{x}_i + \boldsymbol{\varepsilon}_i$ where $\boldsymbol{\varepsilon}$ is the noise vector drawn from $N(0, 0.1)$. We choose 0.5 as the threshold to assign binary label to each sample.

I verify the effectiveness of our proposed approach by comparing the learned cluster structure and the selected features with the groundtruth. Based on the prior knowledge implied by the construction of the groundtruth, we set $k = 5$ and the number of selected features as $K = 15$. Figure 4.3 shows one example of the learned projection matrix 4.3(b) with the comparison of the groundtruth 4.3(a) where the



(a) Groundtruth model (b) Learned model

Figure 4.3: The learned projection matrix and the corresponding groundtruth in the simulation experiments. The white parts are zeros and the black parts are non-zeros.

white part represents zeros and the black part represents non-zeros. The result shows that our approach is able to correctly capture the correct group sparse structures.

Feature Selection

Table 4.1: Classification results (ACC%±std) of different feature slection algorithm on different datasets. (the higher the better).

Algorithm	DataSet	Fisher	mRMR	Relief-F	Info-Gain	MTFS	Proposed
SVM	COIL100	60.66±3.54	55.72±3.34	62.80±2.56	62.00±2.84	78.77±2.35	**79.08±2.12**
	USPS	86.30±2.81	58.44±4.02	86.83±2.83	70.25±3.16	86.25±2.52	**93.15±2.18**
	Isolet	75.64±3.01	70.92±3.72	82.30±2.81	76.51±2.56	84.05±2.24	**87.06±1.98**
	YaleB	66.85±3.65	56.91±4.21	71.91±2.24	71.74±2.11	76.08±2.14	**78.17±2.18**
	ORL	46.50±4.21	84.51±2.32	67.18± 3.01	53.24±2.96	85.62±1.94	**90.51±1.78**
	PIX10P	93.56±2.01	90.45±3.32	96.00±1.77	92.01±1.97	96.81±1.54	**99.54±1.68**
*k*NN	COIL100	63.33±3.21	54.86±4.32	65.11±2.01	63.44±2.76	81.86±1.94	**82.48±1.68**
	USPS	89.39±2.11	59.17±3.72	89.61±2.01	74.70±2.76	90.44±1.54	**95.53±1.18**
	Isolet	75.38±2.45	57.56±3.42	79.87±2.21	73.71±2.42	77.01±2.14	**83.21±2.18**
	YaleB	69.17±3.24	58.41±3.72	65.53±2.81	65.37±2.42	77.08±2.45	**78.96±2.28**
	ORL	53.01±3.44	72.56±2.42	60.38±2.71	52.44±2.76	85.86±2.24	**88.10±2.10**
	PIX10P	94.56±1.91	86.45±2.22	96.00±1.81	86.04±2.04	97.81±1.54	**99.34±1.22**

I verify the feature selection capability on general multi-label datasets in this section. The experiment is conducted on 6 public benchmark feature selection datasets including one object image dataset **COIL100** [1], one hand-written digit image dataset **USPS** [39], one spoken letter speech dataset **Isolet** [29], three face image dataset **YaleB** [34], **ORL** [77] and **PIX10P** [1] . The statistics of the datasets are summarized in Table 4.2. we compare the proposed approach with the following representative feature selection algorithms: Fisher Score [25], mRMR [71], Relief-F [58], Information Gain [21], MTFS [4].

Following the common way to evaluate supervised feature selection, we assess the quality of selected features in terms of the classification performance [36, 15]. The

[1] PIX10P is publicly available from https://featureselection.asu.edu/datasets.php

Table 4.2: Statistics of the Feature Selection datasets

Dataset	# of Samples	# of Features	# of Classes
COIL100	7200	1024	100
YaleB	2414	1024	38
ORL	400	4096	40
PIX10P	100	10000	10
USPS	9298	256	10
Isolet	7797	617	150

larger classification accuracy is, the better performance the corresponding feature selection approach achieves. In our experiments, we employ linear Support Vector Machine (**SVM**) and k-nearest neighbors (k**NN**) classifier with $k = 3$ for evaluation. How to determine the optimal number of selected features is still an open question for feature selection; hence we vary the number of selected features as $\{10{,}30, 50 \ldots{,}90\}$ in this work. In each setup 50% samples are randomly selected for training and the remaining is for testing. Specific constrains are imposed to make sure the class labels of the training set are balanced. The whole experiment is conducted 10 rounds and average accuracies are reported.

Table 4.1 shows the comparison results for **SVM** and kNN on the 6 benchmark datasets when 50 features are selected. The result shows that MTFS and the proposed framework outperform Fisher Score, mRMR and Information Gain. The performance gain comes from that Fisher Score, mRMR and Information Gain select features one by one while MTFS and FSMC select features in a batch model. It is consistent with what was suggested in [87] that it is better to analyze features jointly for feature selection. Besides, most of the case, the proposed framework outperforms MTFS. Better performance gain is usually achieved when fewer number of features are selected. This performance gain suggests that modeling label correlation can significantly improve feature selection performance for multi-class data. Further statistical significance analysis can be applied which we leave for future work.

Attribute Prediction

I then compare our approach with state-of-the-art attribute learning work [19] (referred as MTAL) and [43] (referred as DSVA). Since MTAL is initially proposed for attribute ranking, we replace the original loss function with the one adopted in this chapter for fair comparison. DSVA requires attribute groups as prior, thus we run k-means offline to obtain the clusters for datasets do not have such information.

Table 4.3: Average prediction accuracies of all attributes on Seen and Unseen categories (the higher the better).

Datasets		MTAL	DSVA	Proposed
aPascal/aYahoo	Seen	0.5967±0.020	0.6105±0.018	**0.6363±0.014**
	Unseen	0.5663±0.022	0.5826±0.019	**0.6011±0.015**
AwA	Seen	0.5976±0.011	0.6053±0.015	**0.6254±0.007**
	Unseen	0.5587±0.012	0.5622±0.018	**0.5837±0.008**
SUN	Seen	0.6326±0.021	0.6469±0.025	**0.6682±0.011**
	Unseen	0.6020±0.022	0.6165±0.027	**0.6324±0.013**

The experiments are conducted on three benchmark datasets: aYahoo [31], Animals with Attributes (AwA) [54] and SUN attribute [70] and the statistics of the datasets are summarized in Table 4.4. To obtain a good representation of the high-level attributes, we require that the features can capture both the spatial and context information. Thus, we constructed the features by pooling a variety types of feature histograms including GIST, HoG, SSIM. For **aPascal/aYahoo** and **AwA** datasets we use predefined seen/unseen split published with the datasets. For **SUN** dataset, 60% of categories are randomly split out as “seen” categories in each round with the rest as “unseen” categories. During training 50% of samples are randomly and carefully drawn from each seen categories to ensure the balance of the positive and negative attribute labels. The rest samples from “seen” classes and all samples from “unseen” classes are used for testing.

Table 4.4: Statistics of Attribute Prediction Image Datasets.

Dataset	aPascal/aYahoo	AwA	SUN
# of images	15339	30475	14340
# of attributes	64	85	102
# of classes	32	50	611
# of features	2429	1200	1112

Table 4.3 shows the average prediction accuracy of each approach over all attributes by running the experiment 10 rounds. The result shows that for both “seen” and “unseen” categories, DSVA outperforms MTAL in prediction accuracy and our proposed approach further outperforms DSVA by 2%∼4%. DSVA decorrelates low-correlated attributes compared with MTAL thus achieves better prediction performance. However, the manually specified or off-line learned group structures are not able to achieve the optimal result. Our approach iteratively optimizes the clustering structure and the projection model, which achieves the best performance.

Zero-shot Learning

I also experiment on the zero-shot learning problem on all three datasets. Zero-shot learning aims to learn a classifier based on training samples from some seen categories, and classify some new samples to a new unseen category. We adopt the Direct Attribute Prediction (DAP) framework proposed in [54] with attribute prediction probability from each approaches as input. Since only continuous image level attribute labels are provided on the **SUN** dataset, we construct the class level attribute labels by thresholding the average attribute label values of all samples from the class. Same "Seen"\"Unseen" categories splits are adopted as previous experiments.

The Average classification accuracies of 10 rounds experiment are reported in Table 4.5. The result shows that on aYahoo and **AwA**, our approach achieves significant performance gains than the baseline approaches. The large number of categories in **SUN** dataset make the classification problem very hard which leads to all low performance of all approaches. Our approach still works better than the baseline approaches.

Table 4.5: Zero-shot learning accuracy on both real dataset.

	aYahoo	AwA	SUN
MTAL	0.1834	0.2953	0.1842
DSVA	0.2052	0.3085	0.2010
Proposed	**0.2262**	**0.3258**	**0.2133**

On Choosing the Parameters

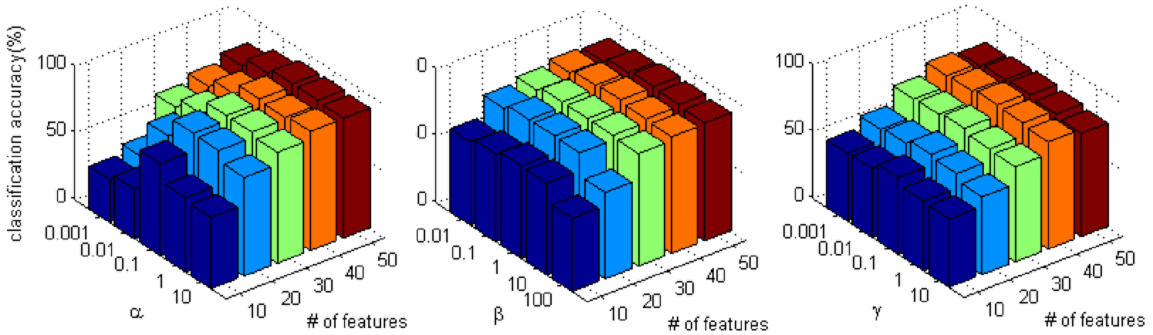


Figure 4.4: Parameter Analysis on SVM.

The proposed framework has three important parameters - α controlling the sparsity of W, β controlling the contribution of modeling label correlation and $gamma$

controls the global penalty. We study the effect of each parameter by fixing the other to see how the performance of the proposed approach varies with the number of selected features. Due to the page limitation, we only report the result on the **Isolet** dataset with SVM but we have similar observations in other datasets.

Figure 4.4 demonstrates the performance variance w.r.t. different parameters and the number of selected features. With the increase of β, the performance first increases, demonstrating the importance of modeling label correlation, and then decreases. This property is practically useful because we can use this pattern to set β. When α increases, the performance also increases dramatically, which suggests the capability of $\ell_{2,1}$-norm for feature selection. The performance also increases with γ and then decrease, but relatively stable. The best performance is achieved around 0.1.

4.3 Embedded Supervised Feature Selection for Multi-label Data

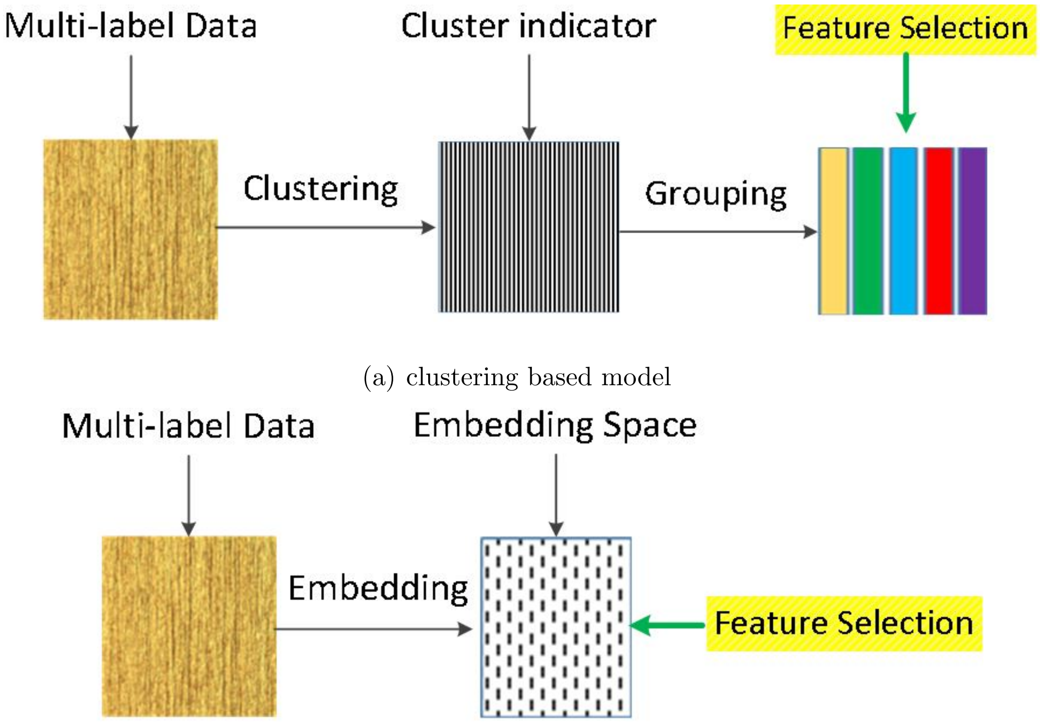
In the approach I discussed in last section, as shown in Figure 4.5(a), given a multi-label data, I first estimate a cluster indicator, and then the attributes are partitioned into different groups. A feature selection framework is further imposed on the grouped attributes to select appropriate features. In this framework, the feature selection quality is highly depend on the estimated group structure, which is very sensitive to the noise. What's more, without prior knowledge the optimal cluster structure is not exclusive, which makes the feature selection framework not very robust.

In this section, I propose a novel embedded supervised feature selection framework for multi-label data. Different from the previous approach, as shown in Figure 4.5(b), given a multi-label dataset, we directly embedded the original data into a new embedding space which captures the correlation among different attributes. Then a sparse based regularization is imposed on the embedding space for joint feature selection.

4.3.1 Proposed Framework

Let $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{d\times n}$ denotes the multi-class dataset with n samples and d features $\mathcal{F} = \{f_1, f_2, \ldots, f_d\}$, let $Y = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n] \in \{0,1\}^{m\times n}$ denotes the corresponding label matrix of m classes $\mathcal{C} = \{c_1, c_2, \ldots, c_m\}$, we aim to select $K(K{\le}d)$ most relevant features from $\mathcal{F}$ by leveraging X, Y and the label correlation in $\mathcal{C}$.

Let $\mathbf{s} = \pi(\overbrace{0,\ldots,0}^{d-K}, \overbrace{1,\ldots,1}^{K})$, where $\pi(\cdot)$ is the permutation function and K is the number of features to select where $\mathbf{s}_i = 1$ indicates that the i-th feature is



(a) clustering based model

(b) Learned model

Figure 4.5: The demonstration of the proposed clustering based feature selection and the embedded supervised feature selection.

selected. The original data can be represented as diag($\mathbf{s}$)$\mathbf{X}$ with K selected features where diag($\mathbf{s}$) is a diagonal matrix. We assume that a linear projection matrix $W = [\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_m] \in \mathbb{R}^{d\times m}$ maps the data X to its label matrix Y where $\boldsymbol{w}_i \in \mathbb{R}^d (i = 1, 2, \ldots, m)$ is the projection vector for the i-th class c_i. If we do not consider label correlation, we can select K features via solving the following optimization problem:

$$\begin{aligned} \arg\min_{W,\mathbf{s}} \quad & L(W^T \text{diag}(\mathbf{s})X, Y) \\ s.t., \quad & \mathbf{s} \in \{0,1\}^d,\ \mathbf{s}^T \mathbf{1}_d = K \end{aligned} \tag{4.21}$$

where $L(\cdot)$ is the loss function and typical choices of loss functions include least square and logistic regression.

Modeling Attributes Correlation

In real-world multi-class applications, class labels might be correlated and they may form several clusters (or groups) [43]. Therefore we can model label correlation via learning clustering structures of labels. A partition of the projection matrix W into k clusters can be formed under the non-negative matrix factorization framework as:

$$\begin{aligned} &\arg\min_{U,V} \|W - UV^\top\|_F^2 \\ &s.t. \quad V \in \{0,1\}^{m\times k}, V^\top \mathbf{1} = \mathbf{1} \end{aligned} \tag{4.22}$$

where $U \in \mathbb{R}^{d\times k}$ is the latent feature matrix and $V \in \mathbb{R}^{m\times k}$ is the cluster indicator. The problem in Eq. 4.22 is difficult to solve due to the constraint on V. Thus we relax the constraint on the label indicator matrix V to orthogonality following [61]. After the relaxation, Eq. 4.22 can be rewritten as:

$$\begin{aligned} &\arg\min_{U,V} \|W - UV^\top\|_F^2 \\ &s.t. \quad V^\top V = I, V{\geq}0 \end{aligned} \tag{4.23}$$

To capture the correlation among labels, similar labels should been partitioned into the same group. Inspired by the spectral analysis [61], we further add the following term to force similar label are clustered in the same group:

$$\min \mathbf{Tr}(V^\top LV) \tag{4.24}$$

where $L = D - S$ is the Laplacian matrix and D is a diagonal matrix with its elements defined as $D_{ii} = \sum_{j=1}^{m} S_{ij}$. $S \in \mathbb{R}^{m\times m}$ denotes the similarity matrix based on W, which is obtained through RBF kernel as

$$S_{ij} = e^{-\frac{\|\boldsymbol{w}_i - \boldsymbol{w}_j\|^2}{\sigma^2}}$$

Combing Eq. (4.23) and (4.24), the label correlation can be modeled as

$$\begin{aligned} &\arg\min_{U,V} \|W - UV^\top\|_F^2 + \beta\mathbf{Tr}(V^\top LV) \\ &s.t. \quad V^\top V = I, V \geq 0 \end{aligned} \tag{4.25}$$

Feature Selection

With the model component to capture label correlation in Eq. (4.25), the proposed embedded supervised feature selection framework (ESFS) for multi-class data is to solve the following optimization problem:

$$\begin{aligned} \arg\min_{W,F,\mathbf{s}} \; & L(W^T \mathrm{diag}(\mathbf{s})X, Y) + \alpha\|W - UV^\top\|_F^2 + \beta\mathbf{Tr}(V^\top LV) \\ s.t. \qquad & V^\top V = I, V \geq 0 \\ & \mathbf{s} \in \{0,1\}^d,\ \mathbf{s}^T\mathbf{1}_d = K \end{aligned} \tag{4.26}$$

According to [92], the feature selection on W is equivalent to perform feature selection on U, thus Eq. (4.26) is equivalent to the following optimization problem:

$$\begin{aligned} \arg\min_{W,F,\mathbf{s}} & \; L(W^\top X, Y) + \alpha\|W - U\text{diag}(\mathbf{s})V^\top\|_F^2 + \beta\mathbf{Tr}(V^\top LV) \\ s.t. & \quad V^\top V = I, V{\geq}0 \\ & \quad \mathbf{s} \in \{0,1\}^d,\; \mathbf{s}^T\mathbf{1}_d = K \end{aligned} \tag{4.27}$$

The constraint on $\mathbf{s}$ makes Eq. (4.27) a mixed integer programming problem, which is difficult to solve. We observe that diag($\mathbf{s}$) and U is as the form of Udiag($\mathbf{s}$). Since $\mathbf{s}$ is a binary vector and $d - K$ rows of the diag($\mathbf{s}$) are all zeros, Udiag($\mathbf{s}$) is a matrix where the elements of many rows are all zeros. This motivates us to absorb diag($\mathbf{s}$) into U as $W = U$diag($\mathbf{s}$), and add $\ell_{2,1}$-norm on U to ensure the sparsity of U in rows and achieve feature selection. With this relaxation, Eq. (4.27) can be rewritten as:

$$\begin{aligned} \arg\min_{W,U,V} & \; L(W^\top X, Y) + \alpha\|W - UV^\top\|_F^2 \\ & + \beta\mathbf{Tr}(V^\top LV) + \gamma\|U\|_{2,1} \\ s.t. & \quad V^\top V = I, V{\geq}0 \end{aligned} \tag{4.28}$$

Since U is forced to sparse where some rows are close to $\mathbf{0}$, some instances of W that poorly reconstruct from U and V. These instances from the decomposition regularizer would dominate the objective function because of the squared errors. To make the model robust to these instances, we replace the decomposition regularizer by $\ell_{2,1}$-norm. Without loss of the generality, we adopt the traditional least square loss for demonstration in the following. The objective function of the proposed framework becomes

$$\begin{aligned} \arg\min_{W,U,V} & \; \|W^\top X - Y\|_F^2 + \alpha\|W - UV^\top\|_{2,1} \\ & + \beta\mathbf{Tr}(V^\top LV) + \gamma\|U\|_{2,1} \\ s.t. & \quad V^\top V = I, V{\geq}0 \end{aligned} \tag{4.29}$$

4.3.2 Algorithm

We first introduce the optimization algorithm and then give an analysis of the proposed algorithm.

Optimization

The objective function in Eq. (4.29) is convex if we update the variables U, V and W alternatively. Following [38], we use Alternating Direction Method of Multiplier (ADMM) [11] to optimize the objective function. By introducing two auxiliary variables $E = W - UV^\top$ and $Z = V$, we can convert Eq. (4.29) into the following equivalent problem:

$$\begin{aligned} \arg\min_{W,U,V,E,Z} & \|W^\top X - Y\|_F^2 + \alpha\|E\|_{2,1} \\ & + \beta\mathbf{Tr}(V^\top LV) + \gamma\|U\|_{2,1} \\ s.t. \;\; & E = W - UV^\top, Z = V, V^\top V = I, Z{\geq}0 \end{aligned} \tag{4.30}$$

which is equivalent to solve the following ADMM problem

$$\begin{aligned} \arg\min_{W,U,V,E,Z,Y_1,Y_2,\mu} & \|W^\top X - Y\|_F^2 + \alpha\|E\|_{2,1} \\ & + \beta\mathbf{Tr}(Z^\top LV) + \gamma\|U\|_{2,1} \\ & + \mathbf{Tr}(Y_1^\top(Z - V)) + \mathbf{Tr}(Y_2^\top(W - UV^\top - E)) \\ & + \frac{\mu}{2}(\|Z - V\|_F^2 + \|W - UV^\top - E\|_F^2) \\ s.t. \;\; & V^\top V = I, Z{\geq}0 \end{aligned} \tag{4.31}$$

where Y_1, Y_2 are two Lagrangian multipliers and μ is a scalar to control the penalty for the violation of equality constraints $E = W - UV^\top$ and $Z = V$.

Update E By fixing the other variables except E, Eq. (4.31) can be reformed as follows by removing terms that are irrelevant to E:

$$\arg\min_{E} \frac{1}{2}\|E - (W - UV^\top + \frac{1}{\mu}Y_2)\|_F^2 + \frac{\alpha}{\mu}\|E\|_{2,1} \tag{4.32}$$

This problem can be solved according to the following lemma due to [12].

Lemma 2. *Let $Q = [\boldsymbol{q}_1; \boldsymbol{q}_2; \cdots; \boldsymbol{q}_m]$ be a given matrix and λ a positive scalar. If the optimal solution of*

$$\arg\min_{W} \frac{1}{2}\|W - Q\|_F^2 + \lambda\|W\|_{2,1}$$

is W^*, then the i-th row of W^* is

$$\boldsymbol{w}_i^* = \begin{cases} (1 - \frac{\lambda}{\|\boldsymbol{q}_i\|})\boldsymbol{q}_i, & if \|\boldsymbol{q}_i\| > \lambda \\ 0, & otherwise \end{cases} \tag{4.33}$$

Thus, let $Q = W - UV^\top + \frac{1}{\mu}Y_2$, E can be updated as follows according Lemma 2:

$$\boldsymbol{e}_i = \begin{cases} (1 - \frac{\alpha}{\mu\|\boldsymbol{q}_i\|})\boldsymbol{q}_i, & if \|\boldsymbol{q}_i\| > \frac{\alpha}{\mu} \\ 0, & otherwise \end{cases} \tag{4.34}$$

Update U By fixing the other variables except U and remove terms that are irrelevant to U, Eq. (4.31) becomes

$$\arg\min_{U} \frac{\mu}{2}\|W - UV^\top - E + \frac{1}{\mu}Y_2\|_F^2 + \gamma\|U\|_{2,1} \tag{4.35}$$

Since $V^\top V = I$, Eq. (4.35) can be rewritten as

$$\arg\min_{U} \frac{1}{2}\|U - (W - E + \frac{1}{\mu}Y_2)V\|_F^2 + \frac{\gamma}{\mu}\|U\|_{2,1}$$

According to Lemma 2, let $K = (W - E + \frac{1}{\mu}Y_2)V$, U can be updated as

$$\boldsymbol{u}_i = \begin{cases} (1 - \frac{\gamma}{\mu\|\boldsymbol{k}_i\|})\boldsymbol{k}_i, & if \|\boldsymbol{k}_i\| > \frac{\gamma}{\mu} \\ 0, & otherwise \end{cases} \tag{4.36}$$

Update Z Similarly, to update Z, we fix the other variables and remove terms irrelevant to Z, which makes Eq. (4.31) become

$$\arg\min_{Z;Z\geq 0} \frac{\mu}{2}\|Z-V\|_F^2 + \beta\mathbf{Tr}(Z^\top LV) + \mathbf{Tr}(Y_1^\top(Z-V)) \tag{4.37}$$

Let $T = V - \frac{1}{\mu}Y_1 - \frac{\beta}{\mu}LV$, Eq. (4.37) can be written as

$$\arg\min_{Z;Z\geq 0} \|Z - (V - \frac{1}{\mu}Y_1 - \frac{\beta}{\mu}LV)\|_F^2$$

This is equivalent to the following element-wise optimization problem

$$\arg\min_{Z_{ij};Z_{ij}\geq 0} \|Z_{ij} - T_{ij}\|^2$$

where the optimal solution is achieved by

$$Z_{ij} = \max(T_{ij}, 0)$$

Update V By removing terms irrelevant to V and fixing other variables, Eq. (4.31) becomes

$$\begin{aligned}
&\arg\min_{V;V^\top V=I} \mathbf{Tr}(Y_1^\top(Z-V)) + \beta\mathbf{Tr}(Z^\top LV) \\
&+ \frac{\mu}{2}(\|Z-V\|_F^2 + \|W - UV^\top - E\|_F^2) \\
&+ \mathbf{Tr}(Y_2^\top(W - UV^\top - E))
\end{aligned} \tag{4.38}$$

Let

$$N = \frac{1}{\mu}Y_1 + Z - \frac{\beta}{\mu}L^\top Z + (W - E + \frac{1}{\mu}Y_2)^\top U, \tag{4.39}$$

utilizing $V^\top V = I$, Eq. (4.38) can be further written as

$$\arg\min_{V,V^\top V=I} \|V - N\|_F^2 \tag{4.40}$$

This problem can be solved according to the following lemma due to [38]:

Lemma 3. *Let P and Q are the left and right singular vectors of the economic singular value decomposition (SVD) of N where $N = P\Sigma Q$, the optimal V of the objective function in Eq. (4.40) is defined as*

$$V = PQ^\top$$

Update W Removing terms irrelevant to W and fixing other variables, we rewrite Eq. (4.31) as

$$\begin{aligned} F(W) = \arg\min_{W} \|W^\top X - Y\|_F^2 + \mathbf{Tr}(Y_2^\top(W - UV^\top - E)) \\ + \frac{\mu}{2}\|W - UV^\top - E\|_F^2 + \beta\mathbf{Tr}(Z^\top LV) \end{aligned} \tag{4.41}$$

Let $\tilde{U} = UV^\top - E$ and $\tilde{V} = VZ^\top$, the gradient of 4.41 corresponding to $\boldsymbol{w}_i$ can be represented as

$$\begin{aligned} \nabla\frac{F(W)}{\boldsymbol{w}_i} &= \boldsymbol{y}_{2,i} + \mu(\boldsymbol{w}_i - \tilde{\boldsymbol{u}}_i) + \sum_{j=1}^{n}(2\boldsymbol{x}_j(\boldsymbol{x}_j^\top)\boldsymbol{w}_i - y_{ij}) \\ &+ \beta \sum_{j=1;j\neq i}^{m} ((\tilde{v}_{ii} - \tilde{v}_{ij} - \tilde{v}_{ji})\frac{2(\boldsymbol{w}_j - \boldsymbol{w}_i)}{\sigma^2}e^{-\frac{\|\boldsymbol{w}_i - \boldsymbol{w}_j\|^2}{\sigma^2}}) \end{aligned}$$

where $\boldsymbol{y}_{2,i}$, $\boldsymbol{w}_i$, $\tilde{\boldsymbol{u}}_i$ and $\boldsymbol{x}_i$ are the i-th column vector of matrix Y_2, W, $\tilde{U}$ and X; y_{ij} and $\tilde{v}_{ij}$ are the (i,j)-th element of matrix Y and $\tilde{V}$.

Thus, the new weight vector $\boldsymbol{w}_i'$ can be updated by gradient descent

$$\boldsymbol{w}_i' = \boldsymbol{w}_i - \eta\nabla\frac{F(W)}{\boldsymbol{w}_i}$$

where η is the step size.

Update Y_1**,**Y_2 **and** μ According to [11], the ADMM parameters can be updated as

$$\begin{aligned} Y_1 &= Y_1 + \mu(Z - V) \\ Y_2 &= Y_2 + \mu(W - UV^\top - E) \\ \mu &= \min(\rho\mu, \mu_{max}) \end{aligned} \tag{4.42}$$

Algorithm 5 The Proposed Feature Selection Framework for Multi-class Data

```
Input:
    1. Multi-class data $\{X, Y\}$;
    2. Parameters $\alpha$, $\beta$, $\gamma$, $k$ and the number of selected features $K$;
    3. The initial projection matrix $W_0$;
Output:
    Top-$K$ selected features;
Initialize $W = W_0$, $\mu = 10^{-3}$, $\rho = 1.1$, $\mu_{max} = 10^{10}$, $U = 0$, $V = 0$ (or initialized by $K$-means);
repeat
    Calculate $Q = W - UV^\top + \frac{1}{\mu}Y_2$ and update $E$ according to Eq. (4.34);
    Calculate $K = (W - E + \frac{1}{\mu}Y_2)V$ and update $U$ according to Eq. (4.36);
    Calculate $T = V - \frac{1}{\mu}Y_1 - \frac{\beta}{\mu}LV$ and update $Z$ according to Eq. (4.38);
    Calculate $N$ according to Eq. 4.39 and update $V$ according to Lemma 3;
    Update $Y_1$,$Y_2$,$\mu$ according to Eq. (4.42);
until Converges
Sort each feature according to $\|\boldsymbol{v}^i\|_2$ in descending order;
return The top-$K$ ranked features;
```

where $\rho > 1$ is a parameter controlling the convergence speed and μ_{max} is a large number preventing μ becomes too large.

Following these updating rules, the proposed algorithm is summarized in Algorithm 5. The importance of the i-th feature is indicated by $\|\boldsymbol{u}^i\|_2$. Therefore, we rank features in descending order according to $\|\boldsymbol{u}^i\|_2$ and select the top-K ranked ones.

Algorithm Analysis

Since we adopt ADMM as the optimization algorithm, the convergence is guaranteed due to the proof in [11]. The convergence criteria can be set as $\frac{J_{t+1}-J_t}{J_t} < \epsilon$ where J_t is the objective function in Eq. (4.29) and ϵ is a tolerance value. In our implementation we control the iteration by setting a maximun number of iteration, e.g., 100 in our experiment.

For the time complexity, the update of E and U involves the computation of Q and K. Since U is sparse, the computation cost is $O(Nd)$. The main computation cost during updating Z is the calculation of T, which is $O(k^2)$. The computation cost for updating V involves the computation of N and the SVD decompostion, which is $O(Ndk)$ and $O(Nk^2)$. The computation cost for update W mainly includes matrix multiplication and matrix inverse whose total time complexity is $O(ndk)$. The computational cost for Y_1 and Y_2 are both $O(Nd)$. Since $d \gg k$, the final computation cost is $O(Ndk)$ for each iteration.

4.3.3 Experiments

In this section we conduct experiments to evaluate the effectiveness of our proposed framework. We first describe the experiment of a simulated dataset to verify the effectiveness of the proposed framework in finding the cluster structure of class labels. Then we focus on the empirical evaluation by introducing the public datasets involved in our experiments and the baseline approaches we compared with followed by the experiment results and parameter analysis.

Experiment using Simulated Data

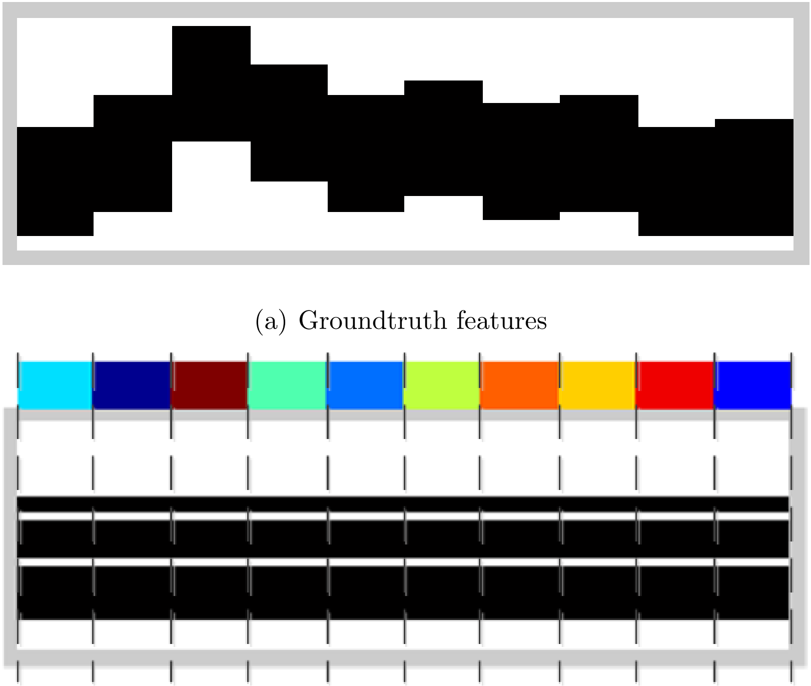
Since it is difficult to obtain the groundtruth cluster structure for real applications, we first verify the effectiveness of the proposed approach in obtaining the cluster structures of the proposed approach on simulated dataset. Following [41, 98], we construct the synthetic data containing 10 clusters with 10 class labels in each cluster, generating a total number of 100 class labels. For the i-th class label, a dataset $X_i \in \mathbb{R}^{d \times n}$ is randomly drawn from a normal distribution $N(0, 1)$ for learning, with the dimension $d = 30$ and the sample size $n = 60$.

We construct the projection model as follows. For the i-th cluster, a cluster weight vector $\boldsymbol{w}_i^c \in \mathbb{R}^d$ is drawn from the normal distribution $N(0, 900)$. Then 15 dimensions of $\boldsymbol{w}_i^c$ are randomly but carefully selected and assigned as zeros, to ensure all $\boldsymbol{w}^c$ are orthogonal to each other. Similarly, for the j-th class label belonging to cluster i, a class-specific weight vector $\boldsymbol{w}_j^s \in \mathbb{R}^d$ is drawn from the normal distribution $N(0, 16)$ with the same dimensions of $\boldsymbol{w}_i^c$ assigned to zeros. Thus, the ultimate weight vector of the j-th class label is the linear combination of the cluster and class-specific weight vector $\boldsymbol{w}_j = \boldsymbol{w}_i^c + \boldsymbol{w}_j^s$.

The corresponding response $\boldsymbol{y}_i$ of the i-th samples $\boldsymbol{x}_i$ in the class j is then obtained by $\boldsymbol{y}_i = \boldsymbol{w}_j^T \boldsymbol{x}_i + \boldsymbol{\varepsilon}_i$ where $\boldsymbol{\varepsilon}$ is the noise vector drawn from $N(0, 0.1)$. We choose 0.5 as the threshold to assign binary label to each sample.

We verify the effectiveness of our proposed approach by comparing the learned cluster structure and the selected features with the groundtruth. Based on the prior knowledge implied by the construction of the groundtruth, We set $k = 10$ and the number of selected features as $K = 15$.

Figure 4.6 demonstrates the detected cluster structures and selected features by our approach (4.6(b)) and the corresponding groundtruth features (4.6(a)). The results show that our approach can detect the correct cluster structures of the class labels, and select important features which exist in majority of labels. For example, for the majority of class labels in Figure 4.6(a), the first and the last several features are important, which are correctly selected by the proposed approach as shown in



(a) Groundtruth features

(b) Selected features

Figure 4.6: The selected features of our approach and the corresponding groundtruth features in the simulation experiment. The horizontal coordinates denote class labels while the vertical coordinates denote features. The colored bins represent the detected cluster structure corresponding to the class labels, where each color denotes an independent cluster. The black parts are zeros and the white parts are non-zeros.

Figure 4.6(b).

Experiment using Real Data

The real data experiment is conducted on 6 public benchmark feature selection datasets including one object image dataset, i.e., **COIL100** [1], one hand written digit image dataset **USPS** [39], one spoken letter speech dataset **Isolet** [29], three face image dataset **YaleB** [34], **ORL** [77] and **PIX10P** [2] . All the datasets are standardized to zero-mean and normalized by the standard deviation. The statistics of the datasets are summarized in Table 4.6.

[2] PIX10P is publicly available from https://featureselection.asu.edu/datasets.php

Table 4.6: Statistics of the Dataset

Dataset	# of Samples	# of Features	# of Classes
COIL100	7200	1024	100
YaleB	2414	1024	38
ORL	400	4096	40
PIX10P	100	10000	10
USPS	9298	256	10
Isolet	7797	617	150

Following the common way to evaluate supervised feature selection, we assess the quality of selected features in terms of the classification performance [36, 15]. The larger classification accuracy is, the better performance the corresponding feature selection approach achieves. In our experiments, we employ linear Support Vector Machine (**SVM**) and k-nearest neighbors (k**NN**) classifier with $k = 3$ for evaluation. How to determine the optimal number of selected features is still an open question for feature selection; hence we vary the number of selected features as $\{10,20,\ldots,50\}$ in this work. In each setup 50% samples are randomly selected for feature selection and training for classification and the remaining is for testing. Specific constrains are imposed to make sure the class labels of the training set are balanced. The whole experiment is conducted 10 rounds and average accuracies are reported.

We compare the proposed approach with the following representative feature selection algorithms:

- Fisher Score [25] determines the most relevant features with the best discriminating ability on fully labeled training data.
- mRMR [71] selects features that correlate the strongest with a classification variable and makes the features mutually different from each other.

- Relief-F [58] chooses instances randomly and update the weight of the feature relevance based on the nearest neighbors.

- Information Gain [21] selects features by computing information gain.

- MTFS [4] applies a $\ell_{2,1}$ norm on the column space of W to constrain a sparse structure for feature selection.

For the parameter setup, we tune the parameters for all methods by cross-validation for a fair comparison. We will further discuss some key parameters of the proposed framework in the following subsection.

Figure 4.7 shows ihe comparison results for **SVM** on the 6 benchmark datasets and we make the following observations:

- MTFS and the proposed framework ESFS outperform Fisher Score, mRMR and Information Gain. For example, the proposed framework achieves a performance gain of 6%∼15% compared with the traditional approaches. Fisher Score, mRMR and Information Gain select features one by one while MTFS and ESFS select features in a batch model. It is consistent with what was suggested in [87] that it is better to analyze features jointly for feature selection.

- Most of the time, the proposed framework ESFS outperforms MTFS. Better performance gain is usually achieved when fewer number of features are selected. For example, ESFS obtains about 10% relative improvement over MTFS in the USPS dataset when 10 features are selected. This performance gain suggests that modeling label correlation can significantly improve feature selection performance for multi-class data.

We make similar observations for k**NN** compared to **SVM**. Due to the page limitation, we leave the comparison results in the final extended version.

On Choosing the Parameters

The proposed framework has three important parameters - α and β controlling the contribution of modeling label correlation and γ controlling the sparsity of W We study the effect of each parameter by fixing the other to see how the performance of ESFS varies with the number of selected features. Due to the page limitation, we only report the result on the **Isolet** dataset in Figure 4.8. However, we have similar observations in other datasets.

Figure 4.8 demonstrates the experiment result of how the classification accuracies varies with the increase of parameters. With the increase of $alpha$ and β, the performance first increases, demonstrating the importance of modeling label correlation, and then decreases. This property is practically useful because we can use this pattern to set these parameters. When γ increases, the performance increases dramatically, which suggests the capability of $\ell_{2,1}$-norm for feature selection.

4.4 Conclusions

In this chapter, we proposed a clustering-base multi-task joint feature selection framework and an embedded supervised feature selection framework for multi-label semantic attribute prediction. Our approach employs both clustering and group-sparsity regularizers for feature selection. The clustering regularizer partitions the attributes into different groups where strong correlation lies among attributes in the same group while weak correlation exists between groups. The group-sparsity regularizer encourages intra-group feature-sharing and inter-group feature competition. With an efficient alternating optimization algorithm, the proposed approach is able to obtain a good group structure and select appropriate features to represent semantic attributes. The proposed approach was verified on both synthetic and real-world benchmark datasets with comparison with state-of-the-art approaches. The result shows effective group structure identification capability of our method, as well as its significant performance gains on feature selection, attribute prediction and zero-shot learning.

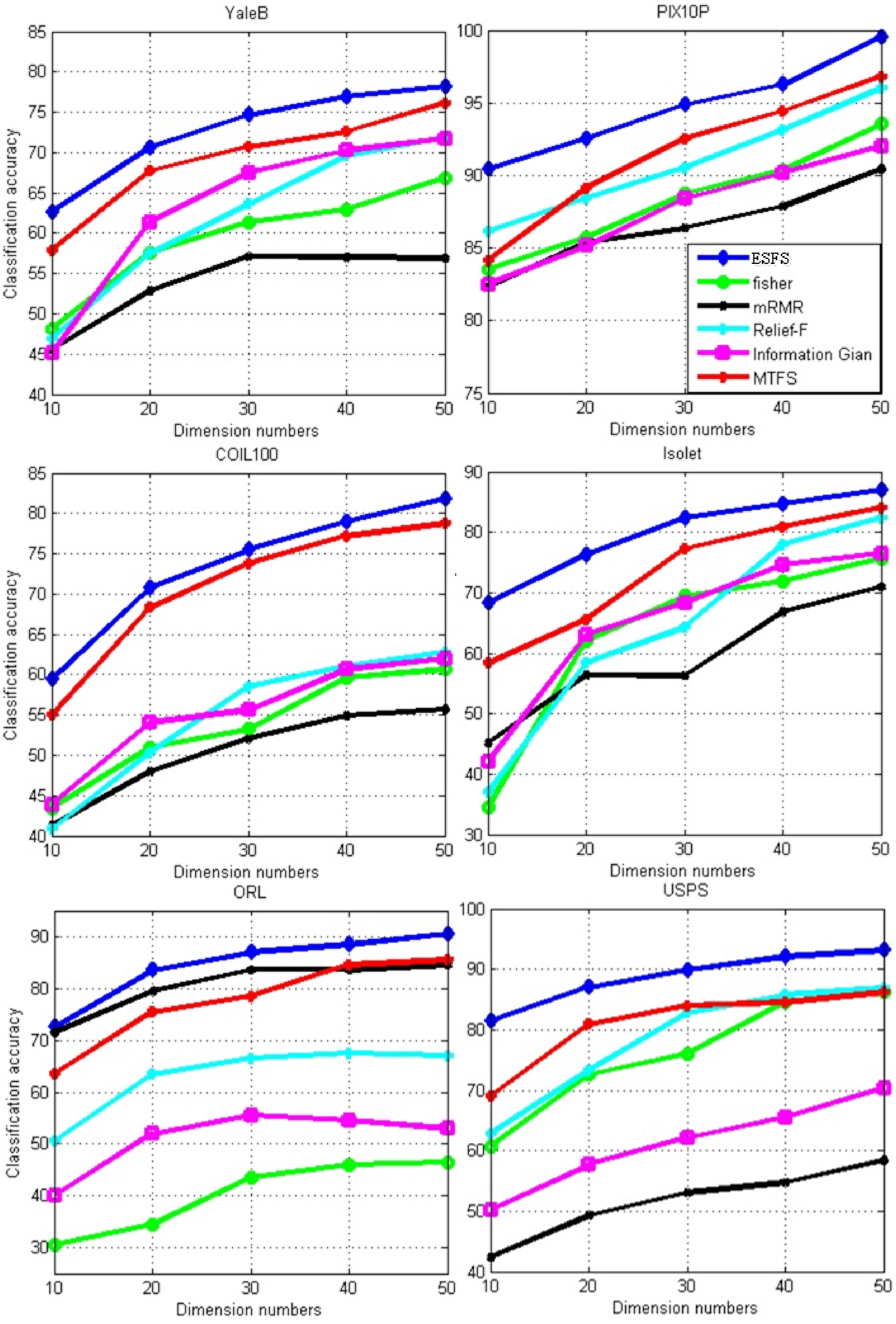


Figure 4.7: Classification accuracies with different dimensions of features selected of SVM.

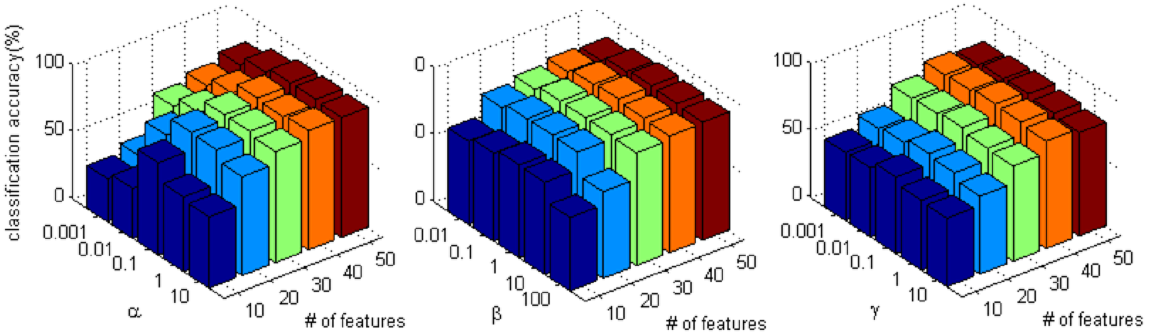


Figure 4.8: Parameter Analysis for the Proposed Framework

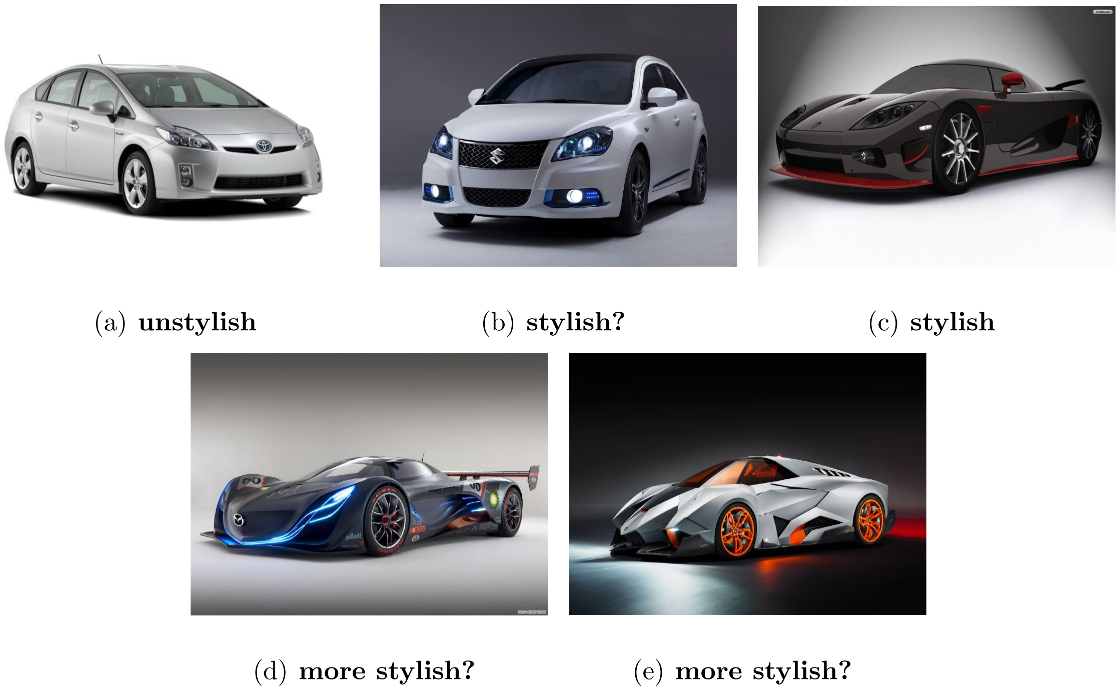
Chapter 5

ATTRIBUTE LEANING BASED ON POINTWISE AND PAIRWISE LABEL FUSION

Since pointwise and pairwise data have different benefits and limitations for learning, in this chapter I propose the work to fuse both pointwise and pairwise labels for improved attribute learning.

5.1 Introduction

The social media era has witnessed phenomenal growth of user-generated images on the Internet. The ever-growing number of images has brought about new chal-



(a) **unstylish** (b) **stylish?** (c) **stylish**

(d) **more stylish?** (e) **more stylish?**

Figure 5.1: stylish and unstylish cars. Considering pointwise label, to most people (a) would be unstylish and (c) would be considered stylish. However, it is ambiguous to classify (b) to be stylish or unstylish. Considering pairwise label, people may have different preference to compare whether (d) or (e) is more stylish.

lenges for efficient image retrieval, and in turn, for applications that rely on image retrieval. Conventional content-based image retrieval approaches learn some general ranking models purely based on the underlying images. In recent years, adaptive image retrieval [26, 50, 78] has emerged as a new trend, which intends to satisfy a user's specific requirements or preference. For example, in search of art images, some people like realism paintings while some may prefer abstract art. A retrieval engine being able to support such personalization would have the best potential to deliver what a user is really looking for. In practice, it is still difficult to learn a well generalized model due to the lack of user-adaptive training data. For example, in applications like on-line shopping, it is unreasonable to assume a user's personal preference data have been made available *a priori* for training the system. Often, desired is an on-line learning approach that accumulates such information over time interactively. My approach adopts a model adaptation strategy and proposes a new ranking model and an online learning algorithm. Such a method is especially proper for applications that utilize interactive input/feedback of a user in achieving adaptive image retrieval/recommendation.

Beneath the above challenge of personalization lie the fundamental problems of **semantic gap** and **intent gap** in general image retrieval. The semantic gap refers to the discrepancy between extractable low-level image features and high-level semantic concepts of images, while the intent gap refers to inadequacy of the representation of a query in expressing a user's true intent. In recent years, towards bridging the semantic gap, methods exploiting semantic attributes of visual objects have attracted significant attention in applications including object recognition [93, 54, 31], face verification [52] and image search [90, 53, 67]. Instead of using low-level features, these approaches describe images by high-level, human-nameable visual attributes, such as keep hair color, presence of beard or mustache, presence of eyeglasses, etc., to describe human faces.

In the meantime, towards bridging the intent gap, learning-to-rank approaches have been proposed. Recent literature on this regard includes three types of approaches distinguished by how the training data are used: pointwise, pairwise and listwise approaches. The first two types of approaches have been adopted in image ranking problems. Pointwise approaches [56, 82] adopt category labels in the training samples to learn a ranking function. For example, to describe the car images in Fig. 5.1, the car in Fig. 5.1(c) is categorized as a “stylish” car and the one in Fig. 5.1(a) is labelled “unstylish”. In a different way, pairwise approaches [13, 89, 33, 46, 16] learn a ranking function by taking comparative sample pairs for training. For example, most people would agree that the car in Fig. 5.1(c) looks more “stylish” than the one in Fig. 5.1(a). Such pair of samples with relative labels can be used to learn ranking functions for processing new images.

Pointwise data and pairwise data have different advantages and limitations in terms of data availability, labelling complexity and representational capability, as elaborated below.

Data availability In practical applications pointwise and pairwise labels are not always available for every data sample, especially considering the subjectivity of the labels. For example, in pointwise labelling, most people would agree that Fig. 5.1(c) is a “stylish” car and Fig. 5.1(a) is an “unstylish” car, but it would be difficult to tell whether Fig. 5.1(b) is a “stylish” car. Some people may think it is “stylish” because of the design of the headlights, while some others may deem the body design unattractive. Similarly, ambiguity also exists in pairwise data labelling. For example, comparing Fig. 5.1(d) and 5.1(e), people may have different opinions on which car is more “stylish” because of subjective preference. When ambiguity exists, it is better not to allow the data to be labeled so as not to produce noisy labels.

Labelling complexity In general, pairwise data may be more expensive to label. For example, given 10 images, we only need to label 10 samples to assign each image into one category. Also, category labels can be acquired from other sources such as image tags. On the other hand, to assign pairwise labels for all 10 images, we would need to compare 45 pairs to completely capture the ranking information. (Although the relative relation is transferable such as $(A \succ B) \& (B \succ C) \Rightarrow A \succ C$, it is difficult to discover those "key pairs" since we usually have to randomly pick pairs without any prior knowledge.) we note, however, that sometimes it is easier for a user to assign pairwise labels through comparison than having to give a pointwise label for a given image.

Representational capability Pairwise data tend to have stronger representational capability than pointwise data in ranking problems, as pointwise label only implies the relative order of data samples from different categories but not those from the same category. In contrast, pairwise labels already give the relative order of every training pairs, and thus contain more knowledge to learn a better ranking model.

As pointwise and pairwise labels encapsulate information of different types/amounts and may have different availability, we set out to develop a new framework for fusing both types of training data for improved ranking performance. Most of current fusion approaches [64, 73] only use pointwise labels and the fusion only appears in the cost function. To my best knowledge, the only work considering fusing pointwise and pairwise data is presented by Sculley [80] whose object function is simply a linear combination of loss functions from regression and ranking.

In this Chapter, towards supporting adaptive image retrieval, I propose a new ranking-based framework. My approach uses visual attributes to describe images, which helps to partially overcome the semantic gap problem. To alleviate the problem of lacking adaptive training samples, my approach attempt to maximize the utilization of all available training data by fusing both pointwise and pairwise labelled data in training. Compared to [80], my approach is formulated as a soft margined SVM which is able to achieve better generalization performance. Furthermore, to support interactivity, which is one natural way of gathering adaptive training data on the fly, we derive an online learning algorithm which can incrementally acquire a user's online feedback to improve the performance of the model incrementally with additional amount of data. As will be demonstrated by experiments, the proposed framework is able to take advantage of both types of data and deliver better performance than the baseline approaches that use only one type of data for learning.

There are three key contributions of my work in this Chapter. (1) I propose a new ranking framework termed "hybrid ranking" which takes both pointwise and pairwise labelled data for learning. (2) I propose an online learning algorithm for my proposed hybrid ranking framework, which can better support applications like adaptive image ranking. (3) I derive my hybrid ranking framework into a kernel form so that different kernel functions (depending on the application) may be applied for better performance.

To summary, the problems I aim to solve can be defined as follows:

Adaptive Image Retrieval We consider the following adaptive image retrieval procedure illustrated in Fig. 5.2: Given a training dataset including both the attribute existence labels of images (pointwise) and the relative attribute strength labels of image pairs (pairwise), we first train a general image ranking functions (the "Offline trained model" in the figure). This is used to retrieve images for a user based on his/her query. Looking at the initial retrieval results, the user may interactively provide feedback, in forms of newly labelled attribute existence labels and/or relative attribute strength labels. Such feedback is used as new training samples by an online-learning algorithm for updating the ranking function. As the feedback is specific

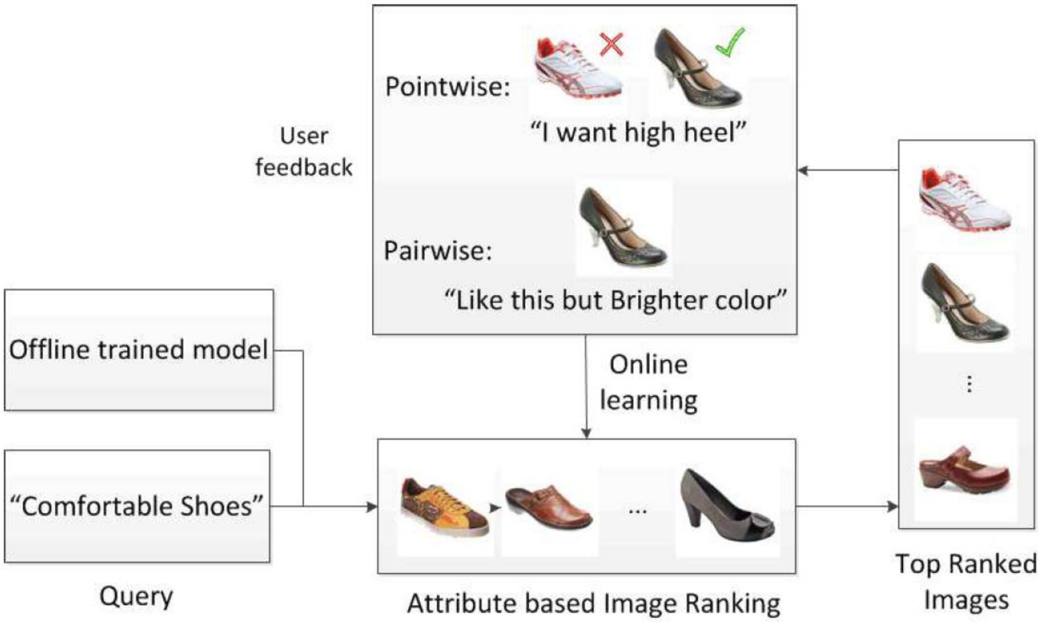


Figure 5.2: Adaptive image retrieval via on-line feedback.

to a user, the updated ranking function (and thus the retrieval engine) is presumably adapted to a user's preferences and hence achieving some level of personalization.

Hybrid Ranking To solve the above adaptive image retrieval task, we propose a hybrid ranking SVM framework. Given (1) the pointwise dataset $\mathcal{P}$ where each data sample $\boldsymbol{x}_i \in \mathcal{P}$ is assigned a category label and (2) the pairwise datasets including both the ordered pair set $\mathcal{O}$ and the un-ordered pair set $\mathcal{S}$ where for any pair $(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathcal{O}$, $\boldsymbol{x}_i \succ \boldsymbol{x}_j$ (e.g., $\boldsymbol{x}_i$ has a stronger attribute than $\boldsymbol{x}_j$), and for any pair $(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathcal{S}$, $\boldsymbol{x}_i \sim \boldsymbol{x}_j$ (e.g., $\boldsymbol{x}_i$ has a similar attribute value to $\boldsymbol{x}_j$), we attempt to learn a ranking model taking both the pointwise and pairwise data into consideration. This hybrid ranking approach aims to capture as much information as possible from all available data so as to achieve better ranking performance especially when labelled data are scarce.

5.2 Proposed Approach

In this section we propose a general hybrid ranking SVM framework and an online algorithm to solve this problem.

5.2.1 Hybrid Ranking SVM

To make the best use of the available knowledge, we propose a hybrid ranking SVM which takes both pointwise and pairwise labelled data samples for learning.

The approach presented in [82] for ordinal regression learns a number of parallel hyperplanes by the large margin principle as a ranking model. One implementation of the approach tries to maximize a fixed margin for all the adjacent classes. Relative attributes [67] applies pairwise learning-to-rank approach on image attributes for image ranking. This approach learns a ranking function for each human-nameable attribute of an image. The relative "strength" of an attribute is measured by some distance metrics learned through SVM-like optimization using (relatively) labeled pairs. Both of these two SVM models aim to optimize a project direction $\boldsymbol{w}$, such that $\langle \boldsymbol{w}, \boldsymbol{w} \rangle$ (i.e., the inverse of the margin) is minimized subject to the separability constraints (modulo margin errors in the non-separable case).

In the situation that the training data are very limited, learning $\boldsymbol{w}$ based on both pointwise and pairwise datasets jointly would become a necessity in order to achieve reasonable performance. To fuse information from both types of data, the margins assigned to them should be different. To this end, we introduce a new superparameter ρ representing the margin corresponding to the pairwise data. We propose the hybrid ranking approach as follows:

$$\begin{aligned}
\min_{\boldsymbol{w},\boldsymbol{b},\boldsymbol{\xi},\boldsymbol{\zeta},\boldsymbol{\eta}} & \frac{1}{2}\|\boldsymbol{w}\|^2 + c_1\tau_1\sum_i\sum_j(\xi_i^j + \xi_i^{*j+1}) + c_2\tau_2\sum\zeta_{ij} + c_2\tau_3\sum\eta_{ij} \\
s.t. \;\; & \boldsymbol{w}\cdot\boldsymbol{x}_i^j - b_j \leq -1 + \xi_i^j, \\
& \boldsymbol{w}\cdot\boldsymbol{x}_i^{j+1} - b_j \geq 1 - \xi_i^{*j+1}, \\
& \boldsymbol{w}\cdot(\boldsymbol{x}_i - \boldsymbol{x}_j) \geq \rho - \zeta_{ij}, \forall(i,j)\in\mathcal{O}, \\
& |\boldsymbol{w}\cdot(\boldsymbol{x}_i - \boldsymbol{x}_j)| \leq \eta_{ij}, \forall(i,j)\in\mathcal{S}, \\
& \xi_i^j \geq 0, \xi_i^{*j} \geq 0, \zeta_{ij} \geq 0; \eta_{ij} \geq 0
\end{aligned}$$

where $\boldsymbol{x}_i^j \in \mathcal{R}^n$ is an object (feature vector) with $j = 1, ..., k-1$ denoting the class number, $i = 1, ..., i_j$ is the index within class j, and k is the total number of classes. $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is sample pairs, ξ_i^j and ξ_i^{*j} are non-negative slack variables measuring the degree of misclassified data, ζ_{ij} and η_{ij} are soft margin slack variables for pairwise ranking, c_1 and c_2 are super parameters controlling the weight for the pointwise and pairwise data, τ_i is the weight function penalizing different training datasets according to the data size. Specifically, let n_1, n_2 and n_3 denote the data sizes of the pointwise, ordered and unordered pairwise datasets respectively, then $\tau_i = \frac{n_i}{\sum_{j=1}^3 n_j}, i = 1, 2, 3$. Note that if only pointwise data are provided then the framework is equivalent to regression, and if only pairwise data are provided the framework becomes pairwise ranking.

In the following discussion, we focus on the image retrieval task which can be simplified as a hybrid ranking model with "binary type " of pointwise label (i.e., existence/non-existence of certain attribute). For clarity, in the following, we use $\boldsymbol{x}_i^{\mathcal{P}}$ to denote the i-th pointwise training sample $x_i \in \mathcal{P}$, and $\boldsymbol{x}_i^{\mathcal{O}}(\boldsymbol{x}_i^{\mathcal{S}})$ denotes the difference of the i-th ordered(unordered) pairwise training sample as $x_p - x_q$ for any $(x_p, x_q) \in \mathcal{O}(\mathcal{S})$. Then the ranking model can be formulated as the following primal form of the hybrid learning problem:

$$\begin{aligned}
&\min_{\boldsymbol{w},\boldsymbol{b},\boldsymbol{\xi},\boldsymbol{\zeta},\boldsymbol{\eta}} \frac{1}{2}\|\boldsymbol{w}\|^2 + c_1\tau_1 \sum_i \xi_i + c_2\tau_2 \sum \zeta_i + c_2\tau_3 \sum \eta_i \\
&s.t.\quad y_i \cdot (\boldsymbol{w} \cdot \boldsymbol{x}_i^{\mathcal{P}} + b) \geq 1 - \xi_i, \\
&\qquad \boldsymbol{w} \cdot \boldsymbol{x}_i^{\mathcal{O}} \geq \rho - \zeta_i, \\
&\qquad \left|\boldsymbol{w} \cdot \boldsymbol{x}_i^{\mathcal{S}}\right| \leq \eta_i, \\
&\qquad \xi_i \geq 0, \zeta_i \geq 0; \eta_i \geq 0.
\end{aligned} \tag{5.1}$$

This formulation can be solved by quadratic programming.

5.2.2 Mini-Batch Online Learning Algorithm

I now propose an online learning algorithm for the hybrid ranking SVM for adaptive image retrieval. In the retrieval application, we first train a general ranking function for the user. Based on the retrieval results, the user may provide feedback (new category and relative labels) according to their preferences. Then our online learning approach will update the ranking function based on the newly labelled data to make the ranking results better fit to the user's personal needs.

The constrained quadratic programming problem of Eqn. (5.1) can be cast as an unconstraint empirical loss minimization with a penalty term for the norm of the classifier that can be learned in the following form:

$$\begin{aligned}
\min_{\boldsymbol{w}} \frac{1}{2}\|\boldsymbol{w}\|^2 &+ c_1\tau_1 \sum_{i\in\mathcal{P}} \ell_1(\boldsymbol{w}; (\boldsymbol{x}_i^{\mathcal{P}}, y_i^{\mathcal{P}})) \\
&+ c_2\tau_2 \sum_{i\in\mathcal{O}} \ell_2(\boldsymbol{w}; \boldsymbol{x}_i^{\mathcal{O}}) + c_2\tau_3 \sum_{i\in\mathcal{S}} \ell_3(\boldsymbol{w}; \boldsymbol{x}_i^{\mathcal{S}})
\end{aligned} \tag{5.2}$$

where

$$\begin{aligned}
&\ell_1(\boldsymbol{w}; (\boldsymbol{x}_i^{\mathcal{P}}, y_i^{\mathcal{P}})) = \max\{0, 1 - y_i^{\mathcal{P}} \langle \boldsymbol{x}_i^{\mathcal{P}}, \boldsymbol{w} \rangle\}, \\
&\ell_2(\boldsymbol{w}; \boldsymbol{x}_i^{\mathcal{O}}) = \max\{0, \rho - \langle \boldsymbol{x}_i^{\mathcal{O}}, \boldsymbol{w} \rangle\}, \\
&\ell_3(\boldsymbol{w}; \boldsymbol{x}_i^{\mathcal{S}}) = \left|\langle \boldsymbol{x}_i^{\mathcal{S}}, \boldsymbol{w} \rangle\right|.
\end{aligned}$$

```
Algorithm 6 The Mini-Batch Online Learning Algorithm
Input:
    1. Training set $\mathcal{A}$ with data type flags;
    2. Parameters $\rho$, $c_1$, $c_2$, $\tilde{t}$, $k$;
Output: $\boldsymbol{w}_{\tilde{t}}$;
Set $\boldsymbol{w}_0 = 0$;
for $t = 1, 2, ..., \tilde{t}$ do
    Choose $\mathcal{A}_t \subseteq \mathcal{A}$ where $|\mathcal{A}_t| = k$ uniformly at random;
    Set $\mathcal{A}_t^{\mathcal{P}} = \{i \in \mathcal{A}_t : (x_i, y_i) \in \mathcal{P} \wedge y_i \langle \boldsymbol{x}_i, \boldsymbol{w}_t \rangle < 1\}$, $n_1 = |\mathcal{A}_t^{\mathcal{P}}|$;
    Set $\mathcal{A}_t^{\mathcal{O}} = \{i \in \mathcal{A}_t : (x_i, y_i) \in \mathcal{O} \wedge \langle \boldsymbol{x}_i, \boldsymbol{w}_t \rangle < \rho\}$, $n_2 = |\mathcal{A}_t^{\mathcal{O}}|$;
    Set $\mathcal{A}_t^{\mathcal{S}} = \{i \in \mathcal{A}_t : (x_i, y_i) \in \mathcal{S}$, $n_3 = |\mathcal{A}_t^{\mathcal{S}}|$;
    Set $\eta_t = \frac{1}{(n_1+n_2+n_3)t}$;
    Set $\tau_1 = \frac{n_1}{\sum_{j=1}^{3} n_j}$, $\tau_2 = \frac{n_2}{\sum_{j=1}^{3} n_j}$, $\tau_3 = \frac{n_3}{\sum_{j=1}^{3} n_j}$;
    Set $\boldsymbol{w}_t \leftarrow (1 - \eta_t)\boldsymbol{w}_{t-1} + \eta_t(c_1\tau_1 \sum_{i \in \mathcal{A}_t^{\mathcal{P}}} y_i \boldsymbol{x}_i + c_2\tau_2 \sum_{i \in \mathcal{A}_t^{\mathcal{O}}} \boldsymbol{x}_i) + c_2\tau_3 \sum_{i \in \mathcal{A}_t^{\mathcal{S}}} \operatorname{sgn} \langle \boldsymbol{x}_i, \boldsymbol{w}_t \rangle \boldsymbol{x}_i)$;
end for
return $\boldsymbol{w}_t$;
```

Inspired by the Pegasos algorithm [81], we also considered the mini-batch algorithm which utilize k ($1 \leq k \leq m$) examples at each iteration. I initiate the model by setting $\boldsymbol{w}_0$ to the zero vector. In iteration t of the algorithm, given a training set $\mathcal{A}$ with m samples of both pointwise and pairwise data (a flag bit is used to identify the type of the data), we choose a subset $\mathcal{A}_t \subseteq \mathcal{A}$ with k examples uniformly at random among the training subset. Thus we will optimize the following approximate objective function:

$$\begin{aligned} f(\boldsymbol{w}; A_t) = \frac{1}{2}\|\boldsymbol{w}\|^2 + c_1\tau_1 \sum_{i\in\mathcal{A}_t} \ell_1(\boldsymbol{w}; (\boldsymbol{x}_i^{\mathcal{P}}, y_i^{\mathcal{P}})) \\ + c_2\tau_2 \sum_{i\in\mathcal{A}_t} \ell_2(\boldsymbol{w}; \boldsymbol{x}_i^{\mathcal{O}}) + c_2\tau_3 \sum_{i\in\mathcal{A}_t} \ell_3(\boldsymbol{w}; \boldsymbol{x}_i^{\mathcal{S}}) \end{aligned} \tag{5.3}$$

I employ the stochastic gradient methods in our algorithm. The sub-gradient of Eqn. (5.3) at iteration t is given by

$$\begin{aligned} \nabla_t =& \boldsymbol{w}_t - c_1\tau_1 \sum_{i\in\mathcal{A}_t} \chi_{\Re^+}(1 - y_i^{\mathcal{P}}\langle \boldsymbol{x}_i^{\mathcal{P}}, \boldsymbol{w}_t\rangle) y_i^{\mathcal{P}} \boldsymbol{x}_i^{\mathcal{P}} \\ & - c_2\tau_2 \sum_{i\in\mathcal{A}_t} \chi_{\Re^+}(\rho - \langle \boldsymbol{x}_i^{\mathcal{O}}, \boldsymbol{w}_t\rangle) \boldsymbol{x}_i^{\mathcal{O}} \\ & - c_2\tau_3 \sum_{i\in\mathcal{A}_t} \operatorname{sgn} \langle \boldsymbol{x}_i^{\mathcal{S}}, \boldsymbol{w}_t\rangle \boldsymbol{x}_i^{\mathcal{S}} \end{aligned} \tag{5.4}$$

where $\chi_A(x)$ is the eigenfunction and $\operatorname{sgn}(x)$ the symbolic function. Then the weight vector can be updated by

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \eta_t \nabla_t$$

with the step size $\eta_t = \frac{1}{(n_1+n_2+n_3)t}$. After a predetermined number $\tilde{t}$ of iterations, we output the final $\boldsymbol{w}_{\tilde{t}}$ as the learned projection model. The pseudocode of our algorithm is given in Algorithm (6). It can be shown that our proposed framework have the same convergence property with [81] and thus we can terminate the procedure at a random stopping time and in at least half of the cases the last hypothesis is an accurate solution. A detailed analysis of the online learning algorithm is attached in Appendix C.

5.2.3 Kernelization

I further derive the framework into the kernel form which strengthens our approach to learn non-linear model. Note that although the derivation is based on the online learning form, it can be generalized to batch learning since we are considering mini-batch learning in this work.

Instead of considering predictors which are linear functions of the training instances $\boldsymbol{x}$ themselves, we consider predictors which are linear functions of some implicit mapping $\phi(\boldsymbol{x})$ of the instances. Then the original optimization problem can be redefined as:

$$\begin{aligned}\min_{\boldsymbol{w}} \frac{1}{2}\|\boldsymbol{w}\|^2 + c_1\tau_1 \sum_{i\in\mathcal{P}} \ell_1(\boldsymbol{w}; (\phi(\boldsymbol{x_i}^{\mathcal{P}}), y_i^{\mathcal{P}})) \\ + c_2\tau_2 \sum_{i\in\mathcal{O}} \ell_2(\boldsymbol{w}; \phi(\boldsymbol{x}_i^{\mathcal{O}})) + c_2\tau_3 \sum_{i\in\mathcal{S}} \ell_3(\boldsymbol{w}; \phi(\boldsymbol{x}_i^{\mathcal{S}}))\end{aligned} \tag{5.5}$$

where

$$\begin{aligned}&\ell_1(\boldsymbol{w}; (\phi(\boldsymbol{x}_i^{\mathcal{P}}), y_i^{\mathcal{P}})) = \max\{0, 1 - y_i^{\mathcal{P}}\langle\phi(\boldsymbol{x}_i^{\mathcal{P}}), \boldsymbol{w}\rangle\}, \\ &\ell_2(\boldsymbol{w}; \phi(\boldsymbol{x}_i^{\mathcal{O}})) = \max\{0, \rho - \langle\phi(\boldsymbol{x}_i^{\mathcal{O}}), \boldsymbol{w}\rangle\}, \\ &\ell_3(\boldsymbol{w}; \phi(\boldsymbol{x}_i^{\mathcal{S}})) = |\langle\phi(\boldsymbol{x}_i^{\mathcal{S}}), \boldsymbol{w}\rangle|.\end{aligned} \tag{5.6}$$

Next we will directly derive the primal problem into the kernel form. For each t, let x_j represents the data sample, and let

$$\begin{aligned}&\alpha_t[j] = |\{t' \leq t : i_{t'} = j \wedge y_j^{\mathcal{P}}\langle\boldsymbol{w}_{t'}, \phi(\boldsymbol{x}_j^{\mathcal{P}})\rangle < 1\}|, \\ &\beta_t[j] = |\{t' \leq t : i_{t'} = j \wedge \langle\boldsymbol{w}_{t'}, \phi(\boldsymbol{x}_j^{\mathcal{O}})\rangle < \rho\}|, \\ &\gamma_t[j] = \sum_j \operatorname{sgn}\langle\phi(\boldsymbol{x}_j^{\mathcal{S}}), \boldsymbol{w}_{t'}\rangle, \forall j \in \{t' \leq t : i_{t'} = j\},\end{aligned}$$

then Eqn. (5.5) and (5.6) can be rewritten as

$$\begin{aligned}\boldsymbol{w}_{t+1} = \frac{1}{\lambda t}(c_1\tau_1 \sum_{j=1}^{n_1} \alpha_{t+1}[j] y_j^{\mathcal{P}} \phi(\boldsymbol{x}_j^{\mathcal{P}}) \\ + c_2\tau_2 \sum_{j=1}^{n_2} \beta_{t+1}[j]\phi(\boldsymbol{x}_j^{\mathcal{O}}) + c_3\tau_3 \sum_{j=1}^{n_3} \gamma_{t+1}[j]\phi(\boldsymbol{x}_j^{\mathcal{S}})).\end{aligned}$$

According to the Representer Theorem [48], the optimal solution to Eqn. (5.2) can be expressed as a linear combination of the training instances, thus we can rewrite

$\boldsymbol{w}$ as:

$$\boldsymbol{w} = \sum_{j=1}^{n_1} \alpha[j]\phi(\boldsymbol{x}_j^{\mathcal{P}}) + \sum_{j=1}^{n_2} \beta[j]\phi(\boldsymbol{x}_j^{\mathcal{O}}) + \sum_{j=1}^{n_3} \gamma[j]\phi(\boldsymbol{x}_j^{\mathcal{S}}),$$

Let $\boldsymbol{\vartheta}$ be the whole parameter vector and $\mathcal{D}$ be the whole training dataset include all three types of labeled data

$$\boldsymbol{\vartheta} = [\alpha[1\cdots n_1], \beta[1\cdots n_2], \gamma[1\cdots n_3]],$$
$$\mathcal{D} = [\phi(\boldsymbol{x}_{[1\cdots n_1]}^{\mathcal{P}})^T, \phi(\boldsymbol{x}_{[1\cdots n_2]}^{\mathcal{O}})^T, \phi(\boldsymbol{x}_{[1\cdots n_3]}^{\mathcal{S}})^T]^T,$$

and $\boldsymbol{d}_i$ is the i-th in $\mathcal{D}$, $n = n_1 + n_2 + n_3$, then the objective function can be written in the following kernel form through a kernel operator $K(\boldsymbol{x}, \boldsymbol{x}') = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle$, yielding the inner products after the mapping $\phi(\cdot)$:

$$\begin{aligned}
\min_{\boldsymbol{\vartheta}} \frac{1}{2} \sum_{i,j=1}^{n} & \boldsymbol{\vartheta}[i]\boldsymbol{\vartheta}[j]K(\boldsymbol{d}_i, \boldsymbol{d}_j) \\
& + c_1\tau_1 \sum_{i=1}^{n_1} \max\{0, 1 - y_i^{\mathcal{P}} \sum_{j=1}^{n} \boldsymbol{\vartheta}[j]K(\boldsymbol{x}_i^{\mathcal{P}}, \boldsymbol{d}_j)\} \\
& + c_2\tau_2 \sum_{i=1}^{n_2} \max\{0, \rho - \sum_{j=1}^{n} \boldsymbol{\vartheta}[j]K(\boldsymbol{x}_i^{\mathcal{O}}, \boldsymbol{d}_j)\} \\
& + c_2\tau_3 \sum_{i=1}^{n_3} |\sum_{j=1}^{m} \boldsymbol{\vartheta}[j]K(\boldsymbol{x}_i^{\mathcal{S}}, \boldsymbol{d}_j)|
\end{aligned}$$

5.3 Experiments

I evaluate our approach on two datasets with augmented relative attribute labels: (1) the **Outdoor Scene Recognition(OSR)** dataset [66, 67] with 2688 images and 7 attributes, and the Shoes dataset [9, 51] with 14568 images and 10 attributes. We directly use the features provided with the dataset of 512-dimensional gist descriptor for the OSR and 960-dimensional gist descriptor plus 20-dimensional color histogram for the Shoes.

Table 5.1: Ranking accuracies and standard deviation of 6 attributes on the OSR dataset when the number of training samples and pairs are 100 for each attribute.

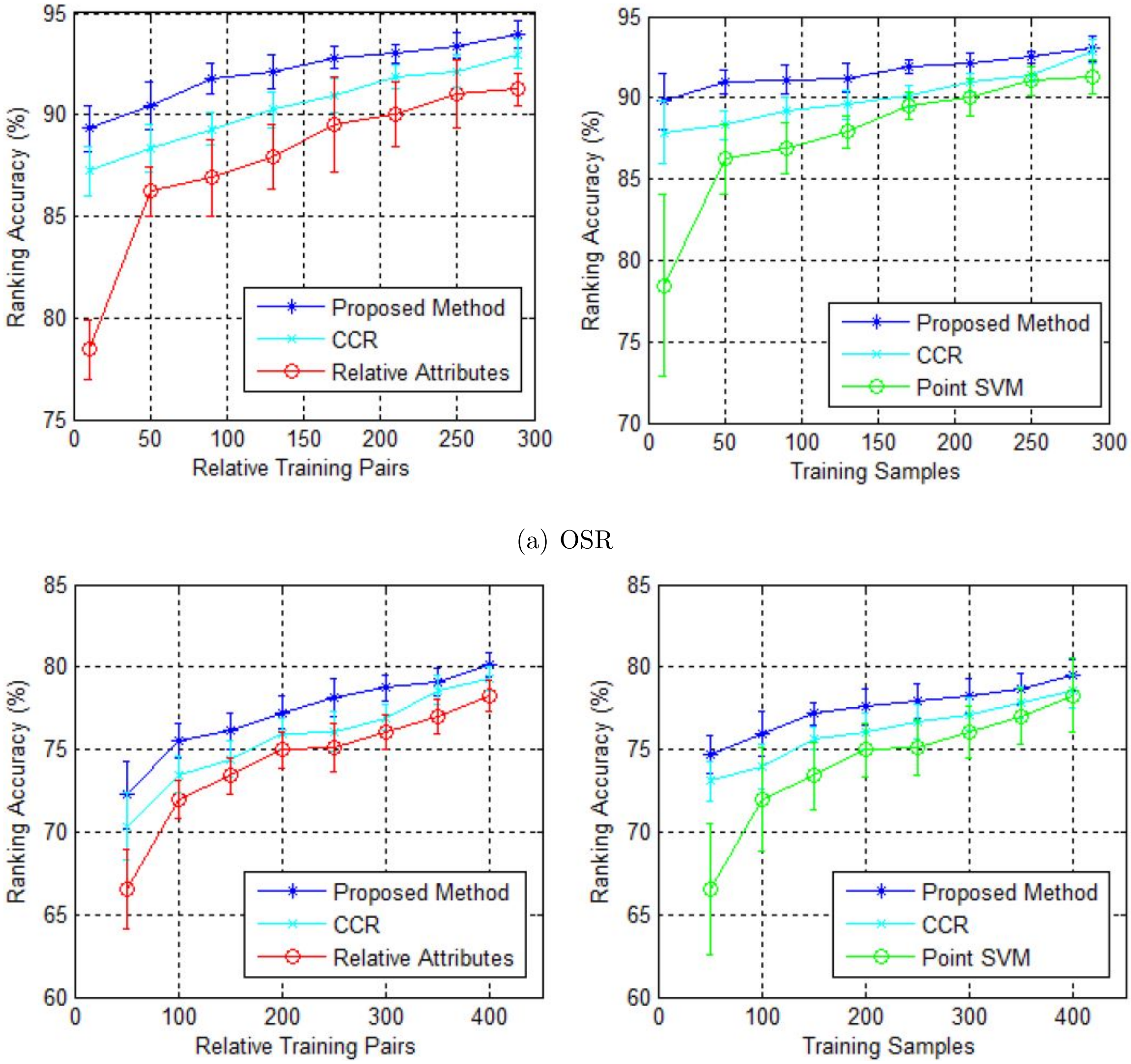
Attribute Name	Hybrid Ranking(%)	CCR(%)	Relative Attribute(%)	Pointwise SVM(%)
Natural	**91.56±0.89**	88.75±0.90	87.09±2.08	88.86±2.24
Open	**88.50±0.62**	87.25±0.82	86.83±1.76	85.72±1.26
Perspective	**83.40±0.78**	82.23±0.81	80.20±1.69	80.56±1.73
Size-large	**72.93±1.04**	70.62±1.21	67.89±1.55	65.17±3.53
Diagonal-plane	**80.35±1.15**	78.42±1.08	76.25±1.76	76.61±2.73
Depth-cloth	**87.06±0.87**	85.24±0.95	84.27±1.66	82.32±1.51
Average	**83.97±0.80**	82.08±0.96	80.42 ±1.75	79.87±1.78

Table 5.2: Ranking accuracies and standard deviation of 10 attributes on the Shoes dataset when the number of training samples and pairs are 200 for each attribute.

Attribute Name	Proposed Method(%)	CRR(%)	Relative Attribute(%)	Pointwise SVM(%)
Point at the front	**82.25±0.79**	81.42±0.82	80.61±1.08	79.13±1.53
Open	**76.02±0.83**	74.24±0.80	71.72±0.88	69.10±1.78
Bright in color	**64.40±0.76**	62.43±0.75	59.38±1.86	58.63±0.76
Covered with ornaments	**71.19±0.58**	70.02±0.61	68.88±0.72	59.10±3.87
Shiny	**79.60±0.52**	78.28±0.68	76.94±0.66	74.12±1.30
High at the heel	**80.71±0.75**	78.93±0.77	77.43±0.99	76.59±1.60
Long on the leg	**75.19±0.92**	73.32±0.82	72.44±1.07	69.08±2.19
Formal	**75.78±0.90**	74.45±0.91	72.37±1.10	70.76±1.57
Sporty	**80.24±0.90**	78.25±0.95	77.39±0.90	68.98±1.70
Feminine	**83.37±0.68**	82.42±0.70	81.38±0.71	81.86±1.50
Average	**76.88±0.76**	75.38±0.78	73.85±1.00	70.74±1.78

5.3.1 Accuracy of Hybrid Ranking

I first demonstrate that the proposed approach is capable of utilizing information from both type of labeled data with the comparison of three baseline approaches: Relative Attributes [67], pointwise SVM for ordinal regression [82] and CRR [80] which optimizes regression and ranking simultaneously. We compute the average ranking accuracy with standard deviation by running 10 rounds of each implemented approach. The average ranking accuracy is evaluated by the frequency of correctly



(a) OSR

(b) Shoes

Figure 5.3: Learning curve of average ranking accuracy and corresponding standard deviation with regard to different number of pairwise or pointwise training samples on both dataset.

ranked pairs. The parameters are selected by cross validation of 5 randomly selected small subsets per dataset.

Tables 5.1 and 5.2 demonstrate the average ranking accuracy with standard deviation of each attribute per dataset. Pointwise and pairwise training samples are randomly selected as 100 for OSR and 200 for Shoes with the rest of OSR and 6000 samples (due to memory limitation) from Shoes for test. The result shows that our approach apparently outperforms all baseline approaches in average ranking accuracies while generating lower standard deviations.

Fig. 5.3 illustrates how the ranking accuracy changes with different size of training samples on the attribute "Natural" of OSR and "Open" of Shoes. Specifically, the result in Fig. 5.3 Left is achieved by keeping the pointwise data size fixed at 100 and increasing the size of pairwise data. The blue curve shows the average ranking accuracy and standard deviation of our approach and the red and cyan curves shows the result of baseline approaches. In Fig. 5.3 Right, results of our approach (blue curve) compared with baseline approaches are shown where the size of training pairs is fixed at 100 and the size of training samples increases gradually. The results show that, in both configurations, our approach achieves obvious higher ranking accuracy and lower standard deviation than all baseline approaches. Besides, the result implies that, when fewer training samples were used, a higher performance gain was observed. For example, the best performance gain is 11% compared with Relative Attributes and 14% compared with pointwise SVM in "natural", when only 10 training samples or pairs are fed.

Fig. 5.4 shows some examples of ranked image pairs. In each column, the top image is more "natural" than the bottom image according to the ranking groundtruth. Fig. 5.4(a) is the comparison of our hybrid approach with Relative Attributes. The first three pairs (columns) are correctly ranked by our approach but incorrectly ranked by Relative Attributes, e.g., coast is more natural than highway, forest is more natural than inside city, mountain is more natural than buildings. The last three pairs are incorrectly ranked by our approach but correctly ranked by Relative Attributes. The reason for the incorrect classification may be that our approach assigned a wrong category label to the scene. For instance, our approach also classified the bottom

Table 5.3: Elapsed times in order to achieve the same ranking accuracy (the less the better). The first row shows given ranking accuracies, and the second/third row shows the times needed for online/batch learning respectively to achieve the corresponding accuracy.

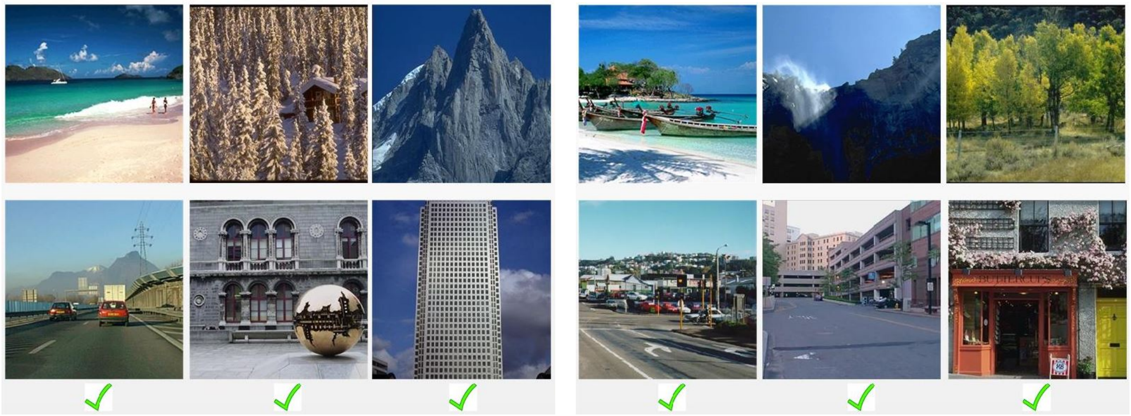
Accuracy	70%	72.5%	75%	77.5%	80%	82.5%	85%
Online Learning	0.003 s	0.004 s	0.006 s	0.009 s	0.042 s	0.105 s	0.156 s
Batch Learning	0.006 s	0.098 s	0.104 s	0.231 s	0.451 s	1.558 s	6.308 s

image of the fifth column as forest because of a tree appears in the scene. Fig. 5.4(b) illustrates the comparison of our approach with pointwise SVM. The first three pairs are correctly ranked by our approach but incorrectly ranked by pointwise SVM, e.g., coast is more natural than street, mountain is more natural than open county and forest is more natural than inside city. The last three pairs are incorrectly ranked by our approach but correctly ranked by pointwise SVM.

Based on these experiments, the potential performance gains of our approach appear to come from the extra information captured from different types of labels. In particular, the smaller standard deviation may result from the joint use of both information sources that help to denoise the training process.

5.3.2 *Zero-Shot Learning*

To further evaluate the proposed approach, we now consider the popular application of zero-shot learning. Given some training samples from some "seen" categories and some "unseen" categories without any training samples, zero-shot learning would predict the category labels of new samples. We compare our approach with the baseline approach Relative Attributes since [67] has shown that this approach outperforms most of the state of the art approaches on this regard. We followed the same parameter prediction rules of unseen categories as [19]. We adopted the same super parameters in Sect. 5.3.1 for model training. The average ranking accuracies with corresponding



(a) Hybrid v.s. Relative Attribute

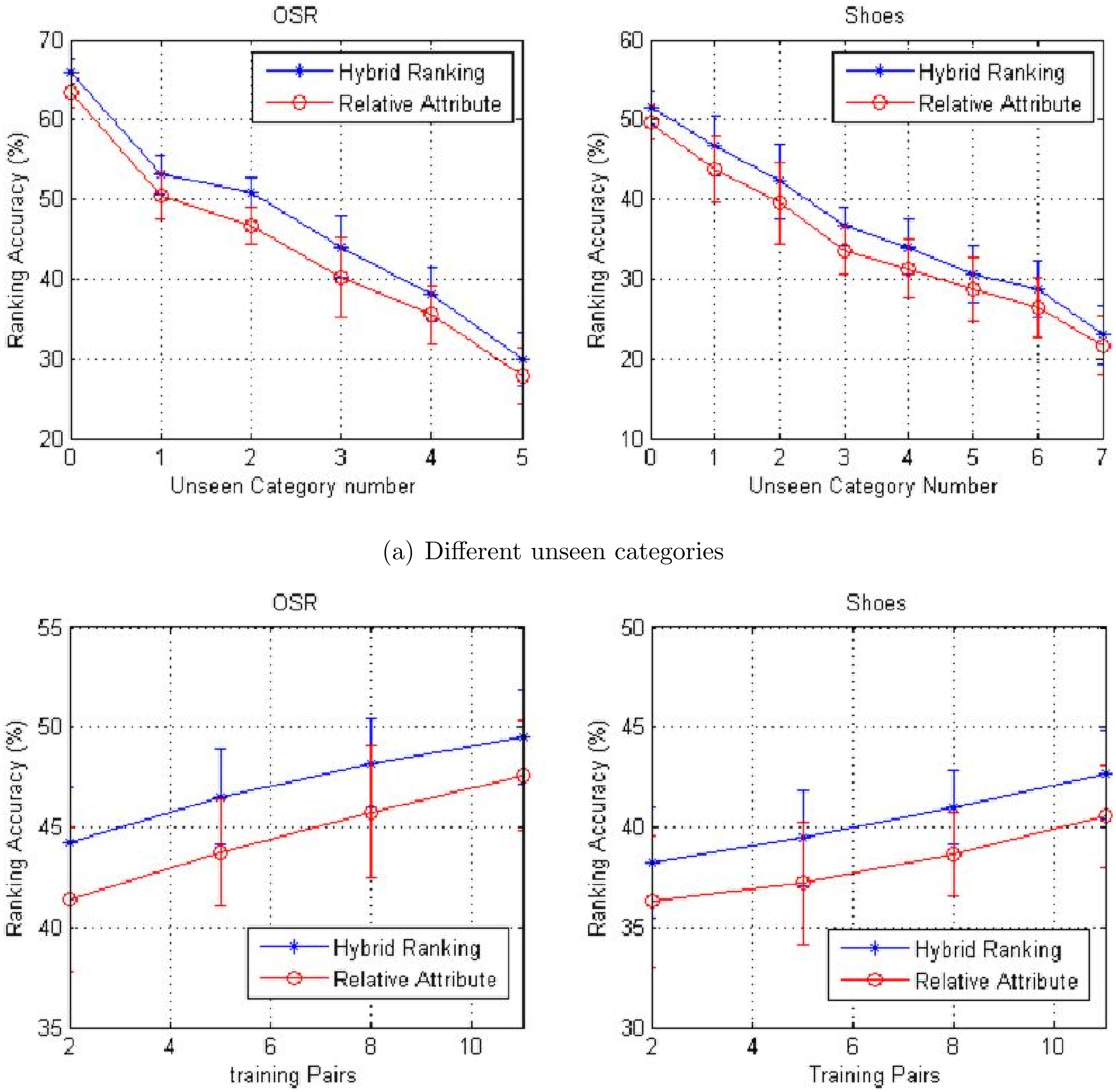
(b) Hybrid v.s. Pointwise SVM

Figure 5.4: Samples illustrating the ranking results. In groundtruth the top image is more "natrual" than the bottom image. The left three column is correctly ranked by the proposed approach while incorrectly ranked by the baseline. The right three is inverse.

standard deviations are reported by running each experiment 10 rounds. Assuming the data follows a Gaussian distribution, we estimated the mean and the covariance matrix of each (seen and unseen) category and assigned the category label of the new sample through maximum likelihood.

Fig. 5.5(a) shows the accuracy as a function of the number of unseen categories. For each seen category 30 images are left out for category parameter prediction, and 10 pointwise and 10 pairwise labelled samples are randomly picked for training. Results show that the ranking accuracy decreases as the number of unseen category increases. Our approach outperforms the baseline approach by around 3%.

Fig. 5.5(b) shows how the accuracy changes with the number of training pairs. In each run, 2 unseen categories and 30 images from the other seen categories are left out. 10 pointwise labelled samples are randomly picked for hybrid approach. Results show that ranking accuracies of both approaches increase with the increase of training pairs. Our approach yields performance gains by around 3% compared with the baseline approach.



(a) Different unseen categories

(b) Different training pairs

Figure 5.5: Learning curve of zero-shot learning accuracy with regard to different unseen category numbers and training sample size on both dataset.

5.3.3 Online Learning Evaluation

In this subsection, we compare the performance of the proposed online learning algorithm with the batch learning algorithm on the shoes dataset. For the online learning algorithm, the super parameters are set as $c_1 = 0.2$, $c_2 = 3$, $\rho = 0.1$. In training, we first construct a data pool mixed with both pointwise and pairwise data, and then randomly pick one data sample without considering the specific label type from the data pool for training. For batch learning, we construct the training dataset as half pointwise and half pairwise samples.

Fig. 5.6 illustrates the average ranking accuracy of 10 attributes by running both implemented approaches 10 rounds for a small time interval T (T=0.1 second in this experiment), simulating very limited training data availability. In each group, the first bar (blue) shows the result of batch learning and the second bar (red) shows the result of online learning. The results shows that in the same elapsed time of 0.1 second, the online learning algorithm clearly outperforms batch learning. The highest performance gain is obtained on the attribute "Sporty" by 9.69% and the lowest performance gain is for the attribute "Formal" by 4.63%.

Table 5.3 collects the elapsed time after both approaches achieved the same ranking performance from 70% to 85%. The results show that the batch learning approach takes longer time than online learning to achieve the same accuracy. With the ranking accuracy increased, the time difference become much more obvious. For example, batch learning takes double time (0.006s vs 0.003s) than online learning to achieve the accuracy of 70%, and takes 40 times (6.308s vs 0.156s) more time to achieve the accuracy of 85%.

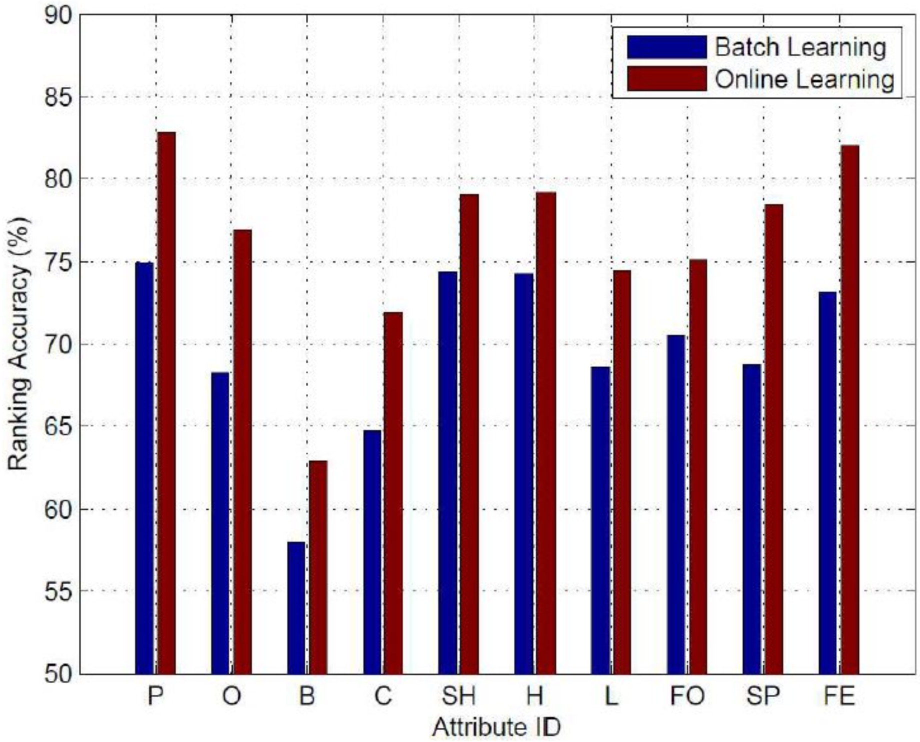


Figure 5.6: Average ranking accuracy of 10 attributes on the Shoes dataset by running the algorithms for 0.1 seconds. In each group the first bar is the result of batch learning and the second bar is the result of online learning.

5.4 Conclusions

I proposed a hybrid ranking framework for supporting adaptive, attribute-based image retrieval. We evaluated the proposed approach on two image datasets. The results show that through capturing the information from both relative attribute strength (pairwise) and absolute attribute scale (pointwise), our method is able to achieve better ranking performance than Relative Attribute and pointwise SVM, which are current leading approaches that learn the ranking function purely based on either pairwise or pointwise data. We also proposed an online learning algorithm for the proposed framework and derived the formulation into the kernel form. The experiments of online learning and batch learning show that our online learning algorithm can achieve much better ranking performance than batch learning given the same running time or can achieve better performance in much less time. The results also suggest that the less training data are available, the more relative performance gains can be obtained by our approach than independent learning.

Chapter 6

ATTRIBUTE LEARNING ON VIDEOS

In this chapter I discuss the exploration of utilizing semantic attribute learning for video analysis. Specifically, my work is focus on skill evaluation.

6.1 Introduction

Video-based coaching systems aim at helping people to improve their skills through capturing their performance via video recordings that allow either on-line or off-line analysis. Applications of such systems include dance [2, 27], sports [45], and machine-operation training [88], etc. Traditionally, the analysis is performed only by humans (e.g., coaches and trainers). Recent years have witnessed increasing interests in developing automated system for doing such analysis for improved training, where the key task is vision-based motion skill understanding since the coaching task often boils down to providing corrective feedback to a trainee regarding his/her movements.

Most existing methods may provide one of the following two types of feedback. The first type is an overall assessment with either a numeric rating [27] or a skill level [76]. While being useful for skill examinations, such type of feedback provides little suggestion to a trainee as to how to improve. Further, many methods [2, 27, 76] are based on comparison using some sort of standard action series or models. This limits the applicability of such methods to complex tasks where defining a standard action or model is impractical due to the existence of a wide range of valid/perfect solutions. The second type of feedback is some statistics computed from a user's training sessions, such as total execution time, movement counts, motion smoothness, etc. Although such statistics are more informative, they do not readily lead to corrective

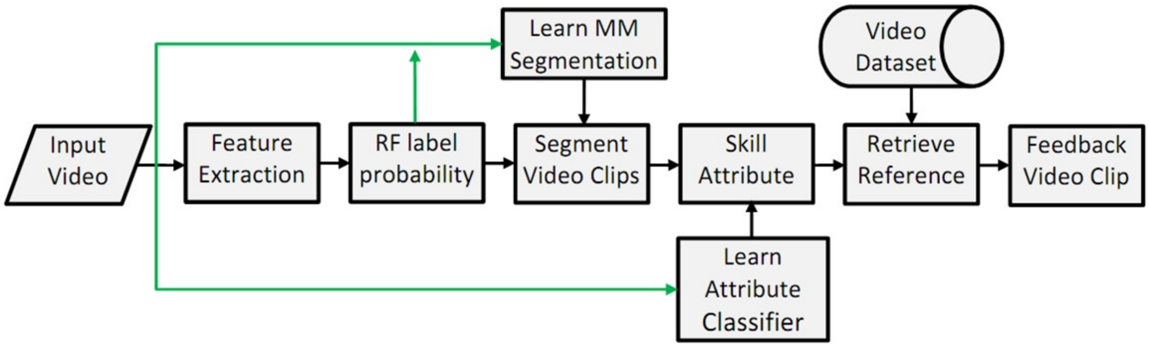


Figure 6.1: Illustration of the proposed skill coaching system that retrieves an illustrative video as feedback while providing specific and expressive verbal suggestions.

actions that the trainees may take to improve their performance.

In this chapter, I present a video retrieval system (illustrated in Figure 6.1) for skill coaching in simulation-based surgical training, which we defined as **instructive video retrieval**. We aim at providing automated video and verbal feedback that has the following three features: (1) **specificity**: the feedback should focus on a trainee's skill weakness; (2) **superiority**: the retrieved illustrative video should represent a better skill than the trainee; (3) **similarity**: the retrieved illustrative video should have a similar operation context to the trainee's video. Note that although the focus of the work is on the specific application of simulation-based surgical training, the above features are deemed as critical to effective skill coaching in general [22], and thus the proposed method can be extended to other video-based skill coaching applications by integrating corresponding domain knowledge.

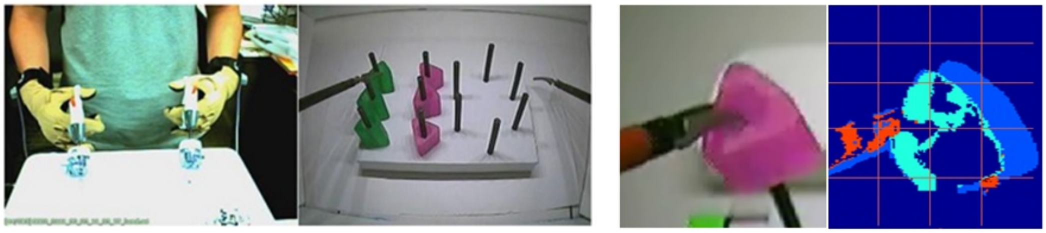
Different from traditional video retrieval such as nearly-duplicated video retrieval [59] or concept retrieval [62], which are purely based on video content, the instructive video retrieval requires both low-level content analysis and high-level semantic skill understanding. To our knowledge, this is still a new research effort with little prior art. In this work, I introduce semantic attributes in video to bridge the gap between inherent vagueness and the subjectivity of "instructive".

The technical contribution of this Chapter is threefold. First, I propose a new video retrieval problem defined as “instructive video retrieval” and a corresponding effective algorithm to solve this problem. This new problem has a wide variety of applications and more efforts are worth investing. Second, I extend image attribute learning into the video domain for skill evaluation, which is useful to bridge the gap between the low-level motion measurements and high-level skill understanding. Third, I develop a vision-based skill coaching system for simulation-based skill training, which provides an automatic and efficient way for self skill improvement without costly human supervision.

This study is primarily based surgical training on the Fundamentals of Laparoscopic Surgery (FLS) trainer box (www.flsprogram.org), a simulation-based platform that has been widely used in many hospitals for minimally-invasive surgery training. The system is essentially a box simulating the human body and a trainee is required to use tools going into the box through small holes to perform actions like cutting synthetic tissues inside the box (Fig. 6.2(a) left). The trainee can see what is going on inside the box only through a monitor that displays a live video (Fig. 6.2(a) right) captured by an on-board camera. In the operation, a trainee is required to lift one of the six objects with a grasper in his non-dominant hand, transfer the object midair to his dominant hand, and then place the object on a peg on the other side of the board. Once all six objects have been transferred, the process is reversed from one side to the other.

To attain the above objectives, we define the following technical tasks and develop a suite of algorithms for addressing these tasks:

- Decomposing a video sequence of a training procedure into primitive action units.



(a) (b)

Figure 6.2: The illustration of (a) the FLS system; (b) object-motion distribution for action recognition.

- Rating each action using expressive attributes derived from established guidelines used by domain experts.

- Recommending an illustrative video as a reference from a pre-stored database.

6.2 Proposed Method

In this section, I present our proposed method for video-based skill coaching which performs three key tasks: (1) Decomposing a video clip into primitive action units; (2) Rating each action unit using semantic attributes; (3) Retrieving an illustrative video for instruction. Fig. 6.3 presents a flow chart of our system, outlining its major algorithmic components and their interactions.

6.2.1 Primitive Action Segmentation

I first segment the given video into clips where each clip only includes one primitive action. The FLS operation consists of 5 primitive actions [49] as building blocks of manipulative surgical activities: (**L**): lift an object from the peg, (**T**): transfer an object, (**P**): place an object on the peg, (**W**): move the grasper with an object, (**U**) move the object without an object. Since the videos we consider exhibit predictable motion patterns arising from the underlying actions of the human subject, we adopt the Hidden Markov model (MM) in the segmentation task. This allows us to in-

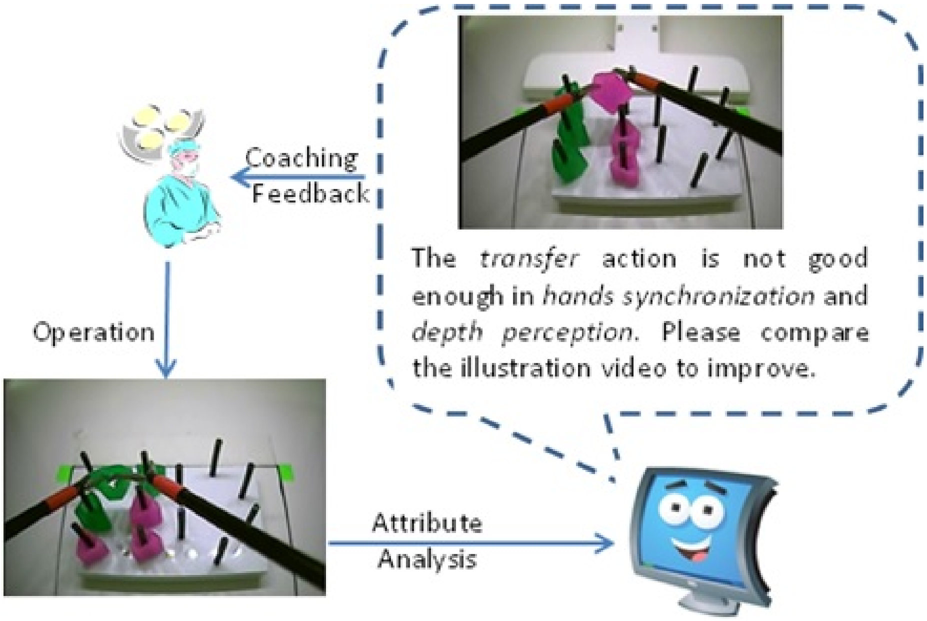

Coaching Feedback
Operation
Attribute Analysis
The *transfer* action is not good enough in *hands synchronization* and *depth perception*. Please compare the illustration video to improve.

Figure 6.3: An overview of the proposed approach. The green components are only used in the training stage.

corporate domain knowledge into the transition probabilities, e.g. the Lift action is followed by itself or by Loaded Move with high probability. Following [75, 88], we define each state as a primitive action. The task of segmentation is then to find the optimal state path for the given video.

Frame-level Feature Extraction The FLS box is a controlled environment includes 4 types of objects: background, rubber cubes, pegs, and tools/graspers. Due to the noisy video clips, we designed a probability representation of the motion information which served as features for action segmentation. Specifically, we first use random forest (RF) [23] to obtain the label probability $p_l(x)$ that a pixel x belongs to the object label l. Then the tool orientation and tips region and can be further detected by the spacial information based on the obtained probabilities. Since all surgical actions occur in the region around the grasper tip, the region is defined as the ROI region (Fig. 6.2(b) left) to filter out other irrelevant background. With the comparison with the distribution of the background region, we estimate the probability of each pixel x is “moving” by frame differencing, which is denoted by $p_m(x)$. Then the joint distribute $p_l(x)\cdot p_m(x)$ represents the probability that the pixel x is the moving object l. This joint object-motion distribution suppresses the static clutter background in the ROI so that only interested motion information will be reserved. To further capture the spacial information, we further split the ROI into blocks, as shown in Fig. 6.2(b) right, and describe the object-motion distribution in each block by the Hu-invariant moment [37]. Finally the moment vectors in each block are cascaded into a frame-level descriptor for action recognition.

Random Forest as Observation Model

After obtain the frame-level descriptor, we further utilize the RF for frame-level action recognition. Since RF is an ensemble classifier with a set of decision trees and the output is based on majority voting of the trees in the forest, the frame-level action distribution provides a good estimation of the observation model for the HMM states. Assuming that there are N trees in the forest and n_i decision trees assign primitive action label i to the input frame, we could view the random forest choose the label i with probability n_i/N which can be taken as the observation probability for State (primitive action) i.

Bayesian Estimation of Transition Probability

The transition probability from State i to State j can be estimated based on small set of data as the ratio the number of (expected) transitions from i to j over the total number of transitions. However, one potential issue of this method is that, in video segmentation we have limited training data in video segmentation. Furthermore, the number of transitions among different states (primitive action), is typically much less than the total number of frames of the video. This will result in a transition prob-

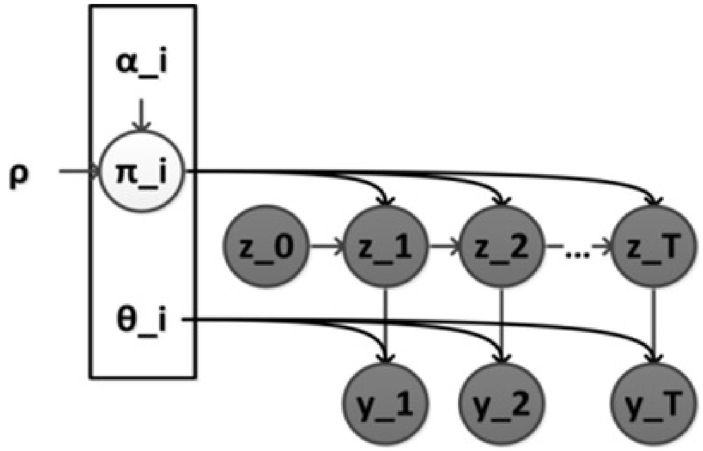


Figure 6.4: The graphical model for Bayesian estimation of transition probability, where the symbols with circles are hidden variable to be estimated, the symbols within gray circle are observations and the symbols without circle are priors.

ability matrix which is dominated by diagonal elements. The resulting transition probability will degrade the benefit of using HMM for video segmentation, i.e., forcing desired transition pattern in the state path. Thus, we propose to use a Bayesian approach for estimating the transition probability, employing the Dirichlet distributionwhich enables us to combine the domain knowledge with the limited training data for transition probability estimation. The model is shown in Fig. 6.4.

Assuming $\alpha_i(\sum_j \alpha_i(j) = 1)$ is our domain knowledge for the transition probabilities from State i to all states, then we can draw the transition probability vector π_i as $\pi_i \sim dir(\rho\alpha_i)$ where dir is the Dirichlet distribution as a distribution over distribution, and represents our confidence of the domain knowledge. Given the transition probability π_i, the count of transition from State i to all states follows a multinomial distribution:

$$n_i \sim multi(n_i|\pi_i) = \frac{(\sum_j x_i(j))!}{\prod_j n_i(j)!} \prod_j \pi_i(j)^{n_i(j)}. \tag{6.1}$$

Because the Dirichlet distribution and multinomial distribution is a conjugate pair, the posterior probability of transition probability is just combining the count of transition among state and domain knowledge (prior) as $\pi \sim dir(n_i + \rho\alpha_i)$. When there

Table 6.1: Action attributes for surgical skill assessment.

Attributes	Description
Time and motion (**T**)	How efficiently a trainee can operate without unnecessary moves.
Flow of operation (**F**)	How smoothly a trainee can operate without frequently stops.
Bimanual dexterity (**B**)	How well two hands can cooperate and work together.
Respect for issue (**R**)	How force is controlled in operation of objects as subjective evaluation of organ damage.
Instrument handling (**I**)	How well a trainee operates instruments without bad attempts and movements.
Depth perception (**D**)	How good a trainee's sense of depth to avoid failed operation on a wrong depth level.

are not enough training data, i.e., $\sum_i n_i(j) \ll \rho$, π_i would be dominated by α_i, i.e., our domain knowledge; as more training data become available, π_i would approximate to the counting of transitions in the data.

6.2.2 Attribute Learning for Action Rating

A fundamental challenge in instructive video retrieval is to map computable visual features to semantic concepts that are meaningful to a trainee. Recognizing the practical difficulty of lacking sufficient amount of exactly labeled data for learning an explicit mapping, we introduce the concept of image attribute into video domain to evaluate the underlying skill of an action clip based on semantic attributes designed using domain knowledge. Following [65, 74, 24], we define 6 attributes to measure the skill level of the trainee's operation, which are listed in Table 6.1. With these semantic attributes, the system will be able to expressively inform a trainee what is the weakness in the operation, since the defined attributes are all semantic concepts used in existing human-expert-based coaching (and thus they are well understood).

Feature Representation

To represent the defined attributes, new motion features are constructed for skill rating in 3 steps. First, a few types of motion measurements, which are summarized in Table 6.2, are calculated based on the previous object segmentation information

Table 6.2: Motion measurements.

Motion	Description	Definition
Instrument	The motion of grasper tip.	$v(t)$
Instrument target	Relative motion between grasper tip and its operation target.	$\hat{v}(t)$ $v_r(t)$
Object	The motion area of objects in ROI.	$A(t)$
ROI	The optical flow field in ROI.	$m(x,t)$

(Sect. 4.1). Second, we extract motion signatures from each of the motion measurement, which are summarized in Table 6.3. The motion signatures are 1-dimensional temporal signals that further compact the motion information. Last, final motion feature are constructed from each motion vector and its motion signatures as follows. In the temporal domain, we divide a signature into equal temporal bins; in the Fourier domain, we also divide the frequency into equal bins. In each temporal and frequency bin, the maximal, minimal, and average values are cascaded into the final feature set.

Table 6.3: Motion signatures. $y(t)/y(x)$ represents any motion vector in Table 6.2, e.g. $v(t)$, $v_r(t)$, $A(t)$, etc. $\bar{y}(t)$ is the smooth result of $y(t)$. m is the shorthand for field motion $m(x,t)$.

Name	Definition	Description
Velocity	$\lvert y(t)\rvert$	Instant velocity
Path	$\int_0^t \lvert y(x)\rvert dx$	Accumulated motion energy
Jitter	$\lvert y(t)-\bar{y}(t)\rvert$	Motion smoothness metric
CAV	$\int \frac{\lvert\langle \nabla \times m, m\rangle\rvert}{\|m\|_2} dx$	Curl angular velocity

Relative Attribute Skill Rating

To rate the skills of each segment primitive action video clip, we introduce the relative attribute learning [67], multi-task relative attribute ranking in Chapter 3 and hybrid relative ranking in Chapter 5 to calculate a relative ranking of the clips with respect to the defined semantic attributes. Formally, for the k-th attribute, we are given a set of ordered pairs of clips $O_k = \{(i,j)\}$ and a set of un-ordered pairs $S_k = \{(i,j)\}$, where $(i,j) \in O_k$ means the video clip v_i has a better skill performance than the video clip v_j (i.e. $v_i \succ v_j$) in terms of the specified attribute and $(i,j) \in S_k$ means v_i and v_j have similar skill performance (i.e. $v_i \sim v_j$).

Relative attribute learning The relative attribute framework to learn the model $\boldsymbol{w}_k$ for relative skill rating can be formulated as:

$$\begin{aligned} &\min_{\boldsymbol{w}_k,\epsilon,\gamma} \frac{1}{2}\|\boldsymbol{w}_k^T\|_2^2 + C(\sum \epsilon_{ij}^2 + \sum \gamma_{ij}^2) \\ &s.t. \;\; \boldsymbol{w}_k^T(\boldsymbol{x}_i - \boldsymbol{x}_j) \geq 1 - \epsilon_{ij}, \forall (i,j) \in O_k; \\ &\qquad |\boldsymbol{w}_k^T(\boldsymbol{x}_i - \boldsymbol{x}_j)| \leq \gamma_{ij}, \forall (i,j) \in S_k; \\ &\qquad \epsilon_i \geq 0, \gamma_{ij} \geq 0. \end{aligned} \tag{6.2}$$

where $\boldsymbol{x}_i$ is the feature vector extracted from the i-th video clip, C is the trade-off constant to balance maximal margin and pairwise attribute order constraints. The relative skill performance strength can be compared by the skill attribute value $\boldsymbol{w}_k^T \boldsymbol{x}_i$, which is used in the subsequent retrieval of illustrative video.

Multi-task Relative Attribute Learning The multi-task relative ranking framework can be formulated as:

$$\begin{aligned} \min\nolimits_{W,\epsilon,\gamma} \quad & \sum_{t}^{T} \frac{1}{2}|W_t|_2^2 + \lambda |W|_* + \rho_1 \sum_{(i,j)\in\mathcal{O}_t} \epsilon_{ij}^t + \rho_2 \sum_{(i,j)\in\mathcal{S}_t} \gamma_{ij}^t \\ \text{s.t.} \quad & W_t^\top (X_{it} - X_{jt}) + \epsilon_{ij}^t \geq 1 \\ & -\gamma_{ij}^t \leq W_t^\top (X_{it} - X_{jt}) \leq \gamma_{ij}^t \\ & \epsilon_{ij}^t \geq 0; \gamma_{ij}^t \geq 0 \end{aligned} \tag{6.3}$$

where $|\cdot|_*$ is the nuclear norm or the sum of singular values of the matrix for casting the low-rank constraint. We refer the proposed solution in Eqn. 6.3 as Max-Margin

Multi-Attribute Learning with Low-Rank Constraint.

Hybrid Relative Attribute Learning Our idea is to introduce an extra variable K as the margin thresh, and formulate the following hybrid relative attribute learning framework:

$$\begin{aligned} \min_{\boldsymbol{w},\epsilon,w_0,\gamma,K} & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C_1\sum\nolimits_{i=1}^{m}\epsilon_i + C_2\sum\nolimits_{i=1}^{m}\gamma_i \\ s.t. \quad & y_i \cdot \langle w, \boldsymbol{x}_i^{(1)} - \boldsymbol{x}_i^{(2)} \rangle \geq 1 - \epsilon_i \\ & z_i \cdot (\langle \boldsymbol{w}, x_i \rangle + w_0) \geq K - \gamma_i \\ & \gamma_i \geq 0, \epsilon_i \geq 0. \end{aligned} \tag{6.4}$$

where $\boldsymbol{w}$ is the ranking model; $y_i = 1(-1)$ means $\boldsymbol{x}_i^{(1)}$ is superior (inferior) to $\boldsymbol{x}_i^{(2)}$; $z_i = 1(-1)$ means x_i is instructive/non-instructive; ϵ and γ are slack variables. This problem can be solved by quadratic programming.

6.2.3 Illustrative Action Clip Retrieval

With the previous processing, the system will retrieve an illustrative video clip from a constructed video repository and present it to a trainee as a an instructive video.

Operation Weakness Detection

I first need to figure out what is the operation weakness. However, a lowest attribute value does not always mean the most urgent attribute in need of improvement, especially when this attribute is “difficult” to most of the people. The weakest attribute should be the one that the trainee did poorly while most other people are significantly better. Thus, we use the average cumulative distribution of one user’s training session to assess the performance strength of each attribute. Specifically, with K attributes, each clip v_i can be characterized by a K-dimensinoal vector $[a_{i,1}, \cdots, a_{i,K}]$, where $a_{i,k} = \boldsymbol{w}_k^T \boldsymbol{x}_i$ is the k-th attribute value of v_i based on its feature vector x_i. The attribute values of all clips (of the same action) in the data repository forms a $N \times K$ matrix A whose column vector a_k is the k-th attribute value of each clip. Similarly, from a user's training session, for the same action under consideration, we have another set of clips with attribute matrix $\hat{A}$ whose column vector $\hat{a}_k$ is the user's k-th attribute values in the training session. The performance strength of the k-th attribute s_k can be calculated by

$$s_k = \frac{1}{n} \sum_{i=1}^{n} P(\boldsymbol{a}_k \geq \hat{a}_{i,k})) \tag{6.5}$$

where n is the total number of video clips in one training session of one primitive action. Note that higher s_k means more users are doing better than the current operation, which means high importance the attribute need to improve.

Video Utility Detection Evaluation

I then need to figure out how much a video clip v_i is helpful with regard to a given training video, which we defined as the utility of v_i on the k-th attribute, denoted as $u_{i,k}$. We measure the utility by normalized distance between the performance strength of these two videos. Specifically, let s_k and $\hat{s}_k$ denotes the performance strength of the input training video and the potential instructive video clip v_i on the k-th attribute, the utility is calculated as

$$u_{i,k} = \frac{s_k - \hat{s}_k}{s_k} \tag{6.6}$$

After the calculation of these measurement, we select the best illustration video clip v_i^* from the data repository using the following criterion:

$$v_i^* = \arg\max_i \sum_{k=1}^{K} s_k \cdot u_{i,k} \tag{6.7}$$

The underlying idea of Function (6.7) is that a good feedback video should have high utility on important attributes.

With the above attribute analysis, we also provide a verbal description with regard to the 3 worst action attributes with an absolute importance above a threshold 0.4, which means that more than 60 percent of the pre-stored action clips are better in this attribute than the trainee. If all attribute importance values are lower than the threshold, we simply select the worst one. With the selected attributes, we retrieve the illustration video clips, inform the trainee about on which attributes he performed poor, and direct him to the illustration video. It is worth noting that in recommending an illustration video, we defined concepts that are context dependent. That is, the importance and utility values of an attribute depends on the given data set. In practice, the data set could be a local database captured and updated frequently in a training center, or a fixed standard dataset, and thus the system allows the setting of some parameters (e.g., the threshold 0.4) based on the nature of the database.

6.3 Experiments

I tested our framework on a 64-bit computer with Intel Core i5-2500 CPU @ 3.30GHz and 8.00G RAM. Experiments have been performed using realistic training videos capturing the performance of resident surgeons (includes experts and novices) in a local hospital during their routine training. A typical testing video contains about 4500 frames with the frame rate of 30 FPS and the resolution of 480 × 720. For acceleration, we down-sampled the resolution and frame rate by 2. Experiments showed that our system takes on average around 242 seconds to process such a video.

For evaluating the proposed methods, we selected 10 representative videos/subjects from trainees of different skill levels. Each video is a full training session consisting of 12 Peg Transfer cycles which leads to 12 video clips for each primitive operation. We have a database of 240 clips for each of the 3 primitive operations, i.e. lift, transfer, and place, with the total clips being 720. We emphasize that, even the same subject does not perform the same action identically (and in fact the variability was observed to be very high), and thus the clip dataset is very diverse, our experiment dataset provides a reasonable basis for evaluating our method. The exact frame-level labeling (which action each frame belongs to) were manually obtained as the ground truth. For each primitive action, we randomly select 150 pairs of video clips and then manually label them by examining all the attributes previously defined. This process manually determines which video in a given pair should have a better skill according to a given attribute.

6.3.1 Action Segmentation

As discussed in the previous section, we first calculated the frame level action classification accuracies, then the classification scores (probabilities) are used as the observation into an HMM to get a final action recognition result, which is solved by the Viterbi algorithm. The confusion matrix of the action recognition accuracies before and after employing HMM is shown in Table 6.4, which is calculated by leave-one-video-out cross validation. It can be seen that the frame level recognition result is already high for some actions, which verifies the effectiveness of our proposed motion descriptors. The recognition accuracies after using the HMM are significantly improved, especially for actions L and P. The overall low accuracy for actions L and P is mainly due to the trainee's unsmooth operation that caused many unnecessary stops and moves, which are hard to distinguish from UM and LM. The overall segmentation accuracy of expert videos is 93.5% while the accuracy of novice videos is 80.3%. The results show that the proposed action segmentation method is able to deliver reasonable accuracy in face of some practically challenges.

Table 6.4: Confusion matrix of the action recognition accuracies. Each cell is the accuracy (%) with/without HMM.

	UM	L	LM	T	P
UM	87.6/88.0	0.2/0.2	0.6/0.8	11.5/10.3	0.8/0.8
L	21.9/36.1	43.4/28.5	21.8/15.8	13.0/13.3	0.0/6.3
LM	3.8/18.0	0.2/1.1	77.3/61.1	12.8/12.3	6.0/7.5
T	5.6/11.3	0.0/0.1	1.0/0.9	93.4/87.7	0.0/0.0
P	28.7/55.1	0.6/2.8	12.0/19.9	1.3/2.4	57.5/19.9

Table 6.5: Accuracy of attribute learning across primitive actions (%). Each row represents a different primitive action and each column represents a different attribute.

	T	F	B	R	I	D
L	92.7	95.1	97.2	92.5	97.4	87.5
T	90.0	97.9	82.9	N/A	N/A	92.5
P	83.0	89.5	88.2	95.2	95.8	89.8

6.3.2 Skill Attribute Evaluation

I conduct the experiment on all three attribute learning approaches on videos, especially the later two approaches which are proposed by us and show significant attribute learning performance improvement in image domain.

Relative Attribute Learning

I verified the effectiveness of the learned attribute evaluator by the ranking accuracies. The ranking accuracy of each attribute is derived by 10-fold cross validation on the 150 labeled pairs in each primitive action, which is shown in Table 6.5. The result in the table demonstrates that our attribute evaluator, albeit learned only from relative information, has a high validity. In this experiment, only 3 primitive actions were considered here, i.e. L, T, and P. We combined segments of LM and UM with their corresponding subsequent operations of L, T and P, since LM and UM can be considered as the "preparation" step for the other operations. Some attributes are not considered intentionally for some actions (the "N/A" entries in Table 6.5) as it is not appropriate to assess the skills of the actions by these attributes. The result shows that the learned attribute learner achieves a significant high accuracy for our defined skill attributes.

Multi-task Relative Attribute Learning

I selected 10 representative videos from trainees of different skill levels, where each video is a full training session consisting of 12 Peg Transfer cycles, which leads to 12 video clips for each therblig. Thus we have in total 120 clips for each therblig. We manually label the relative rankings for 150 pairs of clips, following the guidelines provided by FLS (available on the FLS website). For each pair of clips, we label the attributes described in Tab. 6.1 as either "left is better than right", "right is better than left" or "unsure". Then five-fold random split (one fold for testing and remaining folds for training) is applied to evaluate the proposed method with the comparison to the other two methods. Due to space limitation, we only show the results of two therbligs "lift" and "transfer" in this work, which are presented in Tab. 6.6.

From Tab. 6.6, we can find that, the proposed method (Col 3) and MTRL (Col 4) obtained significantly better result than the relative attribute method (Col 5), except for the attribute "Bimanual dexterity" for therblig "Lift" (Tab. 6.6(A) Row 4) and the attribute "Depth perception" for therblig "Transfer" (Tab. 6.6 (B) Row 7). The improvement can be explained by the explicit consideration of intrinsic relatedness of those attributes in the proposed method and MTRL. The proposed method is on average better than MTRL, although MTRL achieves similar average accuracy in Tab. 6.6(B). This could be due to the fact that both the MTRL constraint and the proposed low-rank constraint did similarly well in capturing the correlation among the attributes for that particular action. However, as discussed earlier, the flexibility of the low-rank constraint in the proposed method would in general lead to a better performance, which is also evidenced by the overall better performance of the proposed method in Tab. VI (and in particular in (A)).

Hybrid Relative Attribute Learning

The training data were labeled in the following procedure. Given a random action clip as query, we randomly pick another two clips from the clips with significant higher attribute values. Then the expert surgeon would evaluate if the returned clips is instructive or not instructive. If both of the two clips are instructive, the expert would further provide a pair-wise label indicating which clip is more instructive. The labeling process follows the three criterions discussed above. For each of the 3 primitive actions, we generated 30 queries and 8 pairs of clips for each query.

Similarly, we illustrate the benefit of hybrid ranking SVM by comparison with both point-wise and pair-wise SVM. We train hybrid ranking SVM with both the point and pair labels acquired in the above process, while the point-wise and pair-wise SVM are trained with their corresponding labels. Tab. 6.7 shows the classification and the ranking accuracies of the hybrid-ranking SVM based on the combination of both point and pairwise labels, compared with the purely point-wise and pair-wise based approaches. Each entry shows the accuracy in lift/transfer/drop action. The results show that our hybrid approach provides better accuracy than the two baseline methods.

Table 6.6: The experimental result in evaluating motions skills of surgical simulations: (a) therblig "lift" and (b) therblig "transfer". Col 2 is the number of dissimilar pairs; Col 3 is the number of similar pair. Note for Attribute R and we of therbligs "transfer", we don't enough ground truth to compute the accuracy.

(a) Lift

Attribute	$\|\mathbf{E}\|$	$\|\mathbf{F}\|$	Proposed	MTRL	Relative
T	90	50	**78.89**%	72.22%	72.22%
F	77	63	**75.32**%	70.13%	67.53%
B	30	110	83.33%	**90.00**%	86.68%
R	62	78	**83.87**%	74.19%	61.29%
I	70	70	81.43%	**82.86**%	75.71%
D	29	111	**75.86**%	72.41%	62.07%
Overall	237	183	**79.61**%	75.70%	70.39%

(b) Transfer

Attribute	$\|\mathbf{E}\|$	$\|\mathbf{F}\|$	Proposed	MTRL	Relative
T	59	41	81.36%	**83.05**%	74.58%
F	46	54	78.26%	**82.61**%	73.91%
B	41	59	**65.85**%	58.54%	56.10%
R	1	99	N.A.	N.A.	N.A.
I	8	92	N.A.	N.A.	N.A.
D	41	59	80.49%	82.93%	**85.37**%
Overall	112	187	**77.55**%	77.04%	73.47%

Table 6.7: The classification and ranking accuracy (%) on surgical video data of three primitive actions.

Labels	Lift	Transfer	Drop
Point/Hybrid	78.1/80.7	83.7/84.4	83.4/88.4
Pair/Hybrid	75.7/76.3	90.0/89.8	89.3/90.4

Table 6.8: Subjective evaluation result of the instructive video retrieval(%).

	Instructive rate	Comparative rate
L	93.3/83.3	50.0/40.0/10.0
T	96.7/73.3	63.3/26.7/10.0
P	95.0/76.7	60.0/26.7/13.3

6.3.3 Instructive Video Evaluation

I compared our instructive video retrieval method with a baseline method that randomly selects one expert video clip of the primitive action. The comparison protocol is as follows. First, a query clip is selected from the database. Then, the recommended clips are obtained from both the proposed method and the baseline method. The two illustrative clips are paired in random order and presented to 8 human evaluators from the local hospital to judge which retrieved clip is more instructive. For each primitive operation, totally 60 queries are generated and the subjective evaluation result is summarized in Table 6.8. The "Instructive rate" shows the percentage of the retrieved videos are instructive of the proposed/baseline approach. The "comparative rate" is the percentage of our proposed approach retrieves a more/similar/less instructive video than the baseline approach does. The result shows that both methods present high instructive rate and the proposed method is persistently better than the baseline method. The result is especially satisfactory since the baseline method already employs an expert video, and thus our method is able to tell which expert video clip is more helpful to serve as an instructive reference. Since the proposed instructive video retrieval method is based on skill attribute analysis, this proves the validity of the attribute learning.

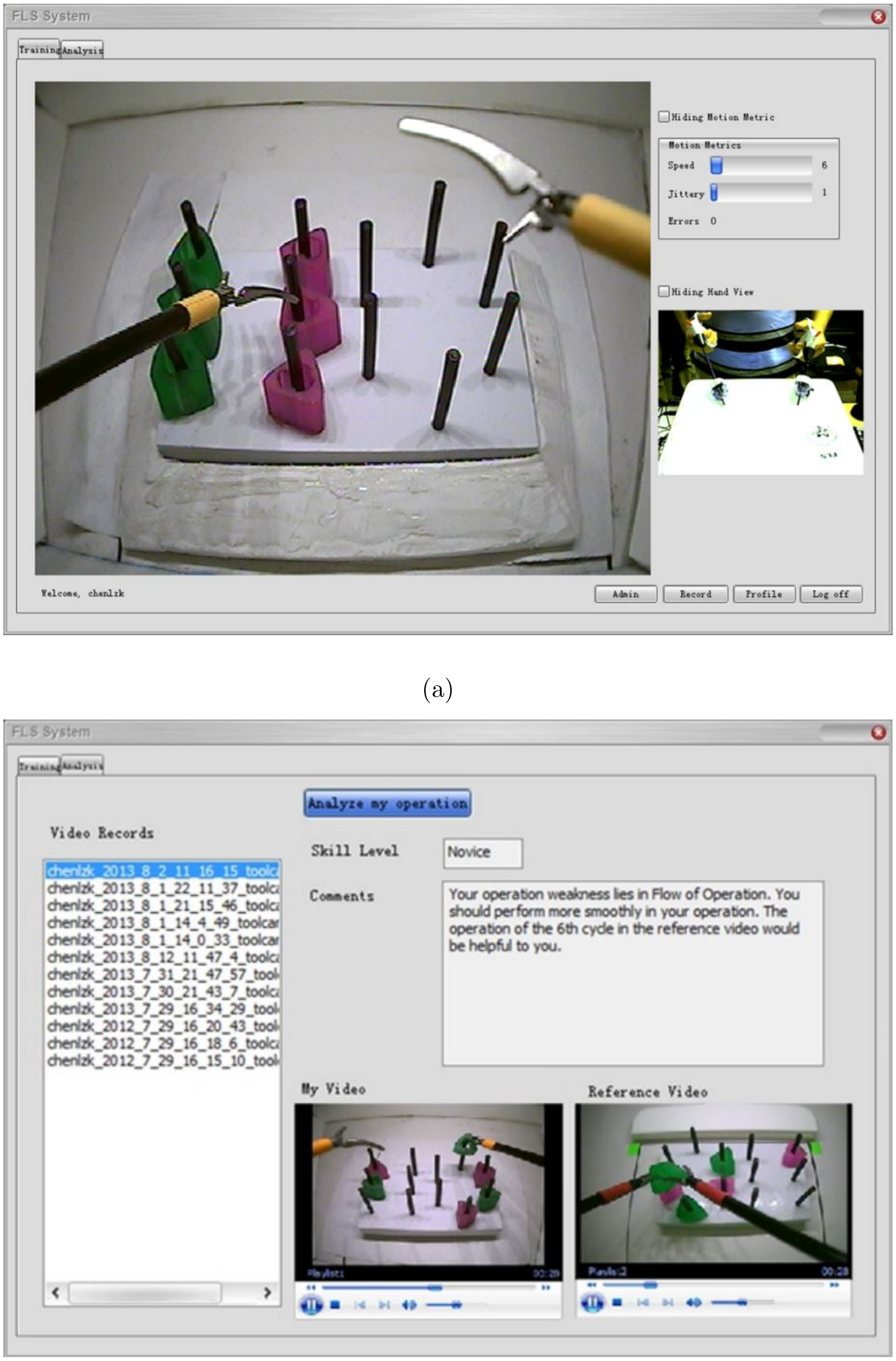
6.3.4 An Integrated Coaching System

I developed a prototype “Surgical Video Coaching System” based on our proposed framework. This system can be used accompanying the FLS box to recommend an instructive reference video for coaching purpose. Part of the system is illustrated in Fig. 7. The following three aspects are considered in our system:

Data archival: The original FLS box is only a pass-through system without memory. Our system stores a trainee’s practice sessionss, which can be used to support many capabilities including comparison of different sessions, enabling a trainee to review his/her errors, etc. The captured stream is indexed by the user’s identification (implemented by the registration and login module) and thus can be effectively retrieved.

Training Mode: It is desirable to allow a trainee to compare his/her performance between different practice sessions as this would provide insights as to how to improve. In the training mode, which is shown in Fig. 6.5 Left, the user can choose to capture the tool movements and hand movements of the training sessions. Besides, the system provides on-line feedback on the user’s operations regarding their speed, jittery and errors.

Analysis Mode: In the analysis mode shown in Fig. 6.5 Right, our system evaluates and analyzes the video selected by the user and provides feedback on the skill level and weakness (reflected by the attributes) in the operations. Also, a reference video for improving the worst attribute will be recommended to assist the operator’s skill improvement.



☐ Hiding Motion Metric

Motion Metrics

Speed 6

Jittery 1

Errors 0

☐ Hiding Hand View

Welcome, chenlzk

(a)

chenlzk_2013_8_2_11_16_15_toolc
chenlzk_2013_8_1_22_11_37_toolc
chenlzk_2013_8_1_21_15_46_toolc
chenlzk_2013_8_1_14_4_49_toolcar
chenlzk_2013_8_1_14_0_33_toolcar
chenlzk_2013_8_12_11_47_4_toolc
chenlzk_2013_7_31_21_47_57_tool
chenlzk_2013_7_30_21_43_7_toolc
chenlzk_2013_7_29_16_34_29_tool
chenlzk_2012_7_29_16_20_43_tool
chenlzk_2012_7_29_16_18_6_toolc
chenlzk_2012_7_29_16_15_10_tool

Skill Level: Novice

Comments: Your operation weakness lies in Flow of Operation. You should perform more smoothly in your operation. The operation of the 6th cycle in the reference video would be helpful to you.

(b)

Figure 6.5: Demonstration of training module (UP) and Analysis module (DOWN) of our developed platform. In the training mode, tool movement and hand movement are captured by two cameras; motion metrics of speed and jittery are calculated online every second. In the analysis mode, our platform will analyze the video selected by the user and give relative feedback about the skill level, weakness of operation and retrieve a helpful reference video.

6.4 Conclusions

In this chapter, we presented a video-based surgical skill coaching system, aiming at providing a weakness specific, skill superior and content similar instructive feedback. To build the system, we proposed a new problem defined as instructive video retrieval together with a effective framework to solve the problem by borrowing the idea of image attribute learning into video domain for high-level skill understanding. In building the system, algorithmic innovations were made to incorporate domain knowledge and to handle practical difficulties arising from the real training platform. To our knowledge, this is the first video-based approach to delivering a systematic solution to the problem of automated skill coaching in simulation-based surgical training. Experiments with real world videos capturing the training sessions of resident surgeons have demonstrated the effectiveness of the idea and the key algorithms.

Chapter 7

CONCLUSION AND FUTURE WORK

In this chapter, I summarize my research work and discuss some promising research directions for robust attribute learning.

7.1 Conclusion

In this dissertation, I presented my exploration on semantic visual attribute prediction/ranking to bridge the gap between low-level visual representations and high-level semantic understanding. My work mainly focused on the following two research problems: (1) how to learn a robust and accurate attribute predictor/ranker given very limited labels from a large amount of semantic attributes; (2) How to extend the idea of image-based visual attributes to video-related applications. To answer these two questions, I conducted my research on four different tasks.

My first work was proposed to rank the images based on the high-level semantic attributes by exploring the correlation among different attributes. For a problem involving multiple attributes, it is reasonable to assume that utilizing such relatedness among the attributes would benefit learning, especially when the number of labeled training pairs are very limited. I proposed a relative multi-attribute learning framework that integrates relative attributes into a multi-task learning scheme. The formulation allows us to exploit the advantages of the state-of-the-art regularization-based approaches in multi-task learning for improved attribute learning, which leads to significant improvements over learning each attribute independently.

My second work aims to learn a robust attribute predictor by exploring the structure of the correlation among different attributes. Observing that the performance of attribute prediction greatly relies on the task structure, I proposed a new approach that can automatically detect problem-specific clustering structures of the attributes for improved attribute prediction. Since obtaining a good representation of semantic attributes usually requires learning from high-dimensional low-level features, my approach utilizes K-means and matrix factorization for attribute structure discovery and group sparsity regularizers for joint feature selection. The approach achieved significant performance gains over other state-of-the-art approaches for attribute prediction.

My third work explores the correlation among different types of labels. Since pointwise data and pairwise data have different advantages and limitations in terms of data availability, labeling complexity and representational capability, I proposed a hybrid learning strategy that fuses knowledge from both pointwise and pairwise training data into one framework for attribute-based (adaptive personalized) image ranking where the ranking performance is updated online based on user feedback. By minimizing the ranking margin by knowledge from both types of labels, my approach outperforms other state-of-the-art approaches which learn each type of label independently.

My last work extends the semantic attributes into video related applications. I present a video-based skill coaching system for simulation-based surgical training. My approach explores a newly-proposed task of instructive video retrieval. By introducing attribute learning into video for high-level skill understanding, I aimed at providing automated feedback and providing an instructive video, to which the trainees can refer for performance improvement. This is achieved by ensuring the feedback is weakness-specific, skill-superior and content-similar. A suite of techniques was integrated to build the coaching system with these features. In particular, algorithms were developed for action segmentation, video attribute learning, and attribute-based video retrieval. Experiments with realistic surgical videos demonstrate the feasibility of the proposed method.

My proposed work was evaluated on both synthetic and real-world data, and the results have shown that my approaches delivered significant performance improvements in various attribute learning problems.

7.2 Future Work

The general research on attribute learning is still in its early stage and my work can be further extended. In the following, I point out some possible future research directions for robust attribute prediction/ranking.

First, my recent work models the structure of attribute correlation by clustering, which groups similar attributes together for learning. However, the attribute correlation in real-world applications may be more complicated. Thus, using more complex structure representations, e.g., tree structure, may further push forward the prediction/ranking performance of attribute learning.

Second, current visual media involves multiple types of low-level representation, e.g., color histogram describing appearance information and SIFT features describing geometric information. Simply concatenating these low-level feature representations for high-level semantic attribute learning may not achieve the optimal solution due to the replicate information and inconsistent noise. Multi-view learning approaches, which learn models by capturing the correlation among different views, can be adopted for improved attribute learning.

Third, my research for video-based semantic attribute learning mainly focused on surgical skill evaluation. There are other challenging practical problems, such as video event understanding, video retrieval and recommendation, that can potentially benefit from reliable visual semantic attribute analysis.

REFERENCES

[1] Columbia Object Image Library (COIL-100). Technical report, Columbia University, 1996.

[2] Dimitrios S. Alexiadis, Philip Kelly, Petros Daras, Noel E. O'Connor, Tamy Boubekeur, and Maher Ben Moussa. Evaluating a dancer's performance using kinect-based skeleton tracking. In *Proc. of ACM MM'11*, 2011.

[3] Rie Kubota Ando and Tong Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *J. Mach. Learn. Res.*, December 2005.

[4] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Multi-task feature learning. *J. Mach. Learn. Res.*, 73(3):243–272, December 2008.

[5] Francis R. Bach. Clustered multi-task learning: a convex formulation.

[6] Francis R. Bach. Consistency of the group lasso and multiple kernel learning. *J. Mach. Learn. Res.*, 9:1179–1225, June 2008.

[7] Bart Bakker and Tom Heskes. Task clustering and gating for bayesian multitask learning. *J. Mach. Learn. Res.*, December 2003.

[8] Douglas A. Baxter and John H. Byrne. Simulator for neural networks and action potentials. November 2007.

[9] Tamara L. Berg, Alexander C. Berg, and Jonathan Shih. Automatic attribute discovery and characterization from noisy web data. In *Proc.*, ECCV 2010.

[10] John Blitzer, Ryan McDonald, and Fernando Pereira. Domain adaptation with structural correspondence learning. EMNLP '06, pages 120–128, 2006.

[11] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, January 2011.

[12] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization.* Cambridge University Press, New York, NY, USA, 2004.

[13] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proc. of ICML '05*, pages 89–96. ACM, 2005.

[14] Jian-Feng Cai, Emmanuel J Candès, and Zuowei Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[15] Xiao Cai, Feiping Nie, and Heng Huang. Exact top-k feature selection via l2,0-norm constraint. In *IJCAI '13*, pages 1240–1246, 2013.

[16] Yunbo Cao, Jun Xu, Tie-Yan Liu, Hang Li, Yalou Huang, and Hsiao-Wuen Hon. Adapting ranking svm to document retrieval. In *Proc. of SIGIR '06*, pages 186–193. ACM, 2006.

[17] Rich Caruana. Multitask learning, 1997.

[18] Jianhui Chen, Jiayu Zhou, and Jieping Ye. Integrating low-rank and group-sparse structures for robust multi-task learning. In *Proc.*, KDD '11, 2011.

[19] Lin Chen, Qiang Zhang, and Baoxin Li. Predicting multiple attributes via relative multi-task learning. In *Proc. of CVPR'14*, pages 1027–1034, June 2014.

[20] Xi Chen, Seyoung Kim, Qihang Lin, Jaime G. Carbonell, and Eric P. Xing. Graph-structured multi-task regression and an efficient optimization method for general fused lasso, 2010.

[21] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* Wiley, 1991.

[22] D. Coyle. *The Talent Code: Greatness Isn't Born. It's Grown. Here's How.* Random House Publishing Group, 2009.

[23] Antonio Criminisi, Jamie Shotton, and Ender Konukoglu. Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Found. Trends. Comput. Graph. Vis.*, February 2012.

[24] Jeffrey D. Doyle, Eric M. Webber, and Ravi S. Sidhu. A universal global rating scale for the evaluation of technical skills in the operating room. *The American Journal of Surgery*, 2007.

[25] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification.* John Wiley & Sons, New York, 2 edition, 2001.

[26] Najeeb Elahi, Randi Karlsen, and Einar J. Holsbø. Personalized photo recommendation by leveraging user modeling on social network. In *Proc. of IIWAS '13*, pages 68:68–68:71. ACM, 2013.

[27] S. Essid, D. Alexiadis, R. Tournemenne, M. Gowing, P. Kelly, D. Monaghan, P. Daras, A. Dremeau, and N.E. O'Connor. An advanced virtual dance performance evaluator. In *Proc. of ICASSP'12*, 2012.

[28] T. Evgeniou and M. Pontil. Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 109–117. ACM, 2004.

[29] Mark Fanty and Ronald Cole. Spoken letter recognition. In *NIPS '91*, pages 220–226. 1991.

[30] A. Farhadi, I. Endres, and D. Hoiem. Attribute-centric recognition for cross-category generalization. In *Proc. of CVPR'10*, pages 2352–2359, June 2010.

[31] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1778–1785, June 2009.

[32] Vittorio Ferrari and Andrew Zisserman. Learning visual attributes. In *Proc. of NIPS'08*, pages 433–440. 2008.

[33] Yoav Freund, Raj Iyer, Robert E. Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. *J. Mach. Learn. Res.*, 4:933–969, December 2003.

[34] A.S. Georghiades, P.N. Belhumeur, and D.J. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(6):643–660, 2001.

[35] Pinghua Gong, Jieping Ye, and Changshui Zhang. Robust multi-task feature learning. In *Proc.*, KDD '12, 2012.

[36] Yahong Han, Yi Yang, and Xiaofang Zhou. Co-regularized ensemble for feature selection. In *IJCAI '13*, pages 1380–1386, 2013.

[37] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *Information Theory, IRE Transactions on*, February 1962.

[38] Jin Huang, Feiping Nie, Heng Huang, and Chris Ding. Robust manifold nonnegative matrix factorization. *ACM Trans. Knowl. Discov. Data*, 8(3):11:1–11:21, June 2014.

[39] J. J. Hull. A database for handwritten text recognition research. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(5):550–554, May 1994.

[40] Laurent Jacob and Guillaume Obozinski. Group lasso with overlap and graph lasso.

[41] Laurent Jacob, Jean philippe Vert, and Francis R. Bach. Clustered multi-task learning: A convex formulation. In *Proc. of NIPS'09*, pages 745–752. 2009.

[42] Ali Jalali, Pradeep D. Ravikumar, and Sujay Sanghavi. A dirty model for multiple sparse regression. *CoRR*, 2011.

[43] D. Jayaraman, Fei Sha, and K. Grauman. Decorrelating semantic visual attributes by resisting the urge to share. In *Proc. of CVPR'14*, pages 1629–1636, June 2014.

[44] Shuiwang Ji and Jieping Ye. An accelerated gradient method for trace norm minimization. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 457–464. ACM, 2009.

[45] Yu Jin, Xiaoxiang Hu, and GangShan Wu. A tai chi training system based on fast skeleton matching algorithm. In *Proc. of ECCV'12*, 2012.

[46] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proc. of KDD '02*, pages 133–142. ACM, 2002.

[47] Seyoung Kim and Eric P. Xing. Tree-guided group lasso for multi-task regression with structured sparsity.

[48] George Kimeldorf and Grace Wahba. Some results on tchebycheffian spline functions. *Journal of Mathematical Analysis and Applications*, 33(1):82 – 95, 1971.

[49] Seung kook Jun, M.S. Narayanan, P. Agarwal, A. Eddib, P. Singhal, S. Garimella, and V. Krovi. Robotic minimally invasive surgical skill assessment based on automated video-analysis motion studies. In *Proc. of BioRob'12*, June 2012.

[50] A Kovashka and K. Grauman. Attribute adaptation for personalized image search. In *Proc. of ICCV '13*, pages 3432–3439, Dec 2013.

[51] A. Kovashka, D. Parikh, and K. Grauman. Whittlesearch: Image search with relative attribute feedback. In *Proc. of CVPR'12*, pages 2973–2980, June 2012.

[52] N. Kumar, A.C. Berg, P.N. Belhumeur, and S.K. Nayar. Attribute and simile classifiers for face verification. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 365 –372, 29 2009-oct. 2 2009.

[53] Neeraj Kumar, Peter Belhumeur, and Shree Nayar. Facetracer: A search engine for large collections of images with faces. In *Proc. of ECCV '08*, pages 340–353, 2008.

[54] C.H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Proc. of CVPR '09*, pages 951–958, June 2009.

[55] Rasmus Munk Larsen. Lanczos bidiagonalization with partial reorthogonalization. *DAIMI PB*, 27(537), 1998.

[56] Ping Li, Qiang Wu, and Christopher J. Burges. Mcrank: Learning to rank using multiple classification and gradient boosting. In *Proc. of NIPS '08*, pages 897–904. Curran Associates, Inc., 2008.

[57] Zhouchen Lin, Minming Chen, and Yi Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint arXiv:1009.5055*, 2010.

[58] H. Liu and H. Motoda, editors. *Computational Methods of Feature Selection.* Chapman & Hall, 2008.

[59] Jiajun Liu, Zi Huang, Hongyun Cai, Heng Tao Shen, Chong Wah Ngo, and Wei Wang. Near-duplicate video retrieval: Current research and future trends. *ACM Comput. Surv.*, August 2013.

[60] Jun Liu, Shuiwang Ji, and Jieping Ye. Multi-task feature learning via efficient l2,1-norm minimization, 2009.

[61] Ulrike Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, December 2007.

[62] Sara Memar, LillySuriani Affendey, Norwati Mustapha, ShyamalaC. Doraisamy, and Mohammadreza Ektefa. An integrated semantic-based approach in concept based video retrieval. *Multimedia Tools and Applications*, 2013.

[63] Charles A. Micchelli and Massimiliano Pontil. Regularized multi-task learning. 2004.

[64] Taesup Moon, Alex Smola, Yi Chang, and Zhaohui Zheng. Intervalrank: Isotonic regression with listwise and pairwise constraints. In *WSDM*, 2010.

[65] Krishna Moorthy, Yaron Munz, Sudip K Sarker, and Ara Darzi. Objective assessment of technical skills in surgery. *BMJ*, 10 2003.

[66] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

[67] D. Parikh and K. Grauman. Relative attributes. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 503 –510, nov. 2011.

[68] D. Parikh and K. Grauman. Relative attributes. In *Proc. of ICCV '11*, pages 503–510, Nov 2011.

[69] Amar Parkash and Devi Parikh. Attributes for classifier feedback. In *Proc.*, ECCV 2012. 2012.

[70] Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *Proc. of CVPR'12*, 2012.

[71] F. Ding C. Peng, H. Long. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(8):1226–1238, 2005.

[72] A. Quattoni, M. Collins, and T. Darrell. Learning visual representations using images with captions. In *CVPR '07*, pages 1–8, June 2007.

[73] C. Renjifo and C. Carmen. The discounted cumulative margin penalty: Rank-learning with a list-wise loss and pair-wise margins. In *MLSP*, Sept 2012.

[74] Richard Reznick, Glenn Regehr, Helen MacRae, Jenepher Martin, and Wendy McCulloch. Testing technical skill via an innovative bench station examination. *The American Journal of Surgery*, 1997.

[75] J. Rosen, J.D. Brown, L. Chang, M.N. Sinanan, and B. Hannaford. Generalized approach for modeling minimally invasive surgery as a stochastic process using a discrete markov model. *Biomedical Engineering, IEEE Transactions on*, March 2006.

[76] J. Rosen, B. Hannaford, C.G. Richards, and M.N. Sinanan. Markov modeling of minimally invasive surgery based on tool/tissue interaction and force/torque signatures for evaluating surgical skills. *Biomedical Engineering, IEEE Transactions on*, May 2001.

[77] F.S. Samaria and A.C. Harter. Parameterisation of a stochastic model for human face identification. In *Applications of Computer Vision' 94*, pages 138–142, Dec 1994.

[78] Jitao Sang, Changsheng Xu, and Dongyuan Lu. Learn to personalized image search from the photo sharing websites. *Multimedia, IEEE Transactions on*, 14(4):963–974, Aug 2012.

[79] W.J. Scheirer, N. Kumar, P.N. Belhumeur, and T.E. Boult. Multi-attribute spaces: Calibration for attribute fusion and similarity search. In *Proc. of CVPR'12*, pages 2933–2940, June 2012.

[80] D. Sculley. Combined regression and ranking. In *Proc. of SIGKDD'10*, pages 979–988, New York, NY, USA, 2010. ACM.

[81] Shai Shalev-Shwartz, Yoram Singer, and Nathan Srebro. Pegasos: Primal estimated sub-gradient solver for svm. In *Proc. of ICML '07*, pages 807–814. ACM, 2007.

[82] Amnon Shashua and Anat Levin. Ranking with large margin principle: Two approaches. In *Proc. of NIPS '03*, pages 961–968. MIT Press, 2003.

[83] B. Siddiquie, R.S. Feris, and L.S. Davis. Image ranking and retrieval based on multi-attribute queries. In *Proc. of CVPR'11*, pages 801–808, June 2011.

[84] Daniel L. Silver and Robert E. Mercer. The task rehearsal method of sequential learning.

[85] Fengyi Song, Xiaoyang Tan, and Songcan Chen. Exploiting relationship between attributes for improved face verification. In *Proc. of BMVC'12*, pages 27.1–27.11, 2012.

[86] Jiliang Tang, Salem Alelyani, and Huan Liu. Feature selection for classification: A review. *Data Classification: Algorithms and Applications. Editor: Charu Aggarwal, CRC Press In Chapman & Hall/CRC Data Mining and Knowledge Discovery Series*, 2014.

[87] Jiliang Tang and Huan Liu. Feature selection with linked data in social media. In *SDM '12*, pages 118–128. SIAM, 2012.

[88] K. Tervo, L. Palmroth, and H. Koivo. Skill evaluation of human operators in partly automated mobile working machines. *Automation Science and Engineering, IEEE Transactions on*, Jan 2010.

[89] Ming-Feng Tsai, Tie-Yan Liu, Tao Qin, Hsin-Hsi Chen, and Wei-Ying Ma. Frank: A ranking method with fidelity loss. In *Proc. of SIGIR '07*, pages 383–390. ACM, 2007.

[90] Daniel Vaquero, Rogerio Feris, Duan Tran, Lisa Brown, Arun Hampapur, and Matthew Turk. Attribute-based people search in surveillance environments. In *Proc. of WACV'09*, December 2009.

[91] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, 2011.

[92] Suhang Wang, Jiliang Tang, and Huan Liu. Embedded unsupervised feature selection, 2015.

[93] Yang Wang and Greg Mori. A discriminative latent model of object classes and attributes. In *Proc. of ECCV'10*, pages 155–168, 2010.

[94] Ya Xue, Xuejun Liao, Lawrence Carin, and Balaji Krishnapuram. Multi-task learning for classification with dirichlet process priors. *Journal of Machine Learning Research*, 2007.

[95] Kai Yu, Volker Tresp, and Shipeng Yu. A nonparametric hierarchical bayesian framework for information filtering. In *Proc.*, SIGIR '04, 2004.

[96] Hongyuan Zha, Xiaofeng He, Chris Ding, Ming Gu, and Horst D. Simon. Spectral relaxation for k-means clustering. In *Proc. of NIPS'02*, pages 1057–1064. 2002.

[97] Tingting Zhang and Jun S. Liu. Nonparametric hierarchical bayes analysis of binomial data via bernstein polynomial priors. *Canadian Journal of Statistics*, 2012.

[98] Jiayu Zhou, Jianhui Chen, and Jieping Ye. Clustered multi-task learning via alternating structure optimization. In *Advances in Neural Information Processing Systems 24*. 2011.

APPENDIX A

ALGORITHM FOR RELATIVE ATTRIBUTE LOW-RANK SUBSPACE LEARNING

According to [28], Eqn. 3.9 is equivalent to the following problem with appropriate parameters $(\lambda, \rho_1, \rho_2)$:

$$\begin{aligned}
\min_{\mathbf{W},\epsilon,\gamma} \quad & \sum_{t}^{T} \frac{1}{2}|\mathbf{V}_t|_2^2 + \frac{\lambda}{2}|\mathbf{W}_0|_2^2 + \rho_1 \sum_{(i,j)\in\mathbb{E}_t} \epsilon_{ij}^t + \rho_2 \sum_{(i,j)\in\mathbb{F}_t} \gamma_{ij}^t \\
\text{s.t.} \quad & (\mathbf{V}_t + \mathbf{W}_0)^\top(\mathbf{X}_{it} - \mathbf{X}_{jt}) + \epsilon_{ij}^t \geq 1 \\
& -\gamma_{ij}^t \leq (\mathbf{V}_t + \mathbf{W}_0)^\top(\mathbf{X}_{it} - \mathbf{X}_{jt}) \leq \gamma_{ij}^t \\
& \epsilon_{ij}^t \geq 0; \gamma_{ij}^t \geq 0
\end{aligned} \tag{A.1}$$

with $\mathbf{W}_t = \mathbf{V}_t + \mathbf{w}_0$. According to [28], we can also define the following mapping functions:

$$\Phi(\mathbf{X}_{it}) = [\sqrt{\frac{1}{T\lambda}}\mathbf{X}_{it}, 0, \cdots, 0, \mathbf{X}_{it}, 0, \cdots, 0] \tag{A.2}$$

$$\Phi(\mathbf{W}) = [\sqrt{T\lambda}\mathbf{w}_0, \mathbf{V}_1, \cdots, \mathbf{V}_t, \cdots, \mathbf{V}_T] \tag{A.3}$$

and get the following formulations:

$$\begin{aligned}
\min_{\mathbf{W},\epsilon,\gamma} \quad & \frac{1}{2}|\Phi(\mathbf{W})|_2^2 + \sum_{t}^{\top} \rho_1 \sum_{(i,j)\in\mathbb{E}_t} \epsilon_{ij}^t + \rho_2 \sum_{(i,j)\in\mathbb{F}_t} \gamma_{ij}^t \\
\text{s.t.} \quad & \Phi^\top(\mathbf{W})\Phi(\mathbf{X}_{it} - \mathbf{X}_{jt}) + \epsilon_{ij}^t \geq 1 \\
& -\gamma_{ij}^t \leq \Phi^\top(\mathbf{W})\Phi(\mathbf{X}_{it} - \mathbf{X}_{jt}) \leq \gamma_{ij}^t \\
& \epsilon_{ij}^t \geq 0; \gamma_{ij}^t \geq 0
\end{aligned} \tag{A.4}$$

Obviously

$$\begin{aligned}
|\Phi(\mathbf{W})|_2^2 = \sum_{t}^{\top} (|\mathbf{V}_t|_2^2 + \lambda|\mathbf{w}_0|_2^2) \\
\Phi^\top(\mathbf{W})\Phi(\mathbf{X}_{it} - \mathbf{X}_{jt}) = (\mathbf{V}_t + \mathbf{W}_0)^\top(\mathbf{X}_{it} - \mathbf{X}_{jt})
\end{aligned} \tag{A.5}$$

By writing $(\mathbf{X}_i - \mathbf{X}_j) = \mathbf{Y}_k$ for $(i,j) \in \mathbb{E}$ and $(\mathbf{X}_i - \mathbf{X}_j) = \mathbf{Z}_l$ for $(i,j) \in \mathbb{F}$ and applying Lagrange multipliers we, can get the dual form of Eqn. 3.9:

$$\begin{aligned}
\min_{\alpha,\beta,\lambda} \quad & \frac{1}{2}|\sum_{k} \alpha_k \Phi(\mathbf{Y}_k) + \sum_{l} (\delta_l - \beta_l)\Phi(\mathbf{Z}_l)|_2^2 - \sum_{k} \alpha_k \\
\text{s.t.} \quad & 0 \leq \beta_l, \delta_l \leq \rho_2 \\
& 0 \leq \alpha_k \leq \rho_1 \\
& 0 \leq \beta_l + \delta_l \leq \rho_2
\end{aligned} \tag{A.6}$$

which can be written as the following quadratic programming problem:

$$\begin{aligned} \min_{\mathbf{u}} \quad & \frac{1}{2}\mathbf{u}^\top \mathbf{K}\mathbf{u} + \mathbf{f}^\top \mathbf{u} \\ \text{s.t.} \quad & \mathbf{lb} \leq z \leq \mathbf{ub} \\ & \mathbf{A}\mathbf{u} \leq \mathbf{b} \end{aligned} \tag{A.7}$$

with

$$\mathbf{u} = [\alpha^\top, -\beta^\top, \delta^\top]^\top \in \mathbb{R}^{T(|\mathbb{E}|+2|\mathbb{F}|)\times 1} \tag{A.8}$$

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{|\mathbb{E}|\times|\mathbb{E}|} & \mathbf{K}_{|\mathbb{E}|\times|\mathbb{F}|} & \mathbf{K}_{|\mathbb{E}|\times|\mathbb{F}|} \\ \mathbf{K}_{|\mathbb{F}|\times|\mathbb{E}|} & \mathbf{K}_{|\mathbb{F}|\times|\mathbb{F}|} & \mathbf{K}_{|\mathbb{F}|\times|\mathbb{F}|} \\ \mathbf{K}_{|\mathbb{F}|\times|\mathbb{E}|} & \mathbf{K}_{|\mathbb{F}|\times|\mathbb{F}|} & \mathbf{K}_{|\mathbb{E}|\times|\mathbb{F}|} \end{bmatrix} \tag{A.9}$$

$$\mathbf{f} = [-\mathbf{e}_{|\mathbb{E}|}^\top, 0\mathbf{e}_{|\mathbb{F}|}^\top, 0\mathbf{e}_{|\mathbb{F}|}^\top]^\top \tag{A.10}$$

$$\mathbf{lb} = [0\mathbf{e}_{|\mathbb{E}|}^\top, -\rho_2\mathbf{e}_{|\mathbb{F}|}^\top, 0\mathbf{e}_{|\mathbb{F}|}^\top]^\top \tag{A.11}$$

$$\mathbf{ub} = [\rho_1\mathbf{e}_{|\mathbb{E}|}^\top, 0\mathbf{e}_{|\mathbb{F}|}^\top, \rho_2\mathbf{e}_{|\mathbb{F}|}^\top]^\top \tag{A.12}$$

$$\mathbf{A} = [0_{|\mathbb{E}|\times|\mathbb{E}|}, -\mathbf{I}_{|\mathbb{F}|\times|\mathbb{F}|}, \mathbf{I}_{|\mathbb{F}|\times|\mathbb{F}|}] \tag{A.13}$$

$$\mathbf{b} = \rho_2\mathbf{e}_{|\mathbb{F}|} \in \mathbb{R}^{T|\mathbb{F}|\times 1} \tag{A.14}$$

$$\mathbf{K}_{|\mathbb{E}|\times|\mathbb{E}|}(i,t;j,s) = \Phi^\top(\mathbf{y}_{it})\Phi(\mathbf{y}_{js}) \tag{A.15}$$

$$\mathbf{K}_{|\mathbb{E}|\times|\mathbb{F}|}(i,t;j,s) = \Phi^\top(\mathbf{y}_{it})\Phi(\mathbf{z}_{js}) \tag{A.16}$$

$$\mathbf{K}_{|\mathbb{F}|\times|\mathbb{E}|}(i,t;j,s) = \Phi^\top(\mathbf{z}_{it})\Phi(\mathbf{z}_{js}) \tag{A.17}$$

where $\mathbf{e}_n \in \mathbb{R}^{n\times 1}$ is a all 1 vector, $0_{m\times n} \in \mathbb{R}^{m\times n}$ is all 0 matrix, $\mathbf{I}_{n\times n} \in \mathbb{R}^{n\times n}$ is identity matrix. The mapping function $\Phi(\cdot)$ is defined in Eqn. A.2.

After we solve the quadratic problem in Eqn. A.7 with optimal solution $u^* = [\alpha^\top, -\beta^\top, \delta^\top]^\top$, we can compute

$$\Phi(\mathbf{W}) = [\frac{1}{\lambda}\mathbf{W}_0^\top, \mathbf{V}_1^\top, \cdots, \mathbf{V}_t^\top]^\top = \sum_t \sum_i \alpha_{it}\Phi(\mathbf{Y}_{it}) + \sum_j (\delta_{jt} - \beta_{jt})\Phi(\mathbf{Z}_{jt}) \tag{A.18}$$

and then recover classifier of each attribute as $\mathbf{W}_t = \mathbf{W}_0 + \mathbf{V}_t$.

As the proposed method can be formulated into a quadratic programming problem, the convergence and global optimality of the solution is guaranteed. The dimension of quadratic programming problem is $T(|\mathbb{E}| + 2|\mathbb{F}|) \times 1$ with T as the number of tasks, $|\mathbb{E}|$ and $|\mathbb{F}|$ as the number of constraints cast by relative rankings. The dimension of the problem and the computational cost could be high, when there are a lot of pairs of relative rankings. To solve this issue, we could utilize the idea of active constraints.

Convergence Analysis

We will show that the proposed algorithm will converge. In this section, we will use $\mathbf{Y}^k$ to represent the variable $\mathbf{Y}$ computed in k_{th} iteration. First, we can easily identify that, the two sub-problems, "low rank problem" and "classification" problem are convex. We define the space

$$\begin{aligned}\Omega = \{\mathbf{Z}, \mathbf{W}, \mathbf{b}, \epsilon, \gamma | \mathbf{W}_t^\top(\mathbf{X}_{it} - \mathbf{X}_{jt}) + \epsilon_{ij}^t \geq 1 \\ \& \ -\gamma_{ij}^t \leq \mathbf{W}_t^\top(\mathbf{X}_{it} - \mathbf{X}_{jt}) \leq \gamma_{ij}^t \& \ \epsilon_{it} \geq 0 \\ \& \ \gamma_{it} \geq 0 \ \forall i, t\}\end{aligned} \tag{A.19}$$

which is obvious convex, and the analysis will be within this space.

Lemma 4. *$\mathbf{Y}^k$ is bounded.*

Proof. Since $\mathbf{Z}^{k+1}$ is optimal for the low-rank problem with $\mathbf{W}^k$, $\mathbf{b}^k$, ϵ^k, γ^k, μ^k and $\mathbf{Y}^k$, we have

$$0 \in \frac{\partial L(\mathbf{Z}, \mathbf{W}^k, \mathbf{b}^k, \epsilon^k, \mu^k, \mathbf{Y}^k)}{\partial \mathbf{Z}} = \frac{\partial \|\mathbf{Z}\|_*}{\partial \mathbf{Z}} - \mathbf{Y}^k + \mu^k(\mathbf{Z}^k - \mathbf{W}^k) \tag{A.20}$$

so we have

$$\mathbf{Y}^{k+1} = \mathbf{Y}^k - \mu^k(\mathbf{Z}^k - \mathbf{W}^k) \in \frac{\partial \|\mathbf{Z}\|_*}{\partial \mathbf{Z}} \tag{A.21}$$

According to [57] Theorem 4 and Lemma 1, $\mathbf{Y}^{k+1}$ is bounded. This ends the proof of Lemma 1. □

Lemma 5. *The sequences $\mathbf{Z}^k$, $\mathbf{W}^k$, $\mathbf{b}^k$, ϵ^k, μ^k will converge to the optimal solution.*

Proof. we define

$$f(\mathbf{W}, \mathbf{b}, \epsilon) = \lambda |\mathbf{W}|_* + \frac{1}{2} \sum_t |\mathbf{W}_t|_2^2 + \rho \sum_i \epsilon_{it} \tag{A.22}$$

as the objective function of the primal problem, we have:

$$\begin{aligned} & L(\mathbf{Z}^{k+1}, \mathbf{W}^{k+1}, \mathbf{b}^{k+1}, \epsilon^{k+1}, \gamma^{k+1}, \mu^k, \mathbf{Y}^k) && \text{(A.23)} \\ = \ & \min_{\mathbf{Z}, \mathbf{W}, \mathbf{b}, \epsilon} L(\mathbf{Z}, \mathbf{W}, \mathbf{b}, \epsilon, \gamma, \mu^k, \mathbf{Y}^k) && \text{(A.24)} \\ \leq \ & \min_{\mathbf{Z} = \mathbf{W}, \mathbf{b}, \epsilon} L(\mathbf{Z}, \mathbf{W}, \mathbf{b}, \epsilon, \gamma, \mu^k, \mathbf{Y}^k) && \text{(A.25)} \\ \leq \ & \min_{\mathbf{Z} = \mathbf{W}, \mathbf{b}, \epsilon} f(\mathbf{W}, \mathbf{b}, \epsilon, \gamma) && \text{(A.26)} \\ = \ & f^* && \text{(A.27)} \end{aligned}$$

with

$$\mathbf{Z}^{k+1} - \mathbf{W}^{k+1} = \frac{1}{\mu^k}(\mathbf{Y}^{k+1} - \mathbf{Y}^k) \tag{A.28}$$

So we have

$$\lim_{t \to \infty} \mathbf{Z}^k - \mathbf{W}^k = 0 \tag{A.29}$$

Thus

$$(\mathbf{W}^*, \mathbf{b}^*, \epsilon^*) = \lim_{t\to\infty} (\mathbf{W}^k, \mathbf{b}^k, \epsilon^k) \tag{A.30}$$

is the feasible solution of the primal problem.

In addition, we have

$$\begin{aligned}
& f(\mathbf{W}^{k+1}, \mathbf{b}^{k+1}, \epsilon^{k+1}, \gamma^{k+1}) \\
= \; & L(\mathbf{Z}^{k+1}, \mathbf{W}^{k+1}, \mathbf{b}^{k+1}, \epsilon^{k+1}, \gamma^{k+1}, \mu^k, \mathbf{Y}^k) \\
& -\frac{1}{2\mu^k}(|\mathbf{Y}^{k+1}|_F^2 - |\mathbf{Y}^k|_F^2) + |\mathbf{W}^{k+1}|_* - |\mathbf{Z}^{k+1}|_* \\
\leq \; & f^* - \frac{1}{2\mu^k}(|\mathbf{Y}^{k+1}|_F^2 - |\mathbf{Y}^k|_F^2) - |\mathbf{W}^{k+1} - \mathbf{Z}^{k+1}|_* \\
\leq \; & f^* - \frac{1}{2\mu^k}(|\mathbf{Y}^{k+1}|_F^2 - |\mathbf{Y}^k|_F^2) - \frac{1}{\mu^k}|\mathbf{Y}^{k+1} - \mathbf{Y}^k|_* \\
= \; & f^* - O(\frac{1}{\mu^k})
\end{aligned} \tag{A.31}$$

where for the last step, we use the boundedness of $\mathbf{Y}^k$ (Lemma 1). Thus we have

$$\lim_{t\to\infty} [f(\mathbf{W}^{k+1}, \mathbf{b}^{k+1}, \epsilon^{k+1})] = \lim_{t\to\infty} f^* - O(\frac{1}{\mu^k}) = f^* \tag{A.32}$$

Besides, by $|\mathbf{Z}|_* \geq |\mathbf{W}|_* - |\mathbf{Z} - \mathbf{W}|_*$, we have

$$\begin{aligned}
& f(\mathbf{W}^{k+1}, \mathbf{b}^{k+1}, \epsilon^{k+1}, \gamma^{k+1}) \\
= \; & L(\mathbf{W}^{k+1}, \mathbf{W}^{k+1}, \mathbf{b}^{k+1}, \gamma^{k+1}, \epsilon^{k+1}) \\
\geq \; & L(\mathbf{Z}^{k+1}, \mathbf{W}^{k+1}, \mathbf{b}^{k+1}, \epsilon^{k+1}, \gamma^{k+1}) - \lambda|\mathbf{Z}^{k+1} - \mathbf{W}^{k+1}|_* \\
\geq \; & L(\mathbf{Z}^{k+1}, \mathbf{W}^{k+1}, \mathbf{b}^{k+1}, \epsilon^{k+1}, \gamma^{k+1}) - \frac{\lambda}{\mu}|\mathbf{Y}^{k+1} - \mathbf{Y}^k|_* \\
\geq \; & L(\mathbf{Z}^{k+1}, \mathbf{W}^{k+1}, \mathbf{b}^{k+1}, \epsilon^{k+1}, \gamma^{k+1}) - O(\frac{\lambda}{\mu}) \\
\geq \; & f^* - O(\frac{\lambda}{\mu})
\end{aligned} \tag{A.33}$$

Combining Eqn. A.31 and Eqn. A.33, we have

$$\lim_{t\to\infty} |f(\mathbf{W}^{k+1}, \mathbf{b}^{k+1}, \epsilon^{k+1}) - f^*| = 0 \tag{A.34}$$

This proves the convergence. □

APPENDIX B

ALGORITHM ANALYSIS FOR CLUSTERED MULTI-ATTRIBUTE PREDICTION

For the computation complexity, in each iteration the updating of the $\ell_{2,1}$ norm takes $O(d)$. The main computation to update M is the matrix multiplication and the eigen decomposition which both take $O(dm^2)$. The computation to update W mainly includes matrix multiplication and matrix inverse whose total time complexity is $O(nd^2)$. In most applications, the number of class is less than the number of training samples and the number of feature dimensions $m^2 \ll nd$, thus the total time complexity for the algorithm in each iteration is $O(nd^2)$.

Since the optimization problem in our work is a relaxed convex smooth problem, ASO is guarantee to converge to a global optimal solution. We empirically understand the convergence of the proposed approach. We report the average classification accuracies by **SVM** and k**NN** with respect to the number of iterations on 6 public benchmark datasets, as shown in Figure B.1. The results from both **SVM** and k**NN** show that on all datasets, the average classification accuracy increases monotonically. Meanwhile, the accuracies increase dramatically at the beginning and then slowly after 15 iterations. This experiment empirically verifies the convergence property of FSMC and implies a fairly quick convergence (around 20 iterations).

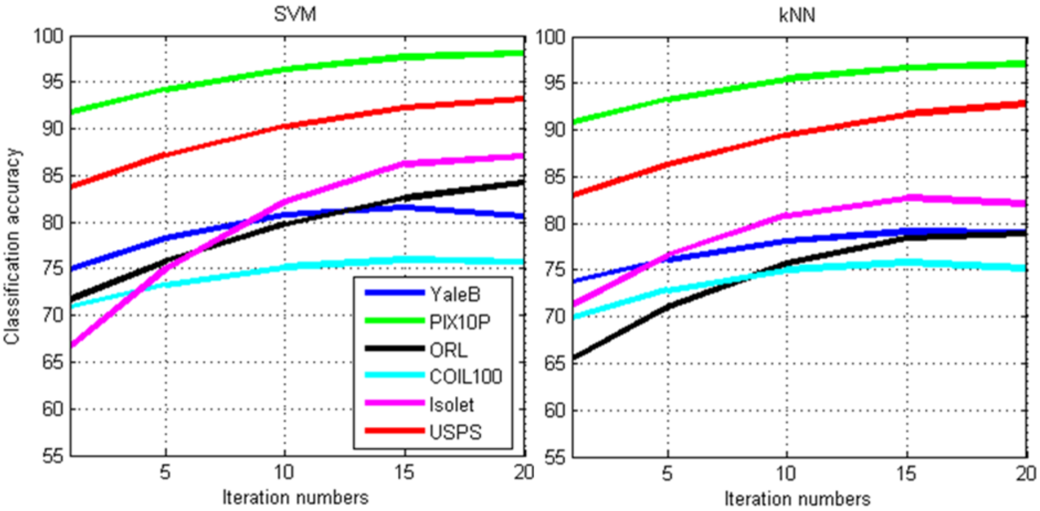


Figure B.1: Classification Accuracy w.r.t. the Number of Iterations

APPENDIX C

ANALYSIS OF THE ONLINE LEARNING ALGORITHM FOR LABEL FUSION LEARNING

We adopt similar techniques to what used in [81] for the convergence analysis of the above algorithm. To bound the average instantaneous objective function, we first introduce the following Lemma:

Lemma 6. *(Lemma 1 in [81]) Let $f_1, \cdots, f_T$ be a sequence of λ-strongly convex functions. Let $\mathcal{B}$ be a closed convex set and define $\Pi_{\mathcal{B}}(\boldsymbol{w}) = argmin_{\boldsymbol{w}'\in\mathcal{B}}\|\boldsymbol{w} - \boldsymbol{w}'\|$. Let $\boldsymbol{w}_1, \cdots, \boldsymbol{w}_{T+1}$ be a sequence of vectors such that $\boldsymbol{w}_1 \in \mathcal{B}$ and for $t \geq 1$, $\boldsymbol{w}_{t+1} = \Pi_{\mathcal{B}}(\boldsymbol{w}_t - \eta_t \nabla_t)$, where ∇_t belongs to the sub-gradient set of f_t at $\boldsymbol{w}_t$ and $\nabla_t = 1/(\lambda t)$. Assume that for all t, $\|\nabla_t\| \leq G$. Then, for all $\boldsymbol{u} \in \mathcal{B}$ we have*

$$\frac{1}{T}\sum_{t=1}^{T} f_t(\boldsymbol{w}_t) \leq \frac{1}{T}\sum_{t=1}^{T} f_t(\boldsymbol{u}) + \frac{G^2(1 + ln(T))}{2\lambda T}$$

Given the above lemma, we can prove the following bounds of the average instantaneous objective function:

Theorem 7. *Assume that for all $(\boldsymbol{x}^{\mathcal{P}}, y^{\mathcal{P}}) \in \mathcal{P}$ the norm of $\boldsymbol{x}^{\mathcal{P}}$ is at most R. Let $\boldsymbol{w}^*$ denote the minimizer of the objective as $\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} f(\boldsymbol{w})$ and let $c = 4R^2$. Then for $T \geq 3$, the objective function of Eq. (5.3) satisfies:*

$$\frac{1}{T}\sum_{t=1}^{T} f(\boldsymbol{w}_t; A_t) \leq \frac{1}{T}\sum_{t=1}^{T} f(\boldsymbol{w}^*; A_t) + \frac{c(1 + ln(T))}{2\lambda T}$$

Proof. If for all $(\boldsymbol{x}^{\mathcal{P}}, y^{\mathcal{P}}) \in \mathcal{P}$ the norm of $\boldsymbol{x}^{\mathcal{P}}$ is at most R, then for all $(\boldsymbol{x}^{\mathcal{O}}, y^{\mathcal{O}}) \in \mathcal{O}$ and $(\boldsymbol{x}^{\mathcal{S}}, y^{\mathcal{S}}) \in \mathcal{S}$ the norm of $\boldsymbol{x}^{\mathcal{O}}$ and $\boldsymbol{x}^{\mathcal{S}}$ is also at most R. Let $\lambda = \frac{1}{c_1 + c_2}$, then Eq. (5.3) and (5.4) are equivalent to

$$\begin{aligned} f(\boldsymbol{w}; A_t) =& \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \lambda c_1 \tau_1 \sum_{i\in\mathcal{A}_t} \ell_1(\boldsymbol{w}; (\boldsymbol{x}_i^{\mathcal{P}}, y_i)) \\ &+ \lambda c_2 \tau_2 \sum_{i\in\mathcal{A}_t} \ell_2(\boldsymbol{w}; \boldsymbol{x}_i^{\mathcal{O}}) + \lambda c_2 \tau_3 \sum_{i\in\mathcal{A}_t} \ell_3(\boldsymbol{w}; \boldsymbol{x}_i^{\mathcal{S}}), \end{aligned}$$

$$\begin{aligned} \nabla_t = \lambda \boldsymbol{w}_t &- \lambda c_1 \tau_1 \sum_{i\in\mathcal{A}_t} \chi_{\Re^+}(1 - y_i^{\mathcal{P}}\langle \boldsymbol{x}_i^{\mathcal{P}}, \boldsymbol{w}_t\rangle) y_i^{\mathcal{P}} \boldsymbol{x}_i^{\mathcal{P}} \\ &- \lambda c_2 \tau_2 \sum_{i\in\mathcal{A}_t} \chi_{\Re^+}(\rho - \langle \boldsymbol{x}_i^{\mathcal{O}}, \boldsymbol{w}_t\rangle) \boldsymbol{x}_i^{\mathcal{O}} \\ &- \lambda c_2 \tau_3 \sum_{i\in\mathcal{A}_t} \mathrm{sgn}\, \langle \boldsymbol{x}_i^{\mathcal{S}}, \boldsymbol{w}_t\rangle \boldsymbol{x}_i^{\mathcal{S}}. \end{aligned} \tag{C.1}$$

Note that Eq. (C.1) is a λ-strongly convex function since it is a sum of a λ-strongly function $\frac{\lambda}{2}\|w\|^2$ and several convex function the average hinge-loss and absolute value functions. Next, we derive a bound on $\|\nabla_t\|$.

Following the sub-gradient descent procedure, the weight vector will be updated by

$$\boldsymbol{w}_{t+1} = (1 - \frac{1}{t})\boldsymbol{w}_t + \frac{1}{\lambda t}\boldsymbol{v}_t \tag{C.2}$$

where

$$\begin{aligned}\boldsymbol{v}_t = \lambda c_1\tau_1 \sum_{i\in\mathcal{A}_t} \chi_{\Re^+}(1-y^{\mathcal{P}}\langle\boldsymbol{x}_i^{\mathcal{P}},\boldsymbol{w}_t\rangle)y_i^{\mathcal{P}}\boldsymbol{x}_i^{\mathcal{P}} \\ + \lambda c_2\tau_2 \sum_{i\in\mathcal{A}_t} \chi_{\Re^+}(\rho-\langle\boldsymbol{x}_i^{\mathcal{O}},\boldsymbol{w}_t\rangle)\boldsymbol{x}_i^{\mathcal{O}} \\ + \lambda c_2\tau_3 \sum_{i\in\mathcal{A}_t} \operatorname{sgn}\langle\boldsymbol{x}_i^{\mathcal{S}},\boldsymbol{w}_t\rangle\boldsymbol{x}_i^{\mathcal{S}}\end{aligned} \tag{C.3}$$

Note that the initial weight of each $\boldsymbol{v}_i$ is $\frac{1}{\lambda i}$ and on round $j=i+1,\cdots,t$ it will be multiplied by $1-\frac{1}{j}=\frac{j-1}{j}$, the overall weight of $\boldsymbol{v}_i$ in $\boldsymbol{w}_{t+1}$ is

$$\frac{1}{\lambda i}\prod_{j=i+1}^{t}\frac{j-1}{j}=\frac{1}{\lambda t},$$

then the update rule can be rewritten as

$$\boldsymbol{w}_{t+1}=\frac{1}{\lambda t}\sum_{i=1}^{t}\boldsymbol{v}_i. \tag{C.4}$$

According to the definition,

$$\lambda c_1\tau_1+\lambda c_2\tau_2+\lambda c_2\tau_3=\frac{\frac{c_1}{c_1+c_2}n_1+\frac{c_2}{c_1+c_2}n_2+\frac{c_2}{c_1+c_2}n_3}{n_1+n_2+n_3}\leq 1$$

thus $\|\boldsymbol{v}_i\|\leq R$ by Eq. (C.3). Therefore, we get $\|\boldsymbol{w}_{t+1}\|\leq R/\lambda$ by Eq. (C.4) and $\|\nabla_t\|\leq 2R$ by Eq. (C.1).

Since here we consider $\mathcal{B}$ as $\mathcal{R}^n$, we have shown that the conditions stated in Lemma 1 hold and Theorem 1 is proved. □

By the convexity of f,

$$f(\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{w}_t)\leq\frac{1}{T}\sum_{t=1}^{T}f(\boldsymbol{w}_t)$$

then we get the following bounds for average hypothesis $\bar{\boldsymbol{w}}$ and final hypothesis $\boldsymbol{w}_{T+1}$ according to the following Lemma from [81]:

Lemma 8. *(Corollary 2 and Lemma 3 in [81]) Assume that the conditions in Thm. 1 hold and that for all t, each element in A_t is sampled uniformly at random from S (with or without repetitions). Assume also that $R\geq 1$ and $\lambda\leq 1/4$. Let $\bar{\boldsymbol{w}}=\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{w}_t$, then with probability of at least $1-\delta$ we have*

$$f(\bar{\boldsymbol{w}})\leq f(\boldsymbol{w}^*)+\frac{21cln(T/\delta)}{\lambda T}$$

if t is selected at random from $[T]$, we have with a probability of at least $\frac{1}{2}$ that

$$f(\boldsymbol{w}_t)\leq f(\boldsymbol{w}^*)+\frac{42cln(T/\delta)}{\lambda T}$$

Lemma 2 shows our proposed framework have the same convergence property and thus we can terminate the procedure at a random stopping time and in at least half of the cases the last hypothesis is an accurate solution.

BIOGRAPHICAL SKETCH

Lin Chen is a PhD candidate from Computer Science and Engineering at Arizona State University. He obtained his Master degree in Computer Science and Bachelor degree in Software Engineering at Shandong University in 2008 and 2011, respectively. His research interests are in applied machine learning and computer vision, specifically semantic attribute learning, multi-task learning, multi-view learning, information retrieval and feature selection. He worked as a research intern at Nokia Technologies in 2015. His research work has been published on various top tier conferences including CVPR, IJCAI, ACMMM, etc. He also served as PC members or reviewers on multiple conference or journals including CVPR, IJCAI, etc.