Collaboration based Multi-Label Learning

Lei Feng1,2, Bo An1, Shuo He3

1School of Computer Science and Engineering, Nanyang Technological University, Singapore
2Alibaba-NTU Singapore Joint Research Institute, Singapore
3College of Computer and Information Science, Southwest University, Chongqing, China
{feng0093, boan}@ntu.edu.sg, hs8207083890@email.swu.edu.cn

Abstract

It is well-known that exploiting label correlations is crucially important to multi-label learning. Most of the existing approaches take label correlations as prior knowledge, which may not correctly characterize the real relationships among labels. Besides, label correlations are normally used to regularize the hypothesis space, while the final predictions are not explicitly correlated. In this paper, we suggest that for each individual label, the final prediction involves the collaboration between its own prediction and the predictions of other labels. Based on this assumption, we first propose a novel method to learn the label correlations via sparse reconstruction in the label space. Then, by seamlessly integrating the learned label correlations into model training, we propose a novel multi-label learning approach that aims to explicitly account for the correlated predictions of labels while training the desired model simultaneously. Extensive experimental results show that our approach outperforms the state-of-the-art counterparts.

Introduction

Multi-label learning deals with the problem where an instance can be associated with multiple labels simultaneously. Formally speaking, let $X \in \mathbb{R}^d$ be $d$-dimensional feature space and $Y = \{y_1, y_2, \ldots, y_q\}$ be the label space with $q$ labels. Given the multi-label training set $D = \{(x_i, y_i)\}_{i=1}^n$, where $x_i \in X$ is a feature vector and $y_i \in \{-1, 1\}^q$ is the label vector, the goal of multi-label learning is to learn a model $f : \mathbb{R}^d \rightarrow \{-1, 1\}^q$, which maps from the space of feature vectors to the space of label vectors. As a learning framework that handles objects with multiple semantics, multi-label learning has been widely applied in many real-world applications, such as image annotation (Yang et al. 2016), document categorization (Li, Ouyang, and Zhou 2015), bioinformatics (Zhang and Zhou 2006), and information retrieval (Gopal and Yang 2010).

The most straightforward multi-label learning approach (Boutell et al. 2004) is to decompose the problem into a set of independent binary classification tasks, one for each label. Although this strategy is easy to implement, it may result in degraded performance, due to the ignorance of correlations among labels. To compensate for this deficiency, the exploitation of label correlations has been widely accepted as a key component of effective multi-label learning approaches (Gibaja and Ventura 2015; Zhang and Zhou 2014).

So far, many methods have been developed to improve the performance of multi-label learning by exploring various types of label correlations (Tsoumakas et al. 2009; Cesa-Bianchi, Gentile, and Zaniboni 2006; Petterson and Caten 2011; Huang, Zhou, and Zhou 2012; Huang, Yu, and Zhou 2012; Zhu, Kwok, and Zhou 2018). There has been increasing interest in exploiting the label correlations by taking the label correlation matrix as prior knowledge (Har-iharan et al. 2010; Cai et al. 2013; Huang et al. 2016; Huang et al. 2018). Concretely, these methods directly calculate the label correlation matrix by the similarity between label vectors using common similarity measures, and then incorporate the label correlation matrix into model training for further enhancing the predictions of multiple label assignments. However, the label correlations are simply obtained by common similarity measures, which may not be able to reflect complex relationships among labels. Besides, these methods exploit label correlations by manipulating the hypothesis space, while the final predictions are not explicitly correlated.

To address the above limitations, we make a key assumption that for each individual label, the final prediction involves the collaboration between its own prediction and the predictions of other labels. Based on this assumption, a novel multi-label learning approach named CAMEL, i.e., CollAboration based Multi-labEl Learning, is proposed. Different from most of the existing approaches that calculate the label correlation matrix simply by common similarity measures, CAMEL presents a novel method to learn such matrix and show that it is equivalent to sparse reconstruction in the label space. The learned label correlation matrix is capable of reflecting the collaborative relationships among labels regarding the final predictions. Subsequently, CAMEL seamlessly incorporates the learned label correlations into the desired multi-label predictive model. Specifically, label-independent embedding is introduced, which aims to fit the final predictions with the learned label correlations while guiding the estimation of the model parameters simultaneously. The effectiveness of CAMEL is clearly demonstrated...
by experimental results on a number of datasets.

## Related Work

In recent years, many algorithms have been proposed to deal with multi-label learning tasks. In terms of the order of label correlations being considered, these approaches can be roughly categorized into three strategies (Zhang and Zhou 2014; Gibaja and Ventura 2015).

For the first-order strategy, the multi-label learning problem is tackled in a label-by-label manner where label correlations are ignored. Intuitively, one can easily decompose the multi-label learning problem into a series of independent binary classification problems (one for each label) (Boutell et al. 2004). The second-order strategy takes into consideration pairwise relationships between labels, such as the ranking between relevant labels and irrelevant labels (Elisseeff and Weston 2002) or the interaction of paired labels (Zhu et al. 2005). For the third-order strategy, high-order relationships among labels are considered. Following this strategy, numerous multi-label algorithms are proposed. For example, by modeling all other labels’ influences on each label, a shared subspace (Ji et al. 2008) is extracted for model training. By addressing connections among random subsets of labels, a chain of binary classifiers (Read et al. 2011) are sequentially trained.

Recently, there has been increasing interest in second-order approaches (Harrihan et al. 2010; Cai et al. 2013; Huang et al. 2016; Huang et al. 2018) that take the label correlation matrix as prior knowledge for model training. These approaches normally directly calculate the label correlation matrix by the similarity between label vectors using common similarity measures, and then incorporate the label correlation matrix into model training for further enhancing the predictions of multiple label assignments. For instance, cosine similarity is widely used to calculate the label correlation matrix (Cai et al. 2013; Huang et al. 2016; Huang et al. 2018). Such label correlation matrix is further incorporated into a structured sparsity-inducing norm regularization (Cai et al. 2013) for regularizing the learning hypotheses, or performing joint label-specific feature selection and model training (Huang et al. 2016; Huang et al. 2018). In addition, there are also some high-order approaches that exploit label correlations on the hypothesis space, while they do not rely on the label correlation matrix. For example, a boosting approach (Huang, Yu, and Zhou 2012) is proposed to exploit label correlations with a hypothesis reuse mechanism.

Note that most of the existing approaches using label correlation matrix are second-order and focus on the hypothesis space. Such simple label correlations exploited in the hypothesis space may not correctly depict the real relationships among labels, and final predictions are not explicitly correlated. In the next section, a novel high-order approach with crafted label correlation matrix that focus on the label space will be introduced.

## The CAMEL Approach

Following the notations used in Introduction, the training set can be alternatively represented by $\mathcal{D} = \{(X, Y)\}$ where $X = [x_1, x_2, \ldots, x_n]^\top \in \mathbb{R}^{n \times d}$ denotes the instance matrix, and $Y = [y_1, y_2, \ldots, y_n]^\top \in \mathbb{R}^{n \times q}$ denotes the label matrix. In addition, we denote by $Y_j \in \mathbb{R}^n$ the $j$-th column vector of the matrix $Y$ (versus $y_j$ $\in \mathbb{R}^q$ for the $j$-th row vector of $Y$), and $Y_{-j} = [y_1, \ldots, y_{j-1}, y_{j+1}, \ldots, y_q]^\top \in \mathbb{R}^{n \times (q-1)}$ represents the matrix that excludes the $j$-th column vector of $Y$.

### Label Correlation Learning

To characterize the collaborative relationships among labels regarding the final predictions, CAMEL works by learning a label correlation matrix $S = [s_{ij}]_{q \times q}$ where $s_{ij}$ reflects the contribution of the $i$-label to the $j$-label. Guided by the assumption that for each individual label, the final prediction involves the collaboration between its own prediction and the predictions of other labels, we thus take the given label matrix as the final prediction, and propose to learn the label correlation matrix $S$ in the following way:

$$
\min_{s_{ij}} \left\| (1 - \alpha)Y_j + \alpha \sum_{i \neq j, i \in [q]} s_{ij} Y_i - Y_j \right\|_2^2 (1)
$$

where $\alpha$ is the tradeoff parameter that controls the collaboration degree. In other words, $\alpha$ is used to balance the $j$-th label’s own prediction and the predictions of other labels. Since each label is normally correlated with only a few labels, the collaborative relationships between one label and other labels could be sparse. With a slight abuse of notation, we denote by $S_j = [s_{1j}, \ldots, s_{j-1, j}, s_{j+1, j}, \ldots, s_{qj}]^\top \in \mathbb{R}^{(q-1)}$ the $j$-th column vector of $S$ excluding $s_{jj}$ ($s_{jj} = 0$).

Under canonical sparse representation, the coefficient vector $S_j$ is learned by solving the following optimization problem:

$$
\min_{s_j} \left\| (1 - \alpha)Y_j + \alpha Y_{-j} S_j - Y_j \right\|_2^2 + \lambda \left\| S_j \right\|_1 (2)
$$

where $\lambda$ controls the sparsity of the coefficient vector $S_j$. By properly rewriting the above problem and setting $\lambda = \lambda / \alpha$, it is easy to derive the following equivalent optimization problem:

$$
\min_{s_j} \left\| Y_{-j} S_j - Y_j \right\|_2^2 + \lambda \left\| S_j \right\|_1 (3)
$$

Here, this problem aims to estimate the collaborative relationships between the $j$-th label and the other labels via sparse reconstruction. The first term corresponds to the linear reconstruction error via $\ell_2$ norm, and the second term controls the sparsity of the reconstruction coefficients by using $\ell_1$ norm. The relative importance of each term is balanced by the tradeoff parameter $\lambda$, which is empirically set to $\frac{1}{100 \sup \left\| Y_j \right\| Y_{-j} \right\|_\infty}$ in the experiments. To solve problem (3), the popular Alternating Direction Method of Multiplier (ADMM) (Boyd et al. 2011) is employed, and detailed information is given in Appendix A. After solving problem (3) for each label, the weight matrix $S$ can be accordingly constructed with all diagonal elements set to 0.
Note that for most of the existing second-order approaches using label correlation matrix (Harihara et al. 2010; Cai et al. 2013; Huang et al. 2016; Huang et al. 2018), only pairwise relationships are considered, and the relationships between one label and the other labels are separated. While for CAMEL, since the final prediction of each label is determined by all the predictions of other labels and itself, the relationships among all labels are exploited in a collaborative manner. Which means, the relationships between one label and the other labels are coordinated (influenced by each other). Therefore, CAMEL is a high-order approach.

Multi-Label Classifier Training

In this section, we propose a novel multi-label learning approach by seamlessly integrating the learned label correlations into the desired predictive model. Suppose the ordinary prediction matrix of \( X \) is denoted by \( f(X) = [f_1(X), f_2(X), \cdots, f_q(X)] \in \mathbb{R}^{n \times q} \) where \( f_1(\cdot), f_2(\cdot), \cdots, f_q(\cdot) \) denotes the individual label predictors respectively. In the ordinary setting, each label predictor is only in charge of a single label, while label correlations are fully lost. To absorb the learned label correlations into predictions, we reuse the assumption that each above formulation, we choose to train the widely-used model where \( f(X) = \phi(X)W + 1b^\top \) where \( W \) and \( b \) are the model parameters, \( 1 = [1, \cdots, 1]^\top \) denotes the column vector with all elements equal to 1, and \( \phi(\cdot) \) is a feature mapping that maps the feature space to some higher (maybe infinite) dimensional Hilbert space. For the regularization term to control the model complexity, we adopt the widely-used squared Frobenius norm, i.e., \( \|W\|_F^2 \). To further facilitate a kernel extension for the general nonlinear case, we finally present the formulation as a constrained optimization problem:

\[
\min_{W,Z,b} \frac{1}{2} \|E\|_F^2 + \frac{\lambda_1}{2} \|ZG - Y\|_F^2 + \frac{\lambda_2}{2} \|W\|_F^2 \quad \text{s.t.} \quad Z - \phi(X)W - 1b^\top = E
\]  

Optimization

Problem (7) is convex with respect to \( W \) and \( b \) with \( Z \) fixed, and also convex with respect to \( Z \) with \( W \) and \( b \) fixed. Therefore, it is a biconvex problem (Gorski, Pfeuffer, and Klamroth 2007), and can be solved by an alternating approach.

Updating \( W \) and \( b \) with \( Z \) fixed

With \( Z \) fixed, problem (7) reduces to

\[
\min_{E,W,b} \frac{1}{2} \|E\|_F^2 + \frac{\lambda_1}{2} \|ZG - Y\|_F^2 + \frac{\lambda_2}{2} \|W\|_F^2 \quad \text{s.t.} \quad Z - \phi(X)W - 1b^\top = E
\]  

By deriving the Lagrangian of the above constrained problem and setting the gradient with respect to \( W \) to 0, it is easy to show \( W = \frac{1}{\lambda_2} \phi(X)^\top A \) where \( A = [a_{ij}]_{n \times q} \) is the matrix that stores the Lagrangian multipliers. Let \( K = \phi(X)\phi(X)^\top \) be the kernel matrix with its element \( k_{ij} = K(x_i, x_j) = \phi(x_i)^\top \phi(x_j) \), where \( K(\cdot, \cdot) \) represents the kernel function. For CAMEL, Gaussian kernel function

\[
\text{Algorithm 1 The CAMEL Algorithm}
\]

**Inputs:**
- \( D \): the multi-label training set \( D = \{(X, Y)\} \)
- \( \alpha, \lambda_1, \lambda_2 \): the hyperparameters
- \( x \): the unseen test instance

**Output:**
- \( y \): the predicted label for the test instance \( x \)

1: learn the label correlation matrix \( S \) by solving problem (3) for each label via ADMM procedure;
2: set \( G = (1 - \alpha)I + \alpha S \);
3: initialize \( Z = Y \);
4: construct the kernel matrix \( K = [K(x_i, x_j)]_{n \times n} \) by Gaussian kernel function;
5: repeat
6: update \( b \) and \( A \) according to (9);
7: update \( T = \frac{1}{\lambda_2} K A + 1b^\top \);
8: update \( Z \) in terms of (11);
9: until convergence.
10: return the predicted label vector \( y \) according to (12).
Evaluation Metrics

For performance evaluation, we use all the employed multi-label evaluation metrics, their values vary within the interval [0,1]. In addition, for the last three metrics, the larger values indicate the better performance, and we use the symbol ↑ to present such positive logic. While for the first five metrics, the smaller values indicate the better performance, which is represented by ↓. More detailed information about these evaluation metrics can be found in (Zhang and Zhou 2014).

Comparing Approaches CAMEL is compared with three well-established and two state-of-the-art multi-label learning algorithms, including the first-order approach BR (Boutell et al. 2004), the second-order approaches LLSF (Huang et al. 2016) and JFSC (Huang et al. 2018), and the high-order approaches ECC (Read et al. 2011), and RAKEL (Tsoumakas, Katakis, and Vlahavas 2011). Here, LLSF and JFSC are the state-of-the-art counterparts using label correlation matrix.

BR, ECC, and RAKEL are implemented under the MULAN multi-label learning package (Tsoumakas et al. 2011) by using the logistic regression model as the base classifier. Furthermore, parameters suggested in the corresponding literatures are used, i.e., ECC: ensemble size 30; RAKEL: ensemble size 2q with k = 3. For LLSF, parameters α, β are chosen from \{2^{-10}, 2^{-9}, \ldots, 2^{10}\}, and ρ chosen from \{0.1, 1, 10\}. For JFSC, parameters α, β, and γ are chosen from \{4^{-5}, 4^{-4}, \ldots, 4^{5}\}, and η is chosen from \{0.1, 1, 10\}. For the proposed approach CAMEL, λ1 is empirically set to 1, λ2 is chosen from \{10^{-3}, 2 \times 10^{-3}, 10^{-2}, 2 \times 10^{-2}, \ldots, 10^{0}\}, and α is chosen from \{0, 0.1, \ldots, 1\}. All of these parameters are decided by conducting 5-fold cross-validation on training set.

Table 1: Characteristics of the benchmark multi-label datasets.

| Dataset   | \(|S|\) | \(dim(S)\) | \(L(S)\) | \(LCard(S)\) | \(F(S)\) |
|-----------|--------|------------|---------|-------------|---------|
| cal500    | 502    | 68         | 174     | 26.04       | numeric |
| emotions  | 593    | 72         | 6       | 1.87        | numeric |
| genbase   | 662    | 1185       | 27      | 1.25        | nominal |
| medical   | 978    | 1449       | 45      | 1.25        | nominal |
| enron     | 1702   | 1001       | 53      | 3.38        | numeric |
| image     | 2000   | 294        | 5       | 1.24        | numeric |
| scene     | 2407   | 294        | 5       | 1.07        | numeric |
| yeast     | 2417   | 103        | 14      | 4.24        | numeric |
| science   | 5000   | 743        | 40      | 1.45        | numeric |
| arts      | 5000   | 462        | 26      | 1.64        | numeric |
| business  | 5000   | 438        | 30      | 1.59        | numeric |
| rcvl-s1   | 6000   | 944        | 101     | 2.88        | nominal |
| rcvl-s2   | 6000   | 944        | 101     | 2.63        | nominal |
| rcvl-s3   | 6000   | 944        | 101     | 2.61        | nominal |
| rcvl-s4   | 6000   | 944        | 101     | 2.48        | nominal |
| rcvl-s5   | 6000   | 944        | 101     | 2.64        | nominal |

Experimental Results

Table 2 and 3 report the detailed experimental results on the regular-scale and large-scale datasets respectively, where the
Compared with the three well-established algorithms, CAMEL ranks first in 69.6% (39/56) cases, and on the large-scale datasets (Table 3), across all the evaluation metrics, CAMEL ranks first in 80.4% (45/56) cases. Specifically, on the regular-size datasets (Table 2), across all the evaluation metrics, CAMEL outperforms other comparing algorithms in most cases. From the two result tables, we can see that CAMEL shows the best performance among all the algorithms is shown in boldface.

## Table 2: Predictive performance of each algorithm (mean±std. deviation) on the regular-scale datasets.

| Comparing algoritisms | One-error↓ | Hamming loss↑ | Tracking loss↓ | Rankin loss↑ | Average precision↑ | Micro-averaging F1↑ |
|-----------------------|------------|---------------|----------------|-------------|-------------------|-------------------|
| CAMEL                 | 0.129±0.053 | 0.292±0.052   | 0.001±0.001    | 0.110±0.021 | 0.207±0.038       | 0.242±0.033       |
| BR                    | 0.893±0.038 | 0.284±0.077   | 0.017±0.016    | 0.322±0.055 | 0.646±0.023       | 0.387±0.027       |
| ECC                   | 0.295±0.036 | 0.296±0.074   | 0.010±0.013    | 0.156±0.037 | 0.421±0.034       | 0.406±0.023       |
| RAKEL                 | 0.634±0.039 | 0.300±0.070   | 0.009±0.007    | 0.243±0.055 | 0.532±0.007       | 0.402±0.024       |
| LLSF                  | 0.138±0.050 | 0.412±0.051   | 0.002±0.005    | 0.120±0.020 | 0.250±0.042       | 0.327±0.030       |
| JFSC                  | 0.116±0.051 | 0.438±0.086   | 0.002±0.005    | 0.128±0.024 | 0.278±0.041       | 0.346±0.023       |

In cases. Compared with the two state-of-the-art algorithms LLSF and JFSC, instead of employing simple similarity measures to regularize the hypothesis space, CAMEL introduces a novel method to learn label correlations for explicitly correlating the final predictions, and achieves superior performance in 80.4% (180/224) cases. These comparative results clearly demonstrate the effectiveness of the collaboration based multi-label learning approach.

### Sensitivity Analysis
In this section, we first investigate the sensitivity of CAMEL with respect to the two tradeoff parameters $\lambda_1$ and $\lambda_2$, and the parameter $\alpha$ that controls the best performance among all the algorithms is shown in boldface. From the two result tables, we can see that CAMEL outperforms other comparing algorithms in most cases. Specifically, on the regular-size datasets (Table 2), across all the evaluation metrics, CAMEL ranks first in 80.4% (45/56) cases, and on the large-scale datasets (Table 3), across all the evaluation metrics, CAMEL ranks first in 69.6% (39/56) cases. Compared with the three well-established algorithms BR, ECC, and RAKEL, CAMEL introduces a new type of label correlations, i.e., collaborative relationships among labels, and achieves superior performance in 93.8% (315/336) cases.
degree of collaboration, then illustrate the convergence of CAMEL. Due to page limit, we only report the experimental results on the enron dataset using the Coverage $\downarrow$ metric. Concretely, we study the performance of CAMEL when we vary one parameter while keeping other parameters fixed at their best setting. Figure 1(a), 1(b), and 1(c) show the sensitivity curve of CAMEL with respect to $\alpha$, $\lambda_1$, and $\lambda_2$ respectively. It can be seen that $\alpha$ and $\lambda_2$ have an important influence on the final performance, because $\alpha$ and $\lambda_2$ control the collaboration degree and the model complexity. Figure 1(d) illustrates the convergence of CAMEL by using the difference of the optimization variable $Z$ between two successive iterations, i.e., $\Delta Z = ||Z^{(t)} - Z^{(t-1)}||_F$. From Figure 1(d), we can observe that $\Delta Z$ quickly decreases to 0 within a few number of iterations. Hence the convergence of CAMEL is demonstrated.

**Conclusion**

In this paper, we make a key assumption for multi-label learning that for each individual label, the final prediction involves the collaboration between its own prediction and the predictions of other labels. Guided by this assumption, we
propose a novel method to learn the high-order label correlations via sparse reconstruction in the label space. Besides, by seamlessly integrating the learned label correlations into model training, we propose a novel multi-label learning approach that aims to explicitly account for the correlated predictions of labels while training the desired model simultaneously. Extensive experimental results show that our approach outperforms the state-of-the-art counterparts.

Despite the demonstrated effectiveness of CAMEL, it only considers the global collaborative relationships between labels, by assuming that such collaborative relationships are shared by all the instances. However, as different instances have different characteristics, such collaborative relationships may be shared by only a subset of instances rather than all the instances. Therefore, our further work is to explore different collaborative relationships between labels for different subsets of instances.

Acknowledgements

This work was supported by MOE, NRF, and NTU.

Appendix A. The ADMM Procedure

To solve problem (3) by ADMM, we first reformulate problem (3) into the following equivalent form:

$$
\min_{S_j, z} \frac{1}{2} \|Y_{-j} S_j - Y_j\|^2_2 + \lambda \|z\|_1
$$

subject to \(S_j - z = 0\)

Following the ADMM procedure, the above constrained optimization problem (13) can be solved as a series of unconstrained minimization problems using augmented Lagrangian function, which is presented as:

$$
\mathcal{L}(S_j, z, \mu) = \frac{1}{2} \|Y_{-j} S_j - Y_j\|^2_2 + \lambda \|z\|_1 + \nu^\top (S_j - z) + \frac{\rho}{2} \|S_j - z\|^2_2
$$

(14)

Here, \(\rho\) is the penalty parameter and \(\nu\) is the Lagrange multiplier. By introducing the scaled dual variable \(\mu = \frac{1}{\rho} \nu\), a sequential minimization of the scaled ADMM iterations can be conducted by updating the three variables \(S_j, z\) and \(\mu\) sequentially:

$$
S_j^{(k+1)} = (Y_{-j}^\top Y_{-j} + \rho I)^{-1}(Y_{-j}^\top Y_j + \rho (z^{(k)} - \mu^{(k)}))
$$

$$
z^{(k+1)} = S_j^{(k+1)} + \mu^{(k)}
$$

$$
\mu^{(k+1)} = \mu^{(k)} + S_j^{(k+1)} - z^{(k+1)}
$$

(15)

where \(S\) is the proximity operator of the \(\ell_1\) norm, which is defined as \(S_w(a) = (a - \omega)_+ - (-a - \omega)_+\).

Appendix B. Model Parameter Optimization

The Lagrangian of problem (8) is expressed as:

$$
\mathcal{L}(W, E, A, b) = \text{tr}(E^\top E) + \lambda_2 \text{tr}(W^\top W) + \lambda_1 \text{tr}(L_1 b^\top b - E)
$$

(16)

where \(\text{tr}\) is the trace operator, and \(A = [a_1, a_2, \cdots, a_n]^\top \in \mathbb{R}^{n \times q}\) is the introduced matrix that stores the Lagrangian multipliers. Besides, we have used the property of trace operator that \(\text{tr}(W^\top W) = \|W\|_F^2\). By Setting the gradient w.r.t. \(E, A, W, b\) to 0 respectively, the following equations will be induced:

$$
\frac{\partial \mathcal{L}}{\partial E} = 0 \Rightarrow A = E
$$

$$
\frac{\partial \mathcal{L}}{\partial A} = 0 \Rightarrow Z - \phi(X)W - 1b^\top = E
$$

$$
\frac{\partial \mathcal{L}}{\partial W} = 0 \Rightarrow W = \frac{1}{\lambda_2} \phi(X)^\top A
$$

$$
\frac{\partial \mathcal{L}}{\partial b} = 0 \Rightarrow A^\top 1 = 0
$$

(17)

The above linear equations can be simplified by the following steps:

$$
Z = \phi(X)W + 1b^\top + E
$$

$$
Z = \frac{1}{\lambda_2} \phi(X)\phi(X)^\top A + 1b^\top + A
$$

$$
Z = \frac{1}{\lambda_2} KA + 1b^\top + A
$$

(18)
Here, we define $H = \frac{1}{\lambda_2}K + I$, then we can obtain:

$$
\begin{align*}
HA + 1b^\top &= Z \\
A + H^{-1}1b^\top &= H^{-1}Z \\
1^\top H^{-1}1b &= 1^\top H^{-1}Z \\
b^\top &= \frac{1}{H^{-1}I}(Z - 1b^\top)
\end{align*}
$$

In this way, $A$ can be calculated as $A = H^{-1}(Z - 1b^\top)$.

References

[Boutell et al. 2004] Boutell, M. R.; Luo, J.; Shen, X.; and Brown, C. M. 2004. Learning multi-label scene classification. Pattern Recognition 37(9):1757–1771.

[Boyd et al. 2011] Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J.; et al. 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine learning 3(1):1–122.

[Cai et al. 2013] Cai, X.; Nie, F.; Cai, W.; and Huang, H. 2013. New graph structured sparsity model for multi-label image annotations. In Proceedings of the IEEE International Conference on Computer Vision, 801–808.

[Cesa-Bianchi, Gentile, and Zaniboni 2006] Cesa-Bianchi, N.; Gentile, C.; and Zaniboni, L. 2006. Hierarchical classification: combining bayes with svm. In Proceedings of the 23rd International Conference on Machine learning, 177–184.

[Elisseeff and Weston 2002] Elisseeff, A., and Weston, J. 2002. A kernel method for multi-labelled classification. In Advances in Neural Information Processing Systems, 681–687.

[Gibaja and Ventura 2015] Gibaja, E., and Ventura, S. 2015. A tutorial on multilabel learning. ACM Computing Surveys 47(3):52.

[Gopal and Yang 2010] Gopal, S., and Yang, Y. 2010. Multilabel classification with meta-level features. In Proceedings of the 33rd international ACM SIGIR Conference on Research and Development in Information Retrieval, 315–322.

[Gorski, Pfeuffer, and Klamroth 2007] Gorski, J.; Pfeuffer, F.; and Klamroth, K. 2007. Biconvex sets and optimization with biconvex functions: A survey and extensions. Mathematical Methods of Operations Research 66(3):373–407.

[Hariharan et al. 2010] Hariharan, B.; Zelnik-Manor, L.; Varma, M.; and Vishwanathan, S. 2010. Large scale max-margin multi-label classification with priors. In Proceedings of the 27th International Conference on Machine Learning, 423–430.

[Huang et al. 2016] Huang, J.; Li, G.; Huang, Q.; and Wu, X. 2016. Learning label-specific features and class-dependent labels for multi-label classification. IEEE Transactions on Knowledge and Data Engineering 28(12):3309–3323.

[Huang et al. 2018] Huang, J.; Li, G.; Huang, Q.; and Wu, X. 2018. Joint feature selection and classification for multilabel learning. IEEE Transactions on Cybernetics 48(3):876–889.

[Huang, Yu, and Zhou 2012] Huang, S.-J.; Yu, Y.; and Zhou, Z.-H. 2012. Multi-label hypothesis reuse. In Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 525–533.

[Huang, Zhou, and Zhou 2012] Huang, S.-J.; Zhou, Z.-H.; and Zhou, Z. 2012. Multi-label learning by exploiting label correlations locally. In Proceedings of the 26th AAAI Conference on Artificial Intelligence, 949–955.

[Li, Ouyang, and Zhou 2015] Li, X.; Ouyang, J.; and Zhou, X. 2015. Supervised topic models for multi-label classification. Neurocomputing 149:811–819.

[Petterson and Caetano 2011] Petterson, J., and Caetano, T. S. 2011. Submodular multi-label learning. In Advances in Neural Information Processing Systems, 1512–1520.

[Read et al. 2011] Read, J.; Pfahringer, B.; Holmes, G.; and Frank, E. 2011. Classifier chains for multi-label classification. Machine Learning 85(3):333.

[Tsoumakas et al. 2009] Tsoumakas, G.; Dimou, A.; Spyromitros, E.; Mezaris, V.; Kompatsiaris, I.; and Vlahavas, I. 2009. Correlation-based pruning of stacked binary relevance models for multi-label learning. In Proceedings of the 1st International Workshop on Learning from Multi-Label Data, 101–116.

[Tsoumakas et al. 2011] Tsoumakas, G.; Spyromitros-Xioufis, E.; Vilcek, J.; and Vlahavas, I. 2011. Mulan: A java library for multi-label learning. Journal of Machine Learning Research 12(7):2411–2414.

[Tsoumakas, Kataki, and Vlahavas 2011] Tsoumakas, G.; Kataki, I.; and Vlahavas, I. 2011. Random k-labelsets for multilabel classification. IEEE Transactions on Knowledge and Data Engineering 23(7):1079–1089.

[Yang et al. 2016] Yang, H.; Tianyi Zhou, J.; Zhang, Y.; Gao, B.-B.; Wu, J.; and Cai, J. 2016. Exploit bounding box annotations for multi-label object recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 280–288.

[Zhao and Zhou 2006] Zhang, M.-L., and Zhou, Z.-H. 2006. Multi-label neural networks with applications to functional genomics and text categorization. IEEE Transactions on Knowledge and Data Engineering 18(10):1338–1351.

[Zhong and Zhou 2014] Zhang, M.-L., and Zhou, Z.-H. 2014. A review on multi-label learning algorithms. IEEE Transactions on Knowledge and Data Engineering 26(8):1819–1837.

[Zhu et al. 2005] Zhu, S.; Ji, X.; Xu, W.; and Gong, Y. 2005. Multi-labelled classification using maximum entropy method. In Proceedings of the 28th International ACM SIGIR Conference on Research and Development in Information Retrieval, 274–281.
cal label correlation. *IEEE Transactions on Knowledge and Data Engineering* 30(6):1081–1094.