Journal on Artificial Intelligence

Tech Science Press

DOI: 10.32604/jai.2024.049083

ARTICLE

Causality-Driven Common and Label-Specific Features Learning

Yuting Xu[1,*], **Deqing Zhang**[1], **Huaibei Guo**[2] **and Mengyue Wang**[1]

[1]School of Intelligent Transportation Modern Industry, Anhui Sanlian University, Hefei, 230601, China

[2]Heyetang Middle School, Jinhua, 322010, China

*Corresponding Author: Yuting Xu. Email: xytingerny@163.com

Received: 27 December 2023 Accepted: 04 March 2024 Published: 05 April 2024

ABSTRACT

In multi-label learning, the label-specific features learning framework can effectively solve the dimensional catastrophe problem brought by high-dimensional data. The classification performance and robustness of the model are effectively improved. Most existing label-specific features learning utilizes the cosine similarity method to measure label correlation. It is well known that the correlation between labels is asymmetric. However, existing label-specific features learning only considers the private features of labels in classification and does not take into account the common features of labels. Based on this, this paper proposes a Causality-driven Common and Label-specific Features Learning, named CCSF algorithm. Firstly, the causal learning algorithm GSBN is used to calculate the asymmetric correlation between labels. Then, in the optimization, both $l_{2,1}$-norm and l_1-norm are used to select the corresponding features, respectively. Finally, it is compared with six state-of-the-art algorithms on nine datasets. The experimental results prove the effectiveness of the algorithm in this paper.

KEYWORDS

Label-specific features learning; causal learning; asymmetric label correlation; common features

1 Introduction

Multi-label learning [1] (MLL) is one of the hot research areas in machine learning, which alleviates the problem that instances covering multiple concepts or semantics in numerous real-world application scenarios cannot be accurately handled by traditional single-label algorithms. In real life, MLL has also long been applied in several domains, such as text classification [2], image annotation [3], protein function detection [4] and personalized recommendation [5], to name a few. With the rapid development of the Internet, data is gradually characterized by high dimensional distribution [6]. This can lead to the problem of dimensional catastrophe suffered by multi-label algorithms for data learning.

Label-specific feature (LSF) learning can effectively solve this problem, which is to establish the label-specific relation between labels and features by learning the connection between features and labels. The core idea is that each label should have a specific feature corresponding to it, i.e., the specific features of the label are learned. In multi-label learning, l_1-norm can attain feature sparsity and extract label-specific features, which we call private features of labels. The $l_{2,1}$-norm can also achieve feature sparsity and extract more relevant features of the labels, which we call the common features of the labels.

Label correlation [7] (LC) has long been commonly used in LSF learning, which effectively improves the classification performance of LSF learning algorithms. However, the correlation calculated by cosine similarity is symmetric, and ignore the asymmetric correlation may introduce redundant information in the model. Cosine similarity is also highly susceptible to dimensional catastrophe. As the amount of data increases, the Euclidean distance metric deteriorates. In the process of calculation, the label relevance calculated by cosine similarity is highly susceptible to the a priori knowledge of the labels. Most of the labels in multi-label datasets rely on manual expert marking. With the increase of data volume and the influence of experts' experience, it is inevitable that there will be omission and miss labeling in the process of marking. For such incomplete datasets, the LC computed by cosine similarity methods are inevitably mixed with many spurious correlations. Therefore, it is necessary to adopt the causal learning [8] algorithm to measure the asymmetric correlation between labels.

In LSF learning, most algorithms only consider the private features of labels and do not consider the common features of labels [9]. However, when we classify two similar labels, the LC of the similar labels are also similarly strongly correlated, but the computed weight matrices are not necessarily similar. As shown in Fig. 1. The labels y_1 and y_2 are strongly correlated labels, yet the learned weight coefficients are really different. This indicates that we should fully consider the common and private features of labels in the process of classification. Only in this way, the LSF learning can obtain more accurate classification performance.

$$\begin{array}{c|cccccc|} & d_1 & d_2 & d_3 & d_4 & d_5 & d_6 \\ \hline n_1 & 0 & 0 & 1 & 0 & 1 & 0 \\ n_2 & 0 & 1 & 0 & 0 & 0 & 1 \\ n_3 & 0 & 0 & 1 & 0 & 1 & 1 \\ n_4 & 0 & 1 & 1 & 0 & 0 & 1 \\ \hline \end{array} \times \begin{array}{|cc|} w_1 & w_2 \\ \hline 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 0 \\ \hline \end{array} = \begin{array}{|cc|} y_1 & y_2 \\ \hline 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ \hline \end{array}$$

Figure 1: The process of addressing the label-specific feature

Based on the above analysis, we propose a causality-driven common and LSF learning. The main contributions of this paper are as follows:

1) We propose a novel CCSF method, which use $l_{2,1}$-norm and l_1-norm to learn the common and private features of labels, respectively. Thereby, more correlated features are extracted for classification.

2) We use a causal learning algorithm to compute asymmetric label correlations, discarding the traditional way of combining correlation matrix and neighbor matrix, which reduces the influence of original labels.

The remaining sections are organized as follows. Section 2 summarizes some state-of-the-art domestic and international research. The proposed framework and model optimization of CCSF are presented in Section 3. Section 4 analyzes the experimental results and other related experiments. Finally, the conclusion is presented in Section 5.

2 Related Work

Traditional MLL considers that all labels are distinguished based on the same features. However, this categorization is not reasonable and brings a lot of redundant information in the process of categorization, and the classification results are often sub-optimal. Zhang et al. proposed the LSF learning algorithm LIFT [10], which considers that each label is classified based on specific features. Compared with the traditional classification methods, it effectively improves the classification performance of MLL algorithm. But the algorithm does not take into account the correlation between labels. We consider that each label does not exist independently, but has a strong or weak correlation with other labels. The LLSF [11] algorithm proposed by Huang et al. uses the cosine similarity method to measure the correlation between labels. Two strongly correlated labels, whose LSF are also strongly correlated, which further improves the performance of the LSF learning algorithm. By different methods to measure the correlation between labels, Cheng et al. proposed the FF-MLLA [12] algorithm, which utilizes the Minkowski distance to measure the inter-sample similarity based on LC, and uses the singular value decomposition and the limit learning machine to classify multiple labels. The LF-LPLC [13] algorithm proposed by Weng et al. uses the nearest-neighbor technique to consider the local correlation of labels on the basis of the LSF learning algorithm. The algorithm not only enriches the semantic information of labels, but also solves the imbalance problem of labels. The MLFC [14] algorithm proposed by Zhang et al. further improves the performance of the LSF learning algorithm by uniting LSF learning and LC to obtain LSF for each label. For the missing label problem occurring in LSF learning algorithms, the LSML [15] algorithm proposed by Huang et al. utilizes the correlation between labels and has better experimental results not only on the complete dataset, but also on the missing label dataset. Zhao et al. proposed the LSGL [16] algorithm, which considers not only global but also local correlations between labels. LSGL algorithm, based on the assumption that both global and local correlations coexist, has more accurate classification performance than the LSF learning algorithm, which only considers local correlations.

However, most of the above algorithms use cosine similarity to measure out symmetric correlations in the learning of LSF. In fact, the correlation between labels is mostly asymmetric. As the data dimension increases, the Euclidean distance metric becomes less effective. ACML [17] algorithm proposed by Bao et al. and CCSRMC [18] algorithm proposed by Zhang et al. measure the asymmetric correlation between labels using the DC algorithm in causal learning, which are both effective in improving the classification performance of MLL. Luo et al. proposed the MLDL [19] algorithm to fully utilize the structural relationship between features and labels. Not only does it use bi-Laplace regularization to mine the local information of the labels, but it also employs a causal learning algorithm to explore the intrinsic causal relationships between the labels. The BDLS [20] algorithm proposed by Tan et al. introduces a bi-mapping learning framework in LSF learning and uses a causal learning algorithm to calculate the asymmetric correlation between labels, which also effectively improves the classification performance of the LSF learning algorithm. However, the above LSF learning only considers the private features of labels and not the common features of labels. CLML [9] algorithm proposed by Li et al. first uses a norm in the LSF framework to extract the common features of the labels. Subsequently, the GLFS [21] algorithm proposed by Zhang et al. builds a group-preserving optimization framework for feature selection by learning the common features of similar labels and the private features of each label using K-means clustering. Based on the above analysis, we adopt a causal learning algorithm to learn asymmetric LC among labels in LSF learning framework. The $l_{2,1}$-norm and l_1-norm used to extract the common and private features of labels, respectively. The effectiveness of the algorithm in this paper is proved through a large number of experiments.

3 CCSF Model Construction and Optimization

3.1 CCSF Model Construction

In MLL, $\boldsymbol{X}$ denotes the feature matrix, Y represents the label matrix, and the dataset $\boldsymbol{D} = \left\{\left(\boldsymbol{x}_1, \boldsymbol{y}_1\right), \left(\boldsymbol{x}_2, \boldsymbol{y}_2\right), \ldots, \left(\boldsymbol{x}_n, \boldsymbol{y}_n\right)\right\}$, where $\boldsymbol{X} \in \mathbb{R}^{n \times d}$, $\boldsymbol{Y} \in \mathbb{R}^{n \times \ell}$, ℓ is the number of labels, n is the number of samples, d is the number of features. $\boldsymbol{x}_n = \{x_{n1}, x_{n2}, \ldots, x_{nd}\}$ and $\boldsymbol{y}_n = \{y_{n1}, y_{n2}, \ldots, y_{n\ell}\}$ denote the feature and label vectors. The basic model of CCSF in conjunction with the LLSF [10] algorithm proposed by Huang et al. can be written as:

$$\min_{W} \frac{1}{2} \|\boldsymbol{XW} - \boldsymbol{Y}\|_F^2 + \alpha \|\boldsymbol{W}\|_1 \tag{1}$$

where α is the feature sparse parameter, $\boldsymbol{W}$ is the weight coefficient and $\boldsymbol{W} = [\boldsymbol{w}_1, \mathbf{w}_2, \mathbf{w}_3, \ldots, \mathbf{w}_\ell] \in \mathbb{R}^{d \times \ell}$, and $\mathbf{w}_\ell \in \mathbb{R}^d$ denotes the LSF of each label. However, Eq. (1) only adopts the l_1-norm, which can only extract the private features of the label, but not the shared features of the label. So, we put $l_{2,1}$-norm in Eq. (1) to extract the common features of labels, and Eq. (2) can be written as:

$$\min_{W} \frac{1}{2} \|\boldsymbol{XW} - \boldsymbol{Y}\|_F^2 + \alpha \|\boldsymbol{W}\|_1 + \beta \|\boldsymbol{W}\|_{2,1} \tag{2}$$

where β is the feature sparse parameter.

LC has been widely used in LSF learning algorithms, which can effectively improve the classification performance of MLL algorithms. But cosine similarity [22] all calculates symmetric correlations. Indeed, correlations between labels are asymmetric [23]. In this paper, we use a globally structured causal learning algorithm GSBN [24]. First, Markov Blanket (MB) or Parent and Child (PC) part-to-whole structure learning for each label is obtained. Then a directed acyclic graph (DAG) framework is constructed using MB or PC learning.

With the constraint of causal LC, assuming that $\boldsymbol{C}$ is the causal LC matrix and C_{ij} denotes the causal relationship between labels y_i and y_j. We improve the learning efficiency of LSF by calculating the Euclidean distance between $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$, $C_{ij} \left\|\boldsymbol{w}_i - \boldsymbol{w}_j\right\|_2^2$. When the labels are causally related, the features are similar. Accordingly, $\boldsymbol{w}_i$ will be closer to $\boldsymbol{w}_j$. The causal correlation matrix $\boldsymbol{C}$ is defined as follows:

$$C_{ij} = \begin{cases} 1 & y_i \rightarrow y_j \\ 0 & y_i \nrightarrow y_j \end{cases} (i,j \in 1, \ldots, \ell) \tag{3}$$

where $y_i \rightarrow y_j$ indicates that the label y_i is causally related to y_j and $C_{ij} = 1$. Conversely $y_i \nrightarrow y_j$ indicates that the label y_i is not causally related to y_j and $C_{ij} = 0$.

Therefore, we add causal constraints based on Eq. (2). The core formula of the CCSF algorithm can be written as:

$$\min_{W} \frac{1}{2} \|\boldsymbol{XW} - \boldsymbol{Y}\|_F^2 + \alpha \|\boldsymbol{W}\|_1 + \beta \|\boldsymbol{W}\|_{2,1} + \gamma \operatorname{tr}\left(\boldsymbol{WCW}^{\mathrm{T}}\right) \tag{4}$$

where γ is the hyperparameter.

3.2 CCSF Model Optimization

Considering the non-smoothness of the $l_{2,1}$-norm, we use the technique in the literature [25] to deal with the non-smoothness.

$$\frac{\partial \left\|\boldsymbol{W}_i\right\|_{2,1}}{\partial \boldsymbol{W}_i} = \frac{\partial Tr\left(\boldsymbol{W}_i^T \boldsymbol{A}_i \boldsymbol{W}_i\right)}{\partial \boldsymbol{W}_i} = 2\boldsymbol{A}_i \boldsymbol{W}_i \tag{5}$$

where $\boldsymbol{A}_i \in \mathbb{R}^{\ell\ell}$ is a diagonal matrix with the jth diagonal element $A_i^{jj} = \frac{\mathbf{1}}{\mathbf{2}\left\|w_i^j\right\|_{\mathbf{2}}}$. If $w_i^j = 0$, then $A_i^{jj} \in \partial$.

The CCSF model is a convex optimization problem. Due to the non-smoothness of the l_1-norm, this paper adopts the accelerated proximal gradient descent method [26] to solve the non-smoothness of the weight matrix W by alternating iterations. The objective function is:

$$\min_{W \in \mathcal{H}} F(\boldsymbol{W}) = f(\boldsymbol{W}) + g(\boldsymbol{W}) \tag{6}$$

where $\mathcal{H}$ is the Hilbert space. The expressions for $f(\boldsymbol{W})$ and $g(\boldsymbol{W})$ are shown in Eqs. (7) and (8), which are both convex functions and satisfy the Lipschitz condition.

$$(\boldsymbol{W}) = \min_{\boldsymbol{W}} \frac{1}{2} \left\|\boldsymbol{X}\boldsymbol{W} - \boldsymbol{Y}\right\|_F^2 + \gamma \operatorname{tr}\left(\boldsymbol{W}\boldsymbol{C}\boldsymbol{W}^{\mathrm{T}}\right) + \beta \left\|\boldsymbol{W}\right\|_{2,1} \tag{7}$$

$$g(\boldsymbol{W}) = \alpha \left\|\boldsymbol{W}\right\|_1 \tag{8}$$

$$\nabla f(\boldsymbol{W}) = \boldsymbol{X}^T\boldsymbol{X}\boldsymbol{W} - \boldsymbol{X}^T\boldsymbol{Y} + 2\gamma \boldsymbol{W}\boldsymbol{C} + \mathbf{2}\boldsymbol{A}\boldsymbol{W} \tag{9}$$

For any matrices $\boldsymbol{W}_1$, $\boldsymbol{W}_2$, there is:

$$\left\|\nabla f(\boldsymbol{W}_1) - \nabla f(\boldsymbol{W}_2)\right\| \leq L_g \left\|\Delta \boldsymbol{W}\right\| \tag{10}$$

where L_g is the Lipschitz constant and $\Delta \boldsymbol{W} = \boldsymbol{W}_1 - \boldsymbol{W}_2$. Introducing the quadratic approximation $F(\boldsymbol{W})$ for $\boldsymbol{Q}\left(\boldsymbol{W}, \boldsymbol{W}^{(t)}\right)$, then

$$Q\left(\boldsymbol{W}, \boldsymbol{W}^{(t)}\right) = f\left(\boldsymbol{W}^{(t)}\right) + \left(\nabla f\left(\boldsymbol{W}^{(t)}\right), \boldsymbol{W} - \boldsymbol{W}^{(t)}\right) + \frac{L_g}{2}\left\|\boldsymbol{W} - \boldsymbol{W}^{(t)}\right\|_F^2 + g(\boldsymbol{W}) \tag{11}$$

Let $\boldsymbol{q}_t(\boldsymbol{W}) = \boldsymbol{W}_t - \frac{1}{L_g}\nabla f(\boldsymbol{W})$, then

$$\boldsymbol{W} = \arg\min_{W} \boldsymbol{Q}\left(\boldsymbol{W}, \boldsymbol{W}^{(t)}\right) = \arg\min_{W} \frac{1}{2}\left\|\boldsymbol{W} - \boldsymbol{q}^{(t)}\right\|_F^2 + \frac{\alpha}{L_g}\left\|\boldsymbol{W}\right\|_1 \tag{12}$$

The optimization algorithm proposed by Lin et al. [27] points out that

$$\boldsymbol{W}^{(t)} = \boldsymbol{W}_t + \frac{\theta_{t+1} - 1}{\theta_t}\left(\boldsymbol{W}_t - \boldsymbol{W}_{t-1}\right) \tag{13}$$

In Eq. (13), b_t satisfies $b_{t+1}^2 - b_{t+1} \leq b_t^2$. Meanwhile, the convergence rate of $\boldsymbol{O}\left(t^{-2}\right)$ is improved, and $\boldsymbol{W}_t$ is the result of the tth iteration. The soft threshold function for performing the iterative operation is shown in Eq. (14).

$$\boldsymbol{W}_{t+1} = \boldsymbol{S}_\varepsilon\left[\boldsymbol{q}^{(t)}\right] = \arg\min_{W} \varepsilon\left\|\boldsymbol{W}\right\|_1 + \frac{1}{2}\left\|\boldsymbol{W} - \boldsymbol{q}^{(t)}\right\|_F^2 \tag{14}$$

where $\boldsymbol{S}_\varepsilon[\cdot]$ is the soft threshold operator. For any one parameter x_{ij} and $\varepsilon = \frac{\alpha}{L_g}$, we have

$$\boldsymbol{S}_\varepsilon\left(x_{ij}\right) = \begin{cases} x_{ij} - \varepsilon & when\ x_{ij} > \varepsilon \\ x_{ij} + \varepsilon & when\ x_{ij} < -\varepsilon \\ 0 & other \end{cases} \tag{15}$$

According to $f(\boldsymbol{W})$, the Lipschitz constant is calculated as:

$$\begin{aligned} \|f(\boldsymbol{W}_1) - f(\boldsymbol{W}_2)\|_F^2 &= \left\|\boldsymbol{X}^T\boldsymbol{X}\Delta\boldsymbol{W}\right\|_F^2 + \|2\gamma\Delta\boldsymbol{W}\boldsymbol{R}\|_F^2 + \|2\beta\Delta\boldsymbol{W}\boldsymbol{A}\|_F^2 \\ &\le 2\left\|\boldsymbol{X}^T\boldsymbol{X}\right\|_2^2\|\Delta\boldsymbol{W}\|_F^2 + 4\gamma\|\boldsymbol{C}\|_2^2\|\Delta\boldsymbol{W}\|_F^2 + 4\beta\|\boldsymbol{A}\|_2^2\|\Delta\boldsymbol{W}\|_F^2 \end{aligned} \tag{16}$$

Therefore, the Lipschitz constant for the CCSF model is:

$$L_g = \sqrt{2\left(\left\|\boldsymbol{X}^T\boldsymbol{X}\right\|_2^2 + 2\gamma\|\boldsymbol{C}\|_2^2 + 2\beta\|\boldsymbol{A}\|_2^2\right)} \tag{17}$$

The CCSF algorithm framework is as following:

Algorithm 1: CCSF

```
Input: Training dataset $\{\boldsymbol{X}, \boldsymbol{Y}\}$, parameters $\alpha, \beta, \gamma$
Output: $\boldsymbol{W}$
(1) Initialization: $\boldsymbol{W}_0 = rand(n, \ell)$, $\theta_0 = \theta_1 = 1$, $t = 1$
(2) Calculate the causal relationship between the labels $\boldsymbol{C}$ by the GSBN algorithm
(3) repeat
(4)     The Lipschitz constant is obtained from Eq. (17)
(5)     Update $\boldsymbol{q}_t(\boldsymbol{W}) = \boldsymbol{W}_t - \frac{1}{L_g}\nabla f(\boldsymbol{W})$ by proximal gradient descent
(6)     $\boldsymbol{W}^{(t)} = \boldsymbol{W}_{\boldsymbol{t}} + \frac{\theta_{t-1} - 1}{\theta_t}(\boldsymbol{W}_t - \boldsymbol{W}_{t-1})$
(7)     Update $\boldsymbol{W}_{t+1}$ by Eq. (14)
(8)     $\boldsymbol{W}^{(t+1)} \leftarrow \boldsymbol{W}^{(t)}$
(9)     $\theta_{t+1} = \left(1 + \sqrt{4\theta_t + 1}\right)/2$
(10)    $t = t + 1$
(11) until convergence
(12) return $\boldsymbol{W}$
```

The validation method is as follows. X_{test} stands for testing dataset. The matrix dimension m is the sample size of the remainder of the test set. Y_{test} represents predictive matrix. S_{test} represents score matrix.

Algorithm 2: Test of CCSF

```
Input: $X_{test} \in R^{m \times d}$, $W \in R^{d \times l}$;
Output: $Y_{test}$, $S_{test}$;
$S_{test} \leftarrow X_{test}W$;
$Y_{test} = sign(S_{test})$.
```

3.3 Complexity Analysis

The time complexity analysis of CCSF and comparison algorithms is shown in Table 1, where n represents the number of samples, d represents the number of features, and l represents the number of labels. The time complexity of CCSF consists of computing the asymmetric correlation matrix and accelerated gradient descent method, which results in $O\left(d^2l + ndl + \mathrm{d}l^2\right)$. According to Table 1, it can be seen that the time complexity of LLSF is lower than that of CCSF, which is $O\left(d^2 + dl + l^2 + nd + nl\right)$, but the classification effect is not as good as that of CCSF. The time complexity of FF-MLLA is not given in the article. The rest of the algorithms have higher time complexity than that of CCSF.

Table 1: Time complexity of the algorithms

Methods	Complexity analysis
LSGL	$O\left(d\left(d^2 + nl + l^2\right) + l\left(l^2 + d^2\right) + n\left(l^2 + d^2\right)\right)$
ACML	$O\left(d^2\left(nl + n + l\right) + l^2\left(n + d + 3/2\right)\right)$
LSML	$O\left(\left(n + l\right)d^2 + \left(n + d\right)l^2 + ndl + d^3 + l^3\right)$
LLSF	$O\left(d^2 + dl + l^2 + nd + nl\right)$
LSI-CI	$O\left(nd^2 + nd + ndl + lg^2 + d^3 + d^2l\right)$
CCSF	$O\left(d^2l + ndl + \mathrm{d}l^2\right)$

4 Experiment

4.1 Datasets

To validate the effectiveness of the algorithm proposed in this paper, five cross-validations were performed on nine multi-label benchmark datasets. The datasets are from different domains, the details of which are shown in Table 2.

Table 2: Multi-label datasets

Datasets	Instance	Feature	Label	Cardinality	Domain
Birds [2]	645	260	20	1.471	Images
Arts [1]	5000	462	26	1.636	Text
Computer [1]	5000	681	33	1.508	Text
Education [1]	5000	550	33	1.461	Text
Entertainment [1]	5000	640	21	1.640	News
Business [1]	5000	438	30	1.438	News
Recreation [1]	5000	606	22	1.606	News
Reference [1]	5000	793	33	1.793	Text
Science [1]	5000	743	40	1.451	Text

Note: [1] http://www.uco.es/kdis/mllresources/. [2] http://mulan.sourceforge.net/datasets-mlc.html.

4.2 Results and Comparison Algorithms

The experimental codes are implemented in MatlabR2021a, with a hardware environment of IntelCore (TM) i5-11600KF 3.90 GHz CPU, 32 G RAM, and an operating system of Windows 10.

In order to compare the effectiveness of CCSF algorithms, six commonly used evaluation metrics in MLL are selected in this paper, which are Hamming Loss (HL), Average Precision (AP), One Error (OE), Ranking Loss (RL), Coverage (CV), and AUC (AUC). Among them, the smaller the HL, OE, RL, CV metrics the better, the larger the AP and AUC metrics the better the experimental effect. Specific formulas and meanings can be found in the literature [28,29]. The parameters of the comparison algorithm are set as follows:

1) In LSGL [16] algorithm, $\lambda_1 \in \left\{10^{-3}, 10^{-2}, \ldots, 10^{3}\right\}$, $\lambda_2, \lambda_3, \lambda_4, \lambda_5 \in \left\{10^{-3}, 10^{-2}, \ldots, 10^{1}\right\}$;

2) The parameters interval of the ACML [17] algorithm are $\alpha \in [2^{-10}, 2^{10}]$, $\beta \in [2^{-10}, 2^{10}]$;

3) Numbers of nearest neighbors in the FF-MLLA [12] algorithm are $k = 15, \beta = 1$, KRBF $= 100$;

4) The parameters of LSML [15] are set as follows $\lambda_1 = 10^{1},\ \lambda_2 = 10^{-5},\ \lambda_3 = 10^{-3},\ \lambda_4 = 10^{-5}$;

5) The parameters of LLSF [11] are set to $\alpha = 2^{-4},\ \beta = 2^{-6},\ \gamma = 1$;

6) The parameters of LSI-CI [30] are set to $\alpha = 2^{10},\ \beta = 2^{8},\ \gamma = 1,\ \theta = 2^{-8}$;

7) The parameters of CCSF are set as $\alpha,\ \beta,\ \gamma \in [2^{-10},\ 2^{10}]$.

The experimental results of the CCSF algorithm on 9 datasets with 6 state-of-the-art algorithms under 6 different metrics are given in Table 2, where "↑" ("↓") indicates that higher (lower) values of the metrics are better, and the experimental results that are dominant are bolded. The details are as follows.

1) As can be seen from Table 3, out of the 54 sets of experimental results, the CCSF algorithm is superior in 49 sets, with a superiority rate of 90.74%. The CCSF algorithm significantly outperforms the other compared algorithms on all 8 datasets. The variance of the CCSF algorithm is smaller, which also proves that the CCSF algorithm is more stable. On the Birds dataset, the CCSF algorithm and the ACML algorithm are equally dominant, due to the fact that both algorithms use causal learning algorithms to compute asymmetric correlations between labels. While the Birds dataset is small, it is difficult to extract more common features of the labels, and the experimental effect dominance is not obvious compared to the larger dataset.

Table 3: Test results of each algorithms on six evaluation metrics (mean ± std)

Data sets	Metrics	CCSF	LSGL	ACML	FF-MLLA	LSML	LLSF	LSF-CI
Birds	HL↓	**0.0470 ± 0.0056**	0.0505 ± 0.0047	0.0512 ± 0.0023	0.0476 ± 0.0040	0.0595 ± 0.0046	0.0506 ± 0.0041	0.0647 ± 0.0063
	AP↑	**0.7710 ± 0.0314**	0.7664 ± 0.0128	0.7648 ± 0.0275	0.7517 ± 0.0247	0.7596 ± 0.0219	0.7582 ± 0.0350	0.6302 ± 0.0165
	OE↓	**0.2742 ± 0.0503**	0.2806 ± 0.0186	0.2821 ± 0.0301	0.3007 ± 0.0358	0.2884 ± 0.0458	0.2915 ± 0.0385	0.4062 ± 0.0233
	RL↓	0.0915 ± 0.0234	0.0948 ± 0.0128	**0.0891 ± 0.0150**	0.1067 ± 0.0116	0.0972 ± 0.0061	0.0962 ± 0.0232	0.2206 ± 0.0168
	CV↓	0.1425 ± 0.0285	0.1486 ± 0.0229	**0.1391 ± 0.0210**	0.1512 ± 0.0206	0.1483 ± 0.0162	0.1471 ± 0.0322	0.2739 ± 0.0156
	AUC↑	0.8787 ± 0.0204	0.8702 ± 0.0184	**0.8953 ± 0.0460**	0.7714 ± 0.0145	0.6498 ± 0.0040	0.7690 ± 0.0178	0.6978 ± 0.0145
Arts	HL↓	**0.0525 ± 0.0014**	0.0529 ± 0.0009	0.0536 ± 0.0007	0.0588 ± 0.0015	0.0582 ± 0.0011	0.0566 ± 0.0009	0.0561 ± 0.0013
	AP↑	**0.6367 ± 0.0112**	0.6340 ± 0.0069	0.6241 ± 0.0141	0.5211 ± 0.0101	0.5932 ± 0.0069	0.5852 ± 0.0147	0.5451 ± 0.0100
	OE↓	**0.4402 ± 0.0157**	0.4454 ± 0.0118	0.4524 ± 0.0179	0.6070 ± 0.0191	0.4762 ± 0.0088	0.4900 ± 0.0181	0.5090 ± 0.0180
	RL↓	**0.1098 ± 0.0052**	0.1263 ± 0.0029	0.1405 ± 0.0074	0.1571 ± 0.0031	0.1770 ± 0.0058	0.1841 ± 0.0106	0.2621 ± 0.0106
	CV↓	**0.1686 ± 0.002**	0.1970 ± 0.0034	0.2141 ± 0.0080	0.2212 ± 0.0050	0.2567 ± 0.0060	0.2650 ± 0.0117	0.3448 ± 0.0107
	AUC↑	**0.8573 ± 0.0053**	0.8316 ± 0.0026	0.7832 ± 0.0891	0.5558 ± 0.0059	0.6723 ± 0.0178	0.6916 ± 0.0114	0.7102 ± 0.0007
Computers	HL↓	**0.0324 ± 0.0017**	0.0332 ± 0.001	0.0339 ± 0.0009	0.0383 ± 0.0010	0.0391 ± 0.0008	0.0389 ± 0.0010	0.0415 ± 0.0018
	AP↑	**0.7193 ± 0.014**	0.7171 ± 0.0121	0.7093 ± 0.0163	0.6424 ± 0.0041	0.6915 ± 0.0059	0.6575 ± 0.0064	0.5839 ± 0.0059
	OE↓	**0.3366 ± 0.0177**	0.3416 ± 0.0216	0.3466 ± 0.0171	0.4302 ± 0.0072	0.3608 ± 0.0060	0.4080 ± 0.0094	0.4614 ± 0.0072
	RL↓	**0.0655 ± 0.005**	0.0841 ± 0.0051	0.0980 ± 0.0086	0.0974 ± 0.0044	0.1230 ± 0.0059	0.1229 ± 0.0059	0.2299 ± 0.0126
	CV↓	**0.1017 ± 0.0082**	0.1261 ± 0.0044	0.1406 ± 0.0106	0.1402 ± 0.0045	0.1725 ± 0.0051	0.1720 ± 0.0088	0.2888 ± 0.0170
	AUC↑	**0.9078 ± 0.007**	0.8878 ± 0.0065	0.7828 ± 0.0916	0.6764 ± 0.0041	0.7505 ± 0.0165	0.6810 ± 0.0136	0.7813 ± 0.0068
Education	HL↓	**0.0369 ± 0.0017**	0.0369 ± 0.0010	0.0371 ± 0.0008	0.0407 ± 0.0002	0.0411 ± 0.0003	0.0414 ± 0.0007	0.0418 ± 0.0012
	AP↑	**0.6437 ± 0.0192**	**0.6437 ± 0.0083**	0.6337 ± 0.0153	0.5497 ± 0.0050	0.6033 ± 0.0082	0.5805 ± 0.0069	0.5290 ± 0.0166
	OE↓	0.4622 ± 0.0274	0.4542 ± 0.0154	**0.4606** ± 0.0203	0.5868 ± 0.0079	0.4826 ± 0.0178	0.5090 ± 0.0070	0.5290 ± 0.0166
	RL↓	**0.0700 ± 0.0054**	0.0953 ± 0.0054	0.1089 ± 0.0057	0.1001 ± 0.0053	0.1526 ± 0.0068	0.1642 ± 0.0065	0.2486 ± 0.0081
	CV↓	**0.0981 ± 0.0062**	0.1425 ± 0.0081	0.1592 ± 0.0086	0.1323 ± 0.0067	0.2123 ± 0.0077	0.2215 ± 0.0068	0.3133 ± 0.0115
	AUC↑	**0.9165 ± 0.0056**	0.8807 ± 0.0082	0.8709 ± 0.0190	0.5612 ± 0.0037	0.6435 ± 0.0003	0.6660 ± 0.0160	0.6784 ± 0.0532
Entertain	HL↓	**0.0491 ± 0.0021**	0.0505 ± 0.0017	0.0508 ± 0.0014	0.0589 ± 0.0005	0.0570 ± 0.0006	0.0550 ± 0.0015	0.0550 ± 0.0014
	AP↑	**0.7052 ± 0.0134**	0.7002 ± 0.0108	0.6925 ± 0.0067	0.5777 ± 0.0110	0.6731 ± 0.0089	0.6669 ± 0.0071	0.6351 ± 0.0076
	OE↓	**0.3772 ± 0.0231**	0.3848 ± 0.0157	0.3912 ± 0.0106	0.5668 ± 0.0178	0.4072 ± 0.0113	0.4092 ± 0.0070	0.4166 ± 0.0063
	RL↓	**0.0875 ± 0.0036**	0.1019 ± 0.0053	0.1163 ± 0.0035	0.1284 ± 0.0047	0.1422 ± 0.0040	0.1460 ± 0.0110	0.2215 ± 0.0096
	CV↓	**0.1234 ± 0.0060**	0.1456 ± 0.0073	0.1605 ± 0.0066	0.1661 ± 0.0052	0.1897 ± 0.0044	0.1918 ± 0.0114	0.2717 ± 0.0109
	AUC↑	**0.8844 ± 0.0062**	0.8648 ± 0.0056	0.8013 ± 0.0009	0.5879 ± 0.0035	0.6128 ± 0.0034	0.7600 ± 0.1127	0.5699 ± 0.0901
Business	HL↓	**0.0239 ± 0.0013**	0.0245 ± 0.0005	0.0266 ± 0.0004	0.0261 ± 0.0016	0.0287 ± 0.0010	0.0295 ± 0.0009	0.0398 ± 0.0008
	AP↑	**0.8978 ± 0.0082**	0.8898 ± 0.0068	0.8809 ± 0.0072	0.8805 ± 0.0110	0.8798 ± 0.0085	0.8484 ± 0.0090	0.7825 ± 0.0116
	OE↓	**0.1024 ± 0.0144**	0.1086 ± 0.0102	0.1158 ± 0.0074	0.1140 ± 0.0101	0.1104 ± 0.0105	0.1452 ± 0.0112	0.2240 ± 0.0185
	RL↓	**0.0304 ± 0.0037**	0.0386 ± 0.0014	0.0443 ± 0.0048	0.0452 ± 0.0049	0.0485 ± 0.0048	0.0635 ± 0.0041	0.1036 ± 0.0071
	CV↓	**0.0651 ± 0.0057**	0.0794 ± 0.0019	0.0895 ± 0.0078	0.0833 ± 0.0068	0.0967 ± 0.0097	0.1096 ± 0.0064	0.1559 ± 0.0092
	AUC↑	**0.9503 ± 0.0060**	0.9371 ± 0.0028	0.8971 ± 0.0142	0.8520 ± 0.0075	0.7990 ± 0.0005	0.7168 ± 0.0093	0.7009 ± 0.0141

(Continued)

Table 3 (continued)

Data sets	Metrics	CCSF	LSGL	ACML	FF-MLLA	LSML	LLSF	LSF-CI
Reference	HL↓	**0.0239** ± **0.0010**	0.0251 ± 0.0007	0.0257 ± 0.0008	0.0292 ± 0.0004	0.0294 ± 0.0010	0.0280 ± 0.0006	0.0298 ± 0.0014
	AP↑	**0.7269** ± **0.0154**	0.7249 ± 0.0051	0.7135 ± 0.0033	0.6301 ± 0.0078	0.7052 ± 0.0072	0.6634 ± 0.0129	0.5929 ± 0.0173
	OE↓	**0.3534** ± **0.0205**	0.3582 ± 0.0062	0.3642 ± 0.0090	0.4658 ± 0.0084	0.3666 ± 0.0088	0.4020 ± 0.0144	0.4692 ± 0.0178
	RL↓	**0.0573** ± **0.0046**	0.0709 ± 0.0049	0.0930 ± 0.0047	0.0934 ± 0.0047	0.1070 ± 0.0060	0.1398 ± 0.0087	0.2426 ± 0.0149
	CV↓	**0.0716** ± **0.0045**	0.0923 ± 0.0067	0.1194 ± 0.0079	0.1100 ± 0.0057	0.1354 ± 0.0072	0.1705 ± 0.0104	0.2745 ± 0.0171
	AUC↑	**0.9277** ± **0.0046**	0.9086 ± 0.0072	0.7505 ± 0.056	0.6461 ± 0.0035	0.7234 ± 0.0093	0.6983 ± 0.0023	0.6728 ± 0.0003
Recreation	HL↓	**0.0517** ± **0.0019**	0.0541 ± 0.0010	0.0535 ± 0.0011	0.9361 ± 0.0016	0.0578 ± 0.0008	0.0571 ± 0.0013	0.0565 ± 0.0003
	AP↑	**0.6539** ± **0.0110**	0.6509 ± 0.0072	0.6391 ± 0.0043	0.4892 ± 0.0039	0.6185 ± 0.0097	0.5985 ± 0.0148	0.5692 ± 0.0097
	OE↓	**0.4334** ± **0.0181**	0.4408 ± 0.0099	0.4444 ± 0.0092	0.6616 ± 0.0071	0.4614 ± 0.0102	0.4890 ± 0.0248	0.5056 ± 0.0113
	RL↓	**0.1194** ± **0.0045**	0.1291 ± 0.0032	0.1485 ± 0.0037	0.1830 ± 0.0015	0.1741 ± 0.0107	0.1868 ± 0.0045	0.2446 ± 0.0075
	CV↓	**0.1620** ± **0.0052**	0.1776 ± 0.0049	0.1992 ± 0.0059	0.2221 ± 0.0028	0.2277 ± 0.0123	0.2392 ± 0.0044	0.2968 ± 0.0088
	AUC↑	**0.8448** ± **0.0071**	0.8305 ± 0.0045	0.8022 ± 0.0071	0.5339 ± 0.0017	0.6991 ± 0.0189	0.7764 ± 0.0080	0.6101 ± 0.0019
Science	HL↓	**0.0302** ± **0.0008**	0.0311 ± 0.0006	0.0311 ± 0.0006	0.0348 ± 0.0008	0.0333 ± 0.0007	0.0348 ± 0.0007	0.0348 ± 0.0007
	AP↑	**0.6144** ± **0.0189**	**0.6144** ± **0.0084**	0.6077 ± 0.0081	0.4556 ± 0.0135	0.5890 ± 0.0158	0.5521 ± 0.0105	0.5166 ± 0.0116
	OE↓	**0.4746** ± **0.027**	0.4768 ± 0.0095	0.4772 ± 0.0108	0.6694 ± 0.0146	0.4884 ± 0.0207	0.5274 ± 0.0075	0.5512 ± 0.0191
	RL↓	**0.0933** ± **0.009**	0.1142 ± 0.0088	0.1289 ± 0.0046	0.1568 ± 0.0025	0.1530 ± 0.0106	0.1774 ± 0.0083	0.2473 ± 0.0056
	CV↓	**0.1280** ± **0.0109**	0.1605 ± 0.0128	0.1778 ± 0.0058	0.1978 ± 0.0057	0.2048 ± 0.0143	0.2296 ± 0.0087	0.3027 ± 0.0065
	AUC↑	0.8829 ± 0.0098	0.8557 ± 0.0109	0.8600 ± 0.0003	0.5346 ± 0.0028	0.6415 ± 0.0340	**0.8962** ± **0.0031**	0.7624 ± 0.0007

2) The CCSF algorithm significantly outperforms the ACML algorithm on these 54 sets of experimental results. This is because the ACML algorithm only takes into account the asymmetric relationship between the labels and does not take into account the fact that the common features of the labels also have a very significant role in multi-label classification.

3) The CCSF algorithm significantly outperforms the traditional LLSF algorithm and the LSGL algorithm. The reason is that the LLSF algorithm only considers the global correlation of labels. The LSGL algorithm is superior to the LLSF algorithm, which is because the LSGL algorithm not only considers the global correlation of labels, but also considers the local correlation of labels. Both of them do not consider the causal relationship between the labels and do not take into account that the common features of labels can effectively improve the performance of multi-label classification algorithms. However, we adopt a global causality and do not consider the local causality between labels, which is also a defect of the algorithm in this paper.

4) The experimental results of the CCSF algorithm for the average ranking of six evaluation metrics on nine datasets are demonstrated in Table 4, which also fully proves that the adoption of causal correlation and common features of labels can effectively improve the classification performance of the LSF model.

Table 4: AVG results of each algorithms on five evaluation metrics

Metrics	Average ranking						
	CCSF	LSGL	ACML	FF-MLLA	LSML	LLSF	LSF-CI
HL↓	**1.0556**	2.1667	3.2222	4.8889	5.5556	5.0000	6.1111
AP↑	**1.0556**	1.9444	3.0000	6.2222	4.1111	5.1111	6.5556
OE↓	**1.2222**	1.8889	2.8889	6.3333	4.1111	5.1111	6.4444
RL↓	**1.1111**	2.1111	3.0000	4.2222	4.8889	5.6667	7.0000
CV↓	**1.1111**	2.4444	3.2222	3.6667	5.0000	5.5556	7.0000
AUC↑	**1.2222**	2.3333	2.7778	6.2222	5.4444	4.5556	5.4444

4.3 Parameter Sensitivity Analysis

The CCSF algorithm has three main hyperparameters. α and β jointly adjust the contribution of the matrix $\boldsymbol{W}$, where α controls the contribution of the private features of the labels and β controls the contribution of the common features of the labels. γ controls the effect of asymmetric LC on the model. In order to test the sensitivity of the CCSF model, we control the other two parameters unchanged and adjust one parameter at $[2^{-10}, 2^{10}]$ for the experiment, respectively, and the experimental results are shown in Fig. 2. $\chi = 2^x$ denotes the log function of log with base 2. As shown in the figure, our algorithms all have better experimental results in general, although there are some fluctuations in $[2^{-10}, 2^{10}]$, which may also be due to the small intervals set by our algorithms. We suggest setting the parameters $\alpha = 2^4, \beta = 2^4, \gamma = 2^4$.

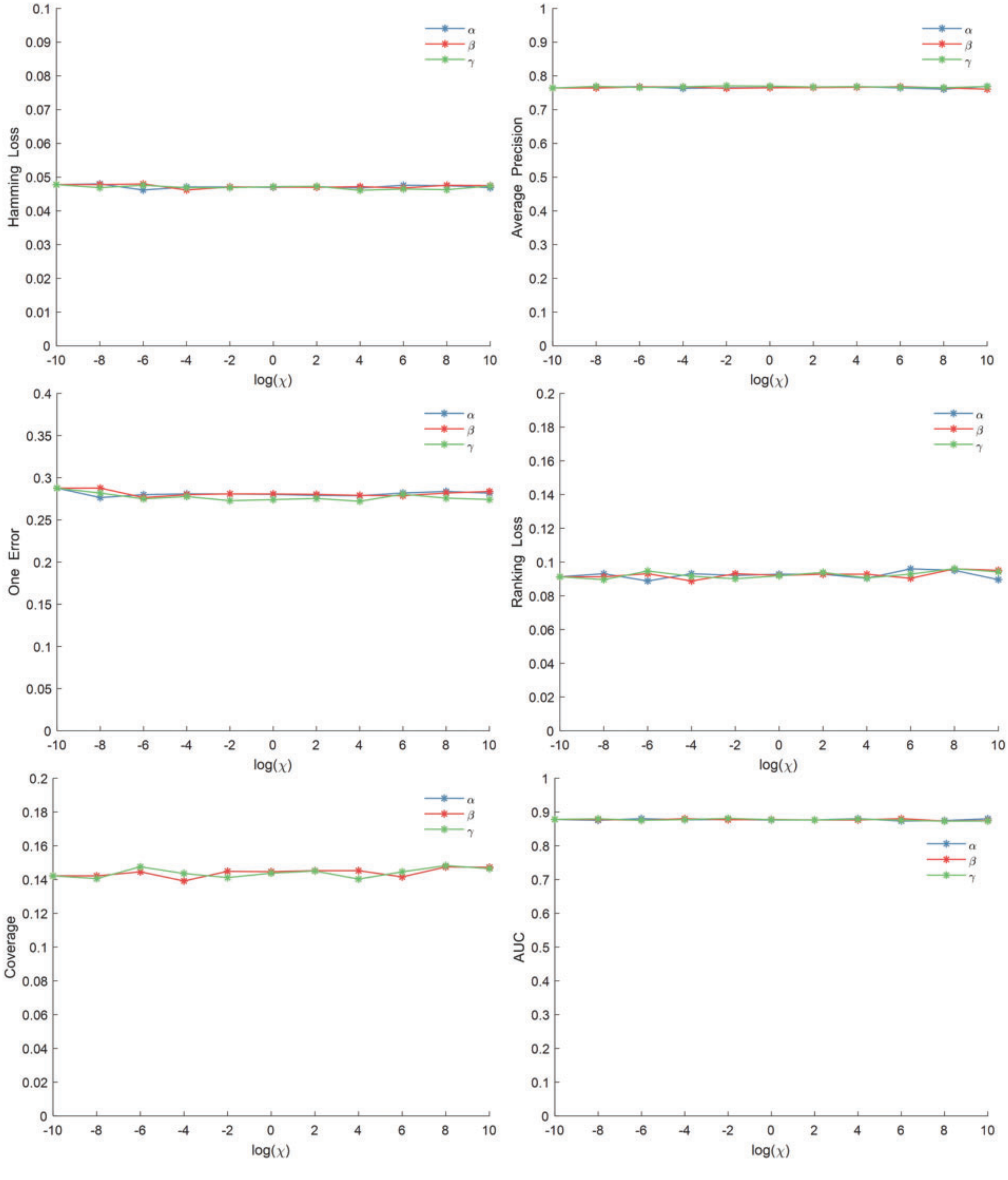


Figure 2: Parameter sensitivity analysis on the Birds dataset

4.4 Component Analysis

In order to verify that introducing common features of labels in the model can effectively improve the performance of multi-label LSF learning algorithms. We conducted component analysis experiments on nine datasets. We compare the CCSF algorithm, which combines the common and private features of label, with the CSF algorithm, which considers only the private features of label. The experimental results are shown in Fig. 3, where the CCSF algorithm outperforms the CSF algorithm on multiple datasets. This indicates that considering the common and private features of labels can effectively improve the performance of LSF algorithm. It also demonstrates that common feature learning of labels introduced into multi-label classification algorithms can improve the accuracy of the algorithms.

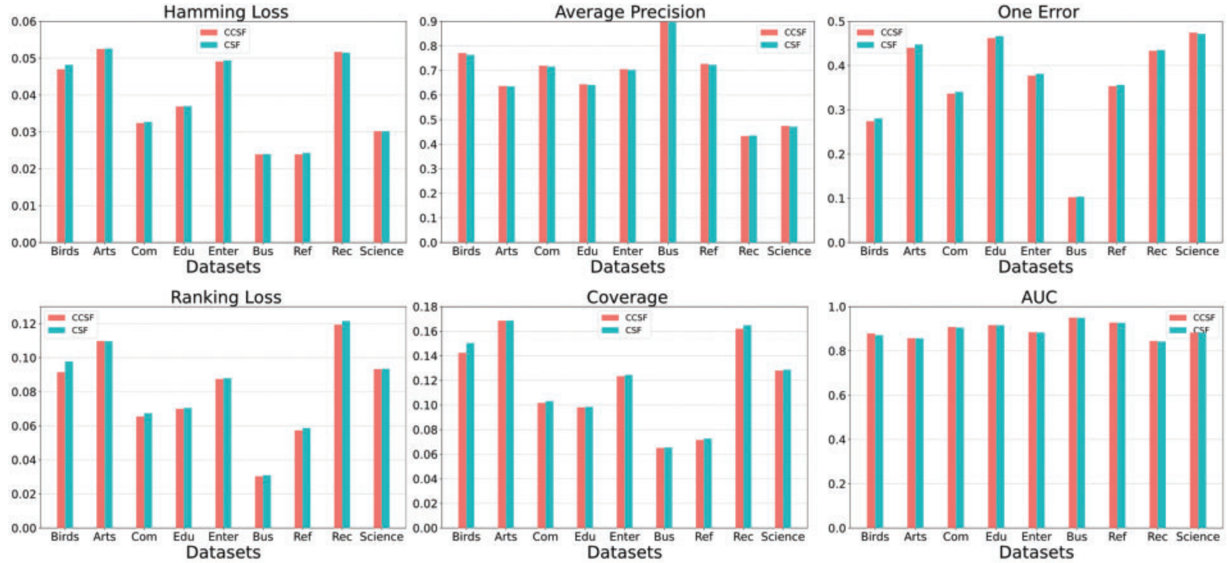


Figure 3: Component analysis on nine datasets

4.5 Statistical Hypothesis Testing

The statistical hypothesis tests in this paper are all based on a significance level of $\theta = 0.05$. The Friedman test [31] was first used to evaluate the comprehensive performance of the CCSF algorithm on all datasets. The obtained F_F is compared with the critical value of the F-test. If it is greater, the original hypothesis is rejected, and vice versa. The experimental results are shown in Table 5. The F_F of the CCSF algorithm is greater than the critical value for all evaluation metrics, so the original hypothesis is rejected for all of them.

Table 5: The Friedman statistics F_F of the critical value and each evaluation metric

Metrics	F_F	Critical value
HL	25.4452	2.2950
AP	153.2800	
OE	108.3077	

(Continued)

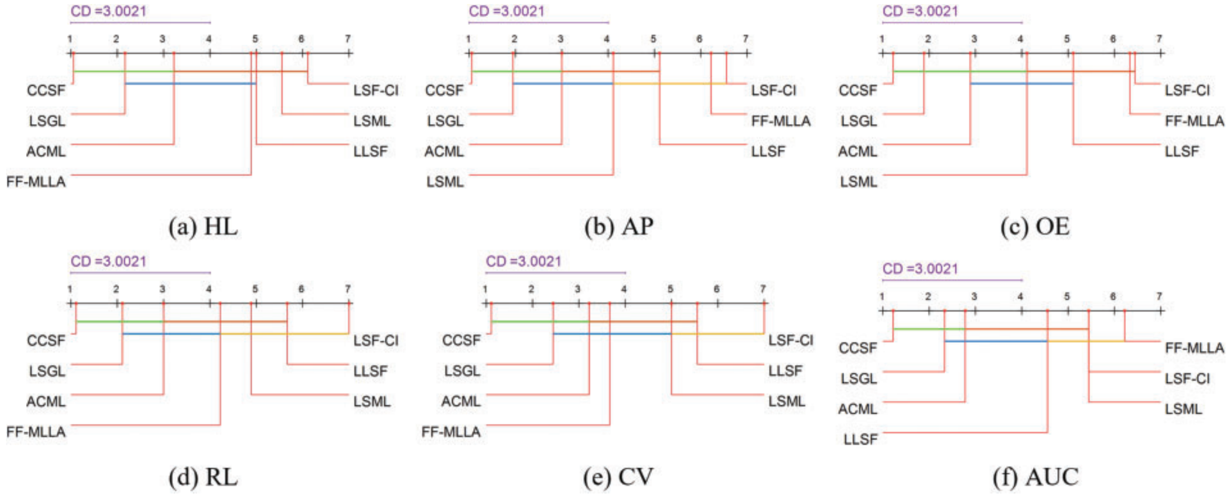
Table 5 (continued)

Metrics	F_F	Critical value
RL	82.7200	
CV	46.6504	
AUC	25.9775	

Nemenyi test [32] is then used to compare the CCSF algorithm with the other six algorithms on all datasets. A significant difference exists when the difference between the average rankings of the two algorithms on all datasets is greater than the Critical Difference (CD) and vice versa. CD value is calculated as follows:

$$CD = q_\theta \sqrt{\frac{K\left(K+1\right)}{6N}} \tag{18}$$

where K = 7, N = 9, q_θ = 2.9480, CD = 3.0021. Fig. 4 demonstrates the CCSF algorithm compared to other algorithms on six evaluation metrics. The algorithm performance decreases in this way from left to right. There is no significant difference between CCSF algorithm and LSGL and ACML algorithms on HL, AP, RL, CV, AUC metrics, and there is no significant difference between CCSF algorithm and LSGL, ACML, LSML algorithms on OE metrics. Other than, there is a significant difference between the CCSF algorithm and the other algorithms in six evaluation metrics. The effectiveness of the algorithm proposed in this paper can be seen from these two statistical hypothesis tests.



(a) HL (b) AP (c) OE

(d) RL (e) CV (f) AUC

Figure 4: Performance comparison of the CCSF algorithm and the comparison algorithm

4.6 Convergence of CCSF

In this paper, the sentiment dataset and the yeast dataset are selected for convergence analysis. As can be seen in Fig. 5, after about forty iterations, the experimental results tend to converge. We conducted the same experiment on other datasets. The convergence results are also similar.

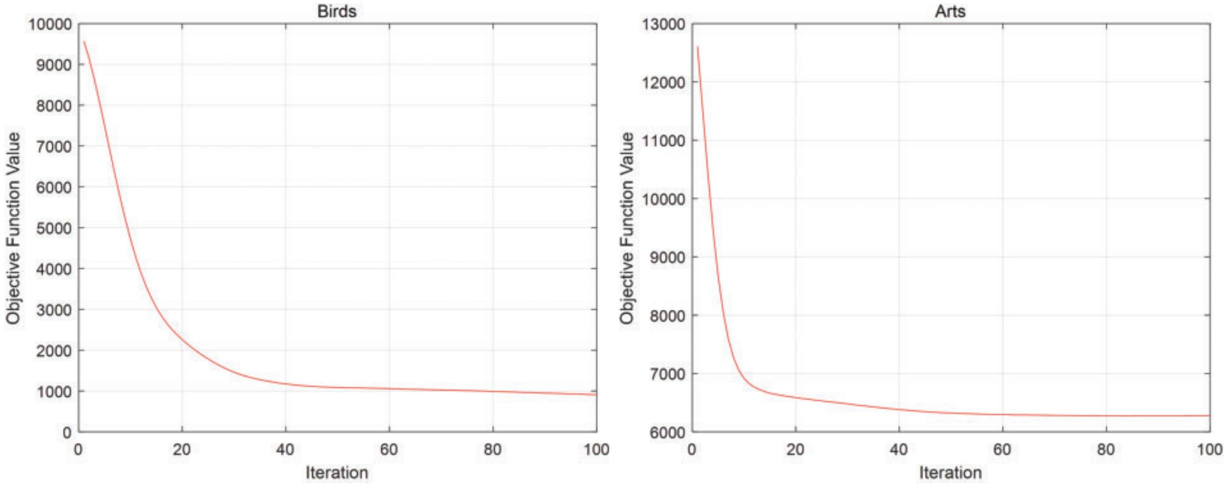


Figure 5: Convergence of CCSF

5 Conclusion

In response to the fact that most of the current LSF learning does not consider the common features of the labels. And only symmetric LC is considered in the calculation of LC. The result is the introduction of much redundant information when classification is performed, which reduces the classification performance of MLL algorithms. Based on the above problem, we use $l_{2,1}$-norm and l_1-norm to extract the common and private features of the labels, respectively. And the asymmetric correlation between labels is calculated utilizing the causal learning algorithm. A large number of experiments are conducted on nine datasets using six evaluation metrics, and the results prove the effectiveness of the algorithm in this paper. But at the same time, we find some problems. We use a global-based causal learning algorithm, which computes the global LC. However, some labels are only associated with local labels and only have local correlation. To minimize the complexity of the model, we also did not utilize instance correlation to improve the classification accuracy of the model. To minimize the complexity of the model, we also did not utilize instance correlation to improve the classification accuracy of the model. In the future, we will try to compute the local correlation of labels using causal learning algorithms and perform experiments in conjunction with instance correlation. We observe the results of the experiments on the complete dataset and try to solve the missing label problem.

Acknowledgement: None.

Funding Statement: 2022 University Research Priorities, No. 2022AH051989.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Y. T. Xu and D. Q. Zhang; analysis and interpretation of results: H. B. Guo and Y. T. Xu; draft manuscript preparation: Y. T. Xu and M. Y. Wang. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: All datasets are publicly available for download. The download URL is in Section 4.1.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] M. L. Zhang and Z. H. Zhou, "A review on multi-label learning algorithms," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 8, pp. 1819–1837, 2013. doi: 10.1109/TKDE.2013.39.

[2] W. Wei *et al.*, "Automatic image annotation based on an improved nearest neighbor technique with tag semantic extension model," *Procedia Comput. Sci.*, vol. 183, no. 24, pp. 616–623, 2021. doi: 10.1016/j.procs.2021.02.105.

[3] T. Qian, F. Li, M. S. Zhang, G. N. Jin, P. Fan and W. Dai, "Contrastive learning from label distribution: A case study on text classification," *Neurocomput.*, vol. 507, no. 7, pp. 208–220, 2022. doi: 10.1016/j.neucom.2022.07.076.

[4] W. Q. Xia *et al.*, "PFmulDL: A novel strategy enabling multi-class and multi-label protein function annotation by integrating diverse deep learning methods," *Comput. Biol. Med.*, vol. 145, pp. 105465, 2022. doi: 10.1016/j.compbiomed.2022.105465.

[5] S. H. Liu, B. Wang, B. Liu, and L. T. Yang, "Multi-community graph convolution networks with decision fusion for personalized recommendation," in *Pacific-Asia Conf. Knowl. Discov. Data Min.*, Chengdu, China, 2022, pp. 16–28.

[6] J. L. Miu, Y. B. Wang, Y. S. Cheng, and F. Chen, "Parallel dual—channel multi-label feature selection," *Soft Comput.*, vol. 27, no. 11, pp. 7115–7130, 2023. doi: 10.1007/s00500-023-07916-4.

[7] Y. B. Wang, W. X. Ge, Y. S. Cheng, and H. F. Wu, "Weak-label-specific features learning based on multidimensional correlation," *J. Nanjing Univ. (Natural Sci.)*, vol. 59, no. 4, pp. 690–704, 2023 (In Chinese).

[8] K. Yu *et al.*, "Causality-based feature selection: Methods and evaluations," *ACM Comput. Surv.*, vol. 53, no. 5, pp. 1–36, 2020.

[9] J. H. Li, P. P. Li, X. G. Hu, and K. Yu, "Learning common and label-specific features for multi-Label classification with correlation information," *Pattern Recogn.*, vol. 121, no. 8, pp. 108257, 2022. doi: 10.1016/j.patcog.2021.108259.

[10] M. L. Zhang and L. Wu, "LIFT: Multi-label learning with label-specific features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 1, pp. 107–120, 2015. doi: 10.1109/TPAMI.2014.2339815.

[11] J. Huang, G. Li, Q. Huang, and X. D. Wu, "Learning label specific features for multi-label classification," in *2015 IEEE Int. Conf. Data Min.*, Atlantic City, NJ, USA, 2015, pp. 181–190.

[12] Y. S. Cheng, K. Qian, Y. B. Wang, and D. W. Zhao, "Multi-label lazy learning approach based on firefly method," *J. Comput. Appl.*, vol. 39, no. 5, pp. 1305–1311, 2019 (In Chinese).

[13] W. Weng, Y. J. Lin, S. X. Wu, Y. W. Li, and Y. Kang, "Multi-label learning based on label-specific features and local pairwise label correlation," *Neurocomput.*, vol. 273, no. 9, pp. 385–394, 2018. doi: 10.1016/j.neucom.2017.07.044.

[14] J. Zhang *et al.*, "Multi label learning with label-specific features by resolving label correlation," *Knowl.-Based Syst.*, vol. 159, no. 8, pp. 148–157, 2018. doi: 10.1016/j.knosys.2018.07.003.

[15] J. Huang *et al.*, "Improving multi-label classification with missing labels by learning label-specific features," *Inf. Sci.*, vol. 492, no. 1, pp. 124–146, 2019. doi: 10.1016/j.ins.2019.04.021.

[16] D. W. Zhao, Q. W. Gao, Y. X. Lu, and D. Sun, "Learning multi-label label-specific features via global and local label correlations," *Soft Comput.*, vol. 26, no. 5, pp. 2225–2239, 2022. doi: 10.1007/s00500-021-06645-w.

[17] J. C. Bao, Y. B. Wang, and Y. S. Cheng, "Asymmetry label correlation for multi-label learning," *Appl. Intell.*, vol. 55, no. 6, pp. 6093–6105, 2022. doi: 10.1007/s10489-021-02725-4.

[18] C. Zhang, Y. S. Cheng, Y. B. Wang, and Y. T. Xu, "Interactive causal correlation space reshape for multi-label classification," *Int. J. Interact. Multimed. Artif. Intell.*, vol. 7, no. 5, pp. 107–120, 2022. doi: 10.9781/ijimai.2022.08.007.

[19] J. Luo, Q. W. Gao, Y. Tan, D. W. Zhao, Y. X. Lu and D. Sun, "Multi label learning based on double Laplace regularization and causal inference," *Comput. Eng.*, vol. 49, pp. 49–60, 2023 (In Chinese).

[20] Y. Tan, D. Sun, Y. Shi, L. Gao, Q. Gao and Y. Lu, "Bi-directional mapping for multi-label learning of label-specific features," *Appl. Intell.*, vol. 52, no. 7, pp. 8147–8166, 2022. doi: 10.1007/s10489-021-02868-4.

[21] J. Zhang *et al.*, "Group-preserving label-specific feature selection for multi-label learning," *Expert. Syst. Appl.*, vol. 213, pp. 118861, 2023. doi: 10.1016/j.eswa.2022.118861.

[22] L. L. Zhang, Y. S. Cheng, Y. B. Wang, and G. S. Pei, "Feature-label dual-mapping for missing label-specific features learning," *Soft Comput.*, vol. 25, no. 14, pp. 9307–9323, 2021. doi: 10.1007/s00500-021-05884-1.

[23] P. Zhao, S. Y. Zhao, X. Y. Zhao, H. T. Liu, and X. Jia, "Partial multi-label learning based on sparse asymmetric label correlations," *Knowl.-Based Syst.*, vol. 245, pp. 108601, 2022. doi: 10.1016/j.knosys.2022.108601.

[24] D. Margaritis and S. Thrun, "Bayesian network induction via local neighborhoods," in *Proc. Conf. Neural Inf. Process. Syst.*, Harrahs and Harveys, Lake Tahoe, USA, 2000, pp. 505–511.

[25] A. Argyriou, T. Evgeniou, and M. Pontil, "Multi-task feature learning," in *Annual Conf. Neural Inf. Process. Syst.*, Vancouver, British Columbia, Canada, 2006, pp. 41–48.

[26] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sci.*, vol. 2, no. 1, pp. 183–202, 2009. doi: 10.1137/080716542.

[27] Z. C. Lin, A. Ganesh, J. Wright, L. Q. Wu, M. M. Chen and Y. Ma, "Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix," *Coordinated Sci. Lab. Report*, vol. 246, pp. 2214, 2009.

[28] D. W. Zhao, Q. W. Gao, Y. X. Lu, and D. Sun, "Learning view-specific labels and label-feature dependence maximization for multi-view multi-label classification," *Appl. Soft Comput.*, vol. 124, no. 8, pp. 109071, 2022. doi: 10.1016/j.asoc.2022.109071.

[29] K. Qian, X. Y. Min, Y. S. Cheng, and F. Min, "Weight matrix sharing for multi-label learning," *Pattern Recogn.*, vol. 136, pp. 109156, 2023. doi: 10.1016/j.patcog.2022.109156.

[30] H. R. Han, M. X. Huang, Y. Zhang, X. G. Yang, and W. G. Feng, "Multi-label learning with label specific features using correlation information," *IEEE Access*, vol. 7, pp. 11474–11484, 2019. doi: 10.1109/ACCESS.2019.2891611.

[31] J. Demsar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach Learn. Res.*, vol. 7, no. 1, pp. 1–30, 2006.

[32] D. Zhao, H. Li, Y. Lu, D. Sun, D. Zhu and Q. Gao, "Multi label weak-label learning via semantic reconstruction and label correlations," *Inf. Sci.*, vol. 623, no. 8, pp. 379–401, 2023. doi: 10.1016/j.ins.2022.12.047.