Optimization of Classifier Chains via Conditional Likelihood Maximization

Lu Sun*, Mineichi Kudo

Graduate School of Information Science and Technology, Hokkaido University, Sapporo 060-0814, Japan

Abstract

Multi-label classification associates an unseen instance with multiple relevant labels. In recent years, a variety of methods have been proposed to handle the multi-label problems. Classifier chains is one of the most popular multi-label methods because of its efficiency and simplicity. In this paper, we consider to optimize classifier chains from the view point of conditional likelihood maximization. In the proposed unified framework, classifier chains can be optimized in either or both of two aspects: label correlation modeling and multi-label feature selection. In this paper we show that previous classifier chains algorithms are specified in the unified framework. In addition, previous information theoretic multi-label feature selection algorithms are specified with different assumptions on the feature and label space. Based on these analyses, we propose a novel multi-label method, k-dependence classifier chains with label-specific features, and demonstrate the effectiveness of the method.

Keywords:

multi-label classification, classifier chains, conditional likelihood maximization, k-dependence Bayesian network, multi-label feature selection.

1. Introduction

Unlike traditional single-label classification where each instance is associated with only one label, Multi-Label Classification (MLC) refers to the problems assigning multiple labels to a single test instance. MLC can be seen in a wide range of real-world applications such as text categorization, semantic image classification, bioinformatics analysis and video annotation. In fact, MLC is ubiquitous in real-world problems. For example, a news article

*Corresponding author

Email addresses: sunlu@main.ist.hokudai.ac.jp (Lu Sun), mine@main.ist.hokudai.ac.jp (Mineichi Kudo)

is possibly relevant to multiple topics, like "science", "technology", "economics", "politics", etc; a single image is probably associated with a set of semantic concepts, like "sky", "sea", "field", "building", etc.

To tackle such multi-label problems, various MLC methods have been proposed. The existing MLC methods fall into two broad categories: problem transformation and algorithm adaptation [1]. As a convenient and straightforward way for MLC, problem transformation strategy transforms an MLC problem into one or a set of single-label classification problems, and learns one or a family of classifiers for modeling the single-label memberships. Most of popular baseline MLC methods, such as Binary Relevance (BR) [2], Calibrated Label Ranking (CLR) [3], and Label Powerset (LP) [4], belong to this strategy. Algorithm adaptation strategy induces conventional machine learning algorithms in the multi-label settings. Various MLC methods adopting one of the above two strategies have been developed and succeeded in dealing with multi-label problems.

Classifier Chains (CC) [5] is one of the most promising MLC methods which follow the problem transformation strategy. Originated from BR which simply ignores label correlations, CC constructs a chain structure on labels and determines the presence/absence of the current label under the condition of previously determined label. CC succeeds in modeling label correlations and achieves higher classification accuracy at similar computational expense with BR. Although CC-based methods have achieved much success in various applications [5, 6, 7], further improvement in classification accuracy is still required. Here we seek the possibility to improve CC in terms of two aspects: label correlation modeling and multi-label feature selection. The intuition behind this idea is that all of the previously determined labels are not always necessary for decision on the current label (necessity of limiting label correlations), and irrelevant and redundant features are usually harmful for the performance of CC (necessity of feature selection). In this paper, we propose a unified framework comprising of both label correlation modeling and multi-label feature selection via conditional likelihood maximization of MLC.

The contributions of this work are cast into three-folds. First, we propose a general framework taking label correlation modeling and multi-label feature selection into account via conditional likelihood maximization. Second, the k-dependence classifier chains method is proposed based on greedy iterative optimization of a sub-problem of likelihood maximization.

Third, a general information theoretic feature selection method is proposed for MLC, where three terms on relevancy, redundancy and label correlations are considered for feature subset selection.

The rest of this paper is organized as follows. Section 2 discusses the related works, mainly focusing on CC-based methods and information theoretic feature selection. Section 3 illustrates the unifying framework for MLC by conditional likelihood maximization, and induces two sub-problems: model selection and multi-label feature selection. Sections 4 and 5 present the solutions on model selection and multi-label feature selection, respectively. Section 6 summarizes several theoretical findings during the development of the proposed method. Section 7 discusses the implementation issues. Section 8 introduces the experiments, and reports the results. Finally, Section 9 concludes this paper and discusses the further research.

2. Related works

Previous efforts have been paid on MLC in terms of various viewpoints, such as label correlations modeling [4, 5], loss function analysis [6, 8, 9], large-scale learning [10, 11] and dimension reduction [12, 13, 14]. In this paper we concentrate mainly on two aspects: label correlations modeling and dimensionality reduction. It has been shown in a number of researches [3, 4, 5] that modeling label correlations is very crucial to perform accurate classification. On the other hand, various dimension reduction algorithms, including of feature selection [15, 16] and feature extraction [12, 17], have been employed in MLC, in order to simplify the learning phase and overcome the curse of dimensionality.

In order to capture label correlations, Classifier chains (CC) based methods [5, 6, 7] have been proposed at tractable computational complexity. CC-based methods originates from Binary Relevance (BR) [2], which simply decomposes a multi-label problem into a set of binary classification problems, totally ignoring label correlations. In this sense, BR is actually a hamming loss risk minimizer [8]. In CC [5], label correlations are expressed in an ordered chain of labels. In the learning phase, according to a predefined chain order, it builds a set of binary classifiers such that each classifier predicts the correct value of a target label by referring to the correct values of all the preceding labels in addition to the features. In the prediction phase, it predicts in turn the value of the target label using the previously

estimated values of its parent labels as extra features. However, the performance of CC is sensitive to the distinct chain orders, and it suffers from the problem of error propagation in the prediction phase. Several efforts have been paid to overcome the limitations of CC. Bayesian Classifier Chains (BCC) [7] introduces a directed tree as the probabilistic structure over labels. The directed tree is established by randomly choosing a label as its root and by assigning directions to the remaining edges. It shares the same model with CC, but restricts the number of parent label no more than 1, which limits its expression ability on label correlations. Probabilistic Classifier Chains (PCC) [6] aims to solve the error propagation problem, providing better estimates than CC at the expense of higher processing time. Although PCC shares the learning model with CC, it chooses the best predictor by searching the *Maximum A Posterior* (MAP) assignment in an exhaustive manner. The exponential cost of PCC in prediction limits its application. To make the prediction tractable for PCC, PCC-beam [18] is proposed by applying beam search to find an approximate MAP assignment of labels to a test instance. MCC [19] utilizes the Monte Carlo scheme to find the sub-optimal chain order and perform efficient inference for the MAP assignment in the learning and prediction phase, respectively. In [20], the dynamic programming technique is used to search the globally optimal chain order of CC. In addition, to speed up the search procedure, a greedy approach is proposed to find locally optimal CC. In a recent work [21], the Classifier Trellis (CT) method is proposed for scalable MLC by extending the 1-dimensional chain of CC to a 2-dimensional trellis structure. CT saves label correlations in the trellis structure, where each label depends only on its adjacent labels. In this way, CT enables to limit the number of parent labels, and thus becomes scalable to the MLC problems with a large number of labels.

In terms of Feature Space Dimensionality Reduction (FS-DR), a variety of traditional supervised dimension reduction approaches have been specifically extended to match the setting of MLC. In [22], a supervised Multi-label Latent Semantic Indexing (MLSI) approach is developed to map the input features into a subspace by preserving the label information. By maximizing the feature-label dependence under the Hilbert-Schmidt independence criterion, Multi-label Dimension reduction via Dependence Maximization (MDDM) [12] derived a closed-form solution to efficiently find the projection into the feature subspace. In addition, several traditional dimension reduction techniques, such as Canonical Correlation Analysis

(CCA) and Linear Discriminant Analysis (LDA), are proposed to handle the MLC problem [23, 24]. In the field of feature selection, an information theoretic approach for Label Powerset (LP) has been developed in [25]. The method introduces a nearest neighbor estimator for computing mutual information, and applies pruned LP to control the problem size. The multivariate mutual information criterion is used in [26] to select useful features. Due to its computational inefficiency, an approximate solution is proposed to estimate the multivariate mutual information. In [15], the authors extend the feature selection framework in [27] to handle two MLC decomposition methods, BR and LP, which achieves significant improvement compared with several multi-label feature selection methods. In fact, all the methods mentioned above can be regarded as *global FS-DR methods*, since they attempt to find an identical feature subspace globally for all the labels. However, it is more reasonable to think that each label holds a specific supporting feature subset. To overcome the limitations, *local FS-DR methods* [17, 16] have been proposed to find label-specific features. In [17], Label-specIfic FeaTures (LIFT) are extracted by conducting cluster analysis on the positive and negative instances of each label. The Learning Label-Specific Features (LLSF) method is proposed in [16]. LLSF selects label-specific features by optimizing the least squares problem with constraints of label correlations and feature sparsity.

3. The unified framework for MLC via likelihood maximization

3.1. Multi-label classification

In the scenario of MLC, an observation $(\mathbf{x}, \mathbf{y})$ consists of a d-dimensional feature vector $\mathbf{x}$ and a q-dimensional target label vector $\mathbf{y}$, drawn from the underlying random variables $\mathbf{X} = (X_1, ..., X_d) \in \mathbb{R}^d$ and $\mathbf{Y} = (Y_1, ..., Y_q) \in \{0, 1\}^q$, respectively. For an observation of labels $\mathbf{y} = (y_1, ..., y_L)$, $y_j = 1$ if the label j is relevant to the instance, and $y_j = 0$ otherwise, $j = 1, ..., q$.

The task of MLC is to find an optimal classifier $h : \mathbb{R}^d \to \{0, 1\}^q$, which assigns a label vector $\hat{\mathbf{y}} = h(\mathbf{x})$ to each instance $\mathbf{x}$ such that h minimizes a loss function between $\hat{\mathbf{y}}$ and $\mathbf{y}$. For a loss function $L(\mathbf{Y}, h(\mathbf{X}))$, the optimal classifier h^* is

$$h^* = \arg\min_h \mathbb{E}_{\mathbf{xy}} L(\mathbf{Y}, h(\mathbf{X})). \tag{1}$$

Specifically, given the subset 0-1 loss $L_s(\mathbf{y}, \hat{\mathbf{y}}) = \mathbb{1}_{\mathbf{y} \neq \hat{\mathbf{y}}}$, where $\mathbb{1}_{(\cdot)}$ denotes the indicator

function. Eq. (1) can be rewritten in a pointwise way,

$$\hat{\mathbf{y}} = h^*(\mathbf{x}) = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x}). \tag{2}$$

Here we use $p(\mathbf{Y}|\mathbf{X})$ to represent the conditional probability distribution of label variables $\mathbf{Y}$ given feature variables $\mathbf{X}$. According to (2), the following two capabilities are necessary for an optimized multi-label classifier h^* in subset 0-1 loss. In the training phase, it should capture the underlying probability distribution $p(\mathbf{Y}|\mathbf{X})$; In the testing phase, given a test instance $\hat{\mathbf{x}}$, the *Maximum A Posteriori* (MAP) assignment $\hat{\mathbf{y}}$ should be obtained by solving the optimization problem $\max_{\mathbf{y}} p(\mathbf{y}|\hat{\mathbf{x}})$. In this study, we focus on training a multi-label classifier with the first capability, i.e., successfully modeling $p(\mathbf{Y}|\mathbf{X})$, in order to learn an approximate optimized classifier in subset 0-1 loss.

3.2. A conditional likelihood view on MLC

According to (2), the objective of MLC is actually to approximate the underlying conditional probability $p(\mathbf{Y}|\mathbf{X})$. In this paper, we assume that the underlying probability $p(\mathbf{Y}|\mathbf{X})$ can be approximated by a Bayesian network B of relatively simple structure with conditional probability $p_B(\mathbf{Y}|\mathbf{X})$, which optimally captures the label correlations. In addition, we further assume that $p_B(\mathbf{Y}|\mathbf{X})$ can be modeled by a subset of $\mathbf{X}$, i.e., the relevant features $\mathbf{X}_\theta \in \mathbf{X}$. A p-dimensional binary vector θ is adopted with 1 indicating selected and 0 unselected. Hence, $\mathbf{X}_\theta/\mathbf{X}_{\bar{\theta}}$ denotes the selected/unselected feature subset, and thus $\mathbf{X} = \{\mathbf{X}_\theta, \mathbf{X}_{\bar{\theta}}\}$. Hence we have $p_B(\mathbf{Y}|\mathbf{X}) = p_B(\mathbf{Y}|\mathbf{X}_\theta)$. Last, a predictive model $f(\mathbf{Y}|\mathbf{X}_\theta, \tau)$ is built to model $p_B(\mathbf{Y}|\mathbf{X}_\theta)$, where τ denotes the parameters of f used to predict $\mathbf{y}$. In this way, we find the optimal Bayesian network B, identify the relevant feature subset $\mathbf{X}_\theta$, and then build the predictive model f.

Given a sample of N observations $\mathcal{D} = \{(\mathbf{x}^i, \mathbf{y}^i)|i = 1, ..., N\}$ drawn from an underlying i.i.d. process $p : \mathbf{X} \mapsto \mathbf{Y}$, the conditional likelihood $\mathcal{L}$ of the observations $\mathcal{D}$ given parameters $\{B, \theta, \tau\}$ becomes

$$\mathcal{L}(B, \theta, \tau|\mathcal{D}) = \prod_{i=1}^{N} f(\mathbf{y}^i|\mathbf{x}^i_\theta, \tau). \tag{3}$$

Therefore, our objective becomes

$$\{B^*, \theta^*, \tau^*\} = \arg\max_{B,\theta,\tau} \mathcal{L}(B, \theta, \tau|\mathcal{D}) = \arg\max_{B,\theta,\tau} \prod_{i=1}^{N} f(\mathbf{y}^i|\mathbf{x}^i_\theta, \tau). \tag{4}$$

For convenience, we use the average *log*-likelihood ℓ instead of $\mathcal{L}$:

$$\ell = \frac{1}{N}\sum_{i=1}^{N} \log f(\mathbf{y}^i|\mathbf{x}^i_\theta, \tau). \tag{5}$$

By taking $p_B(\mathbf{Y}|\mathbf{X}_\theta)$ into consideration, (5) is rewritten as

$$\ell = \frac{1}{N}\sum_{i=1}^{N} \log \left[\frac{f(\mathbf{y}^i|\mathbf{x}^i_\theta, \tau)}{p_B(\mathbf{y}^i|\mathbf{x}^i_\theta)} \cdot \frac{p_B(\mathbf{y}^i|\mathbf{x}^i_\theta)}{p_B(\mathbf{y}^i|\mathbf{x}^i)} \cdot \frac{p_B(\mathbf{y}^i|\mathbf{x}^i)}{p(\mathbf{y}^i|\mathbf{x}^i)} \cdot p(\mathbf{y}^i|\mathbf{x}^i) \right] \tag{6}$$

By the law of large numbers, ℓ approaches in probability the expected version

$$\mathbb{E}_{\mathbf{xy}} \left\{ \log \left[\frac{f(\mathbf{Y}|\mathbf{X}_\theta, \tau)}{p_B(\mathbf{Y}|\mathbf{X}_\theta)} \cdot \frac{p_B(\mathbf{Y}|\mathbf{X}_\theta)}{p_B(\mathbf{Y}|\mathbf{X})} \cdot \frac{p_B(\mathbf{Y}|\mathbf{X})}{p(\mathbf{Y}|\mathbf{X})} \cdot p(\mathbf{Y}|\mathbf{X}) \right] \right\} \tag{7}$$

We negate the above formula to minimize

$$-\ell \approx \mathbb{E}_{\mathbf{xy}} \left\{ \log \frac{p_B(\mathbf{Y}|\mathbf{X}_\theta)}{f(\mathbf{Y}|\mathbf{X}_\theta, \tau)} \right\} + \mathbb{E}_{\mathbf{xy}} \left\{ \log \frac{p_B(\mathbf{Y}|\mathbf{X})}{p_B(\mathbf{Y}|\mathbf{X}_\theta)} \right\} + \mathbb{E}_{\mathbf{xy}} \left\{ \log \frac{p(\mathbf{Y}|\mathbf{X})}{p_B(\mathbf{Y}|\mathbf{X})} \right\} - \mathbb{E}_{\mathbf{xy}} \left\{ \log p(\mathbf{Y}|\mathbf{X}) \right\} \tag{8}$$

Since $\mathbf{X} = \{\mathbf{X}_\theta, \mathbf{X}_{\bar{\theta}}\}$, the second term can be developed as follows,

$$\begin{aligned} \mathbb{E}_{\mathbf{xy}} \left\{ \log \frac{p_B(\mathbf{Y}|\mathbf{X})}{p_B(\mathbf{Y}|\mathbf{X}_\theta)} \right\} &= \mathbb{E}_{\mathbf{xy}} \left\{ \log \frac{p_B(\mathbf{Y}|\mathbf{X}_\theta, \mathbf{X}_{\bar{\theta}})}{p_B(\mathbf{Y}|\mathbf{X}_\theta)} \right\}, \\ &= \mathbb{E}_{\mathbf{xy}} \left\{ \log \frac{p_B(\mathbf{X}_{\bar{\theta}}, \mathbf{Y}|\mathbf{X}_\theta)}{p_B(\mathbf{X}_{\bar{\theta}}|\mathbf{X}_\theta) p_B(\mathbf{Y}|\mathbf{X}_\theta)} \right\}, \\ &= I_{p_B}(\mathbf{X}_{\bar{\theta}}; \mathbf{Y}|\mathbf{X}_\theta), \end{aligned} \tag{9}$$

where $I(\mathbf{X}; \mathbf{Y}|\mathbf{Z})$ denotes the mutual information between $\mathbf{X}$ and $\mathbf{Y}$ conditioned on $\mathbf{Z}$,

$$I(\mathbf{X}; \mathbf{Y}|\mathbf{Z}) = \sum_{\mathbf{xyz}} p(\mathbf{x}, \mathbf{y}, \mathbf{z}) \log \frac{p(\mathbf{x}, \mathbf{y}|\mathbf{z})}{p(\mathbf{x}|\mathbf{z}) p(\mathbf{y}|\mathbf{z})} = \mathbb{E}_{\mathbf{xyz}} \log \frac{p(\mathbf{X}, \mathbf{Y}|\mathbf{Z})}{p(\mathbf{X}|\mathbf{Z}) p(\mathbf{Y}|\mathbf{Z})}. \tag{10}$$

Using the K-L divergence [28]:

$$D_{KL}(p||q) = \sum_{\mathbf{x}} p(\mathbf{x}) \log \frac{p(\mathbf{x})}{q(\mathbf{x})} = \mathbb{E}_{\mathbf{x}} \left\{ \log \frac{p(\mathbf{X})}{q(\mathbf{X})} \right\}, \tag{11}$$

(8) is finally rewritten as

$$-\ell \approx D_{KL}(p_B||f) + I_{p_B}(\mathbf{X}_{\bar{\theta}}; \mathbf{Y}|\mathbf{X}_\theta) + D_{KL}(p||p_B) + H(\mathbf{Y}|\mathbf{X}) \tag{12}$$

From (12), our objective function can be decomposed into four different terms.

1. $D_{KL}(p_B||f)$: the K-L divergence between the conditional probability $p_B(\mathbf{Y}|\mathbf{X}_\theta)$ and the predictive model $f(\mathbf{Y}|\mathbf{X}_\theta, \tau)$, which measures how well f approximates p_B given the selected feature subset $\mathbf{X}_\theta$. This parameter τ could be optimized by the predefined predictive model given the optimized parameters B^* and θ^*,

$$\tau^* = \arg\min_{\tau} D_{KL}(p_{B^*}(\mathbf{Y}|\mathbf{X}_{\theta^*})||f(\mathbf{Y}|\mathbf{X}_{\theta^*}, \tau)). \tag{13}$$

   Thus (13) is the *parameter selection* problem. Distinct predictive models would produce the different τ. It depends on our choice of the baseline model f.

2. $I_{p_B}(\mathbf{X}_{\bar{\theta}}; \mathbf{Y}|\mathbf{X}_\theta)$: the mutual information between the unselected features $\mathbf{X}_{\bar{\theta}}$ and the labels $\mathbf{Y}$ conditioned on the selected features $\mathbf{X}_\theta$. This term depends on both the approximate Bayesian network B and the selected features $\mathbf{X}_\theta$. Given the optimized B^*, the optimal θ^* can be obtained as

$$\theta^* = \arg\min_{\theta} I_{p_{B^*}}(\mathbf{X}_{\bar{\theta}}; \mathbf{Y}|\mathbf{X}_\theta). \tag{14}$$

   In fact (14) is the *multi-label feature selection* problem.

3. $D_{KL}(p||p_B)$: the K-L divergence between the underlying probability $p(\mathbf{Y}|\mathbf{X})$ and the approximate probability $p_B(\mathbf{Y}|\mathbf{X})$ modeled by a Bayesian network B, which measures how well p_B approximate p. This term depends only on the Bayesian network B, hence we have

$$B^* = \arg\min_{B} D_{KL}(p(\mathbf{Y}|\mathbf{X})||p_B(\mathbf{Y}|\mathbf{X})). \tag{15}$$

   Note that (15) is actually the *model selection* problem, which aims to find the optimal Bayesian network to capture label correlations.

4. $H(\mathbf{Y}|\mathbf{X})$: the conditional entropy of labels $\mathbf{Y}$ given features $\mathbf{X}$. This term presents the uncertainty in the labels when all features are known, which is a bound on the Bayes error [29]. Since it is independent of all parameters, we can remove this term from our optimization problem.

Based on above discussion, we will take the following strategy:

Optimization Strategy. *We address the problem of multi-label classification with feature selection in three stages: first learning label space structure, then selecting useful feature subset, and last building the predictive model.*

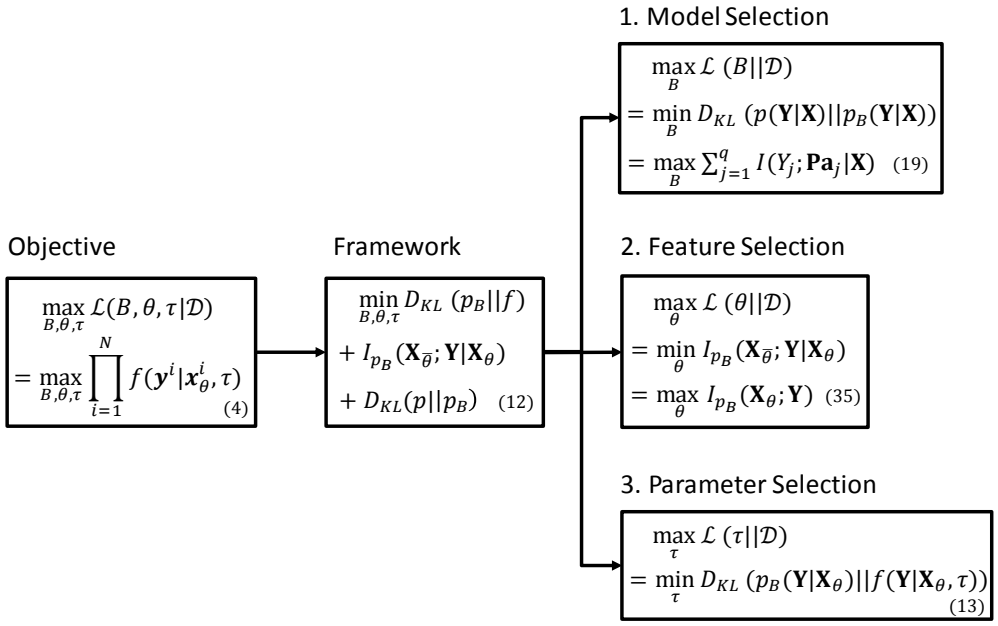


Figure 1: The framework of MLC via conditional likelihood maximization.

Fig. 1 shows the framework of MLC via conditional likelihood maximization following the Optimization Strategy. In this strategy, instead of directly addressing the optimization problem (4), we solve the sub-problems (15), (14), and (13) independently. In the following sections, we shall discuss how the sub-problems (15) and (14) can be solved in Section 4 and 5, respectively. In addition, we see that some popular MLC methods and multi-label feature selection algorithms are embedded in this framework as the special cases of optimization of sub-problems with appropriate assumptions on the label or feature space.

4. Model selection

Under the Optimization Strategy, to model the underlying probability distribution $p(\mathbf{Y}|\mathbf{X})$, the optimal Bayesian network B^* of a special type could be obtained by optimizing

$$\arg\max_B \mathcal{L}(B|\mathcal{D}) = \arg\min_B D_{KL}(p(\mathbf{Y}|\mathbf{X})||p_B(\mathbf{Y}|\mathbf{X})). \tag{16}$$

Here we limit B in the k-Dependence Bayesian (kDB) network B:

$$p_B(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{q} p(Y_j|\mathbf{Pa}_j, \mathbf{X}), \tag{17}$$

where $\mathbf{Pa}_j$ represents the parents of label j, $|\mathbf{Pa}_j| = \min\{j-1, k\}$, $k \in [0, q-1]$. Note that kDB is limited in the number of parents up to k compared with the chain rule in the

canonical order of $Y_1, Y_2, ..., Y_q$:

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{q} p(Y_j|Y_1, Y_2, ..., Y_{j-1}, \mathbf{X}). \tag{18}$$

The optimization problem (16) has been addressed in our previous work [30] as a theorem.

Theorem 1. *To approximate a conditional probability $p(\mathbf{Y}|\mathbf{X})$ in a certain family of Bayesian networks, the optimal Bayesian network B^* in K-L divergence is obtained if the sum of conditional mutual information between each variable of $\mathbf{Y}$ and its parent variables given the observation $\mathbf{X}$ is maximized.*

Proof. The optimization problem of (15) can be developed as follows:

$$\begin{aligned} B^* &= \arg\min_B \mathbb{E}_{\mathbf{xy}} \left\{ \log \frac{p(\mathbf{Y}|\mathbf{X})}{p_B(\mathbf{Y}|\mathbf{X})} \right\} \\ &= \arg\max_B \mathbb{E}_{\mathbf{xy}} \left\{ \log p_B(\mathbf{Y}|\mathbf{X}) \right\} + H(\mathbf{Y}|\mathbf{X}), \end{aligned}$$

$H(\mathbf{Y}|\mathbf{X})$ can be omitted due to its independence with B, thus we have

$$\begin{aligned} B^* &= \arg\max_B \mathbb{E}_{\mathbf{xy}} \left\{ \log p_B(\mathbf{Y}|\mathbf{X}) \right\} \\ &= \arg\max_B \mathbb{E}_{\mathbf{xy}} \left\{ \log \prod_{j=1}^{q} p(Y_j|\mathbf{Pa}_j, \mathbf{X}) \right\} \\ &= \arg\max_B \sum_{j=1}^{q} \mathbb{E}_{\mathbf{xy}} \left\{ \log \frac{p(Y_j, \mathbf{Pa}_j|\mathbf{X})}{p(Y_j|\mathbf{X}) p(\mathbf{Pa}_j|\mathbf{X})} \cdot p(Y_j|\mathbf{X}) \right\} \\ &= \arg\max_B \sum_{j=1}^{q} I(Y_j; \mathbf{Pa}_j|\mathbf{X}) - \sum_{j=1}^{q} H(Y_j|\mathbf{X}). \end{aligned}$$

Since $\sum_{j=1}^{q} H(Y_j|\mathbf{X})$ is independent of B, we reach our conclusion:

$$B^* = \arg\min_B D_{KL}(p(\mathbf{Y}|\mathbf{X})||p_B(\mathbf{Y}|\mathbf{X})) = \arg\max_B \sum_{j=1}^{q} I(Y_j; \mathbf{Pa}_j|\mathbf{X}). \tag{19}$$

□

The objective of (16) is to save label correlations in a Bayesian network B. In [8], a formal definition on two types of label correlations, namely *conditional* and *marginal* dependence, is given. Conditional dependence captures the dependence of labels given a specific instance, while marginal dependence can be considered as the expected dependence averaged over all

instances [8]. According to the definition, the optimal Bayesian network B^* in (19) obtained by Theorem 1 actually models conditional label correlations. For more information on label dependence in MLC, see [8].

According to Theorem 1, the following corollary is derived.

Corollary 1. *Given a specific order of labels, the probability p_B modeled by k-dependence Bayesian network optimally approximates $p(\mathbf{Y}|\mathbf{X})$ in terms of K-L divergence if the parents of the label Y_j holds the restriction $|\boldsymbol{Pa}_j| = j-1$, $j = 1, ..., q$.*

Proof. We assume that labels are ordered in $Y_1, Y_2, ..., Y_q$, the possible parents of label j have been restricted as its previous labels $\mathbf{S}_j = \{Y_1, Y_2, ..., Y_{j-1}\}$, $j = 1, ..., q$. Hence, the optimization problem (19) can be independently solved by each label, and is equivalent to select optimally the parent labels from the previous labels.

$$B^* = \arg\max_B I(Y_j; \mathbf{Pa}_j|\mathbf{X}), \quad j = 1, ..., q. \tag{20}$$

Let $\overline{\mathbf{Pa}}_j$ be $\mathbf{S}_j \setminus \mathbf{Pa}_j$. Based on the chain rule of mutual information, we have

$$I(Y_j; \mathbf{S}_j|\mathbf{X}) = I(Y_j; \mathbf{Pa}_j|\mathbf{X}) + I(Y_j; \overline{\mathbf{Pa}}_j|\mathbf{Pa}_j, \mathbf{X}). \tag{21}$$

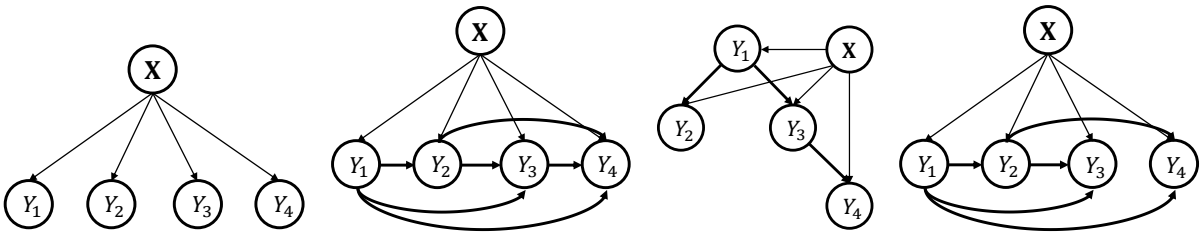
Since conditional mutual information is always non-negative, i.e., $I(Y_j; \overline{\mathbf{Pa}}_j|\mathbf{Pa}_j, \mathbf{X}) \geq 0$, we have $I(Y_j; \mathbf{S}_j|\mathbf{X}) \geq I(Y_j; \mathbf{Pa}_j|\mathbf{X})$. Hence, (20) is optimized by treating all previous labels as the parents of Y_j, i.e., $\mathbf{Pa}_j = \mathbf{S}_j$. In this way, a fully-connected Bayesian network is chosen for B^*, thus $|\mathbf{Pa}_j| = j-1$, $j = 1, ..., q$. □

4.1. Review CC-based methods

For the Classifier Chains (CC) and Probabilistic Classifier Chains (PCC) methods, given a predefined chain order, the classifier for each label is trained by taking the previous labels as extra features. In this sense, a fully-connected Bayesian network is constructed by CC and PCC according to a particular chain order. The difference between CC and PCC is that, in the testing phase, CC finds its prediction by the greedy search:

$$\hat{y}_j = \arg\max_{y_j} p(y_j|\hat{\mathbf{pa}}_j, \hat{\mathbf{x}}), \quad j = 1, ..., q. \tag{22}$$

In contrast, PCC aims to find the MAP assignment by searching 2^q paths according to (2), resulting in an exponential complexity in q for the testing phase [6]. Thus PCC is actually a



(a) BR (b) CC and PCC (c) BCC (d) kCC ($k = 2$)

Figure 2: Graphical models of CC-based methods in order $Y_1 \rightarrow Y_2 \rightarrow Y_3 \rightarrow Y_4$.

risk minimizer in terms of subset 0-1 loss [31], while CC can be considered as a deterministic approximation to PCC. The predictions of PCC ($\hat{\mathbf{y}}_{pcc}$) and CC ($\hat{\mathbf{y}}_{cc}$) shall be equal if the conditional probability of the MAP assignment ($\hat{\mathbf{y}}_{map}$) is more than 0.5 [6], i.e., $\hat{\mathbf{y}}_{cc} = \hat{\mathbf{y}}_{pcc}$, if $p(\hat{\mathbf{y}}_{map}|\hat{\mathbf{x}}) > 0.5$.

As stated in Sec. 3.1, this study focuses on modeling $p(\mathbf{Y}|\mathbf{X})$ rather than finding the MAP assignment. The following corollary is introduced for four CC-based methods, CC, kCC, BCC and BR, which conduct the simple greedy search in the testing phase. Based on Corollary 1, the following corollary on these CC-based methods is derived.

Corollary 2. *In the same chain order of q labels, in the ideal case, classifier chains asymptotically outperform k-dependence classifier chains in terms of subset 0-1 loss, when $k < q-1$. Similarly, k_1-dependence classifier chains asymptotically outperforms k_2-dependence classifier chains in subset 0-1 loss for $0 \leq k_2 < k_1 \leq q-1$.*

Proof. Based on Corollary 1, given a chain order, classifier chains optimally models $p(\mathbf{Y}|\mathbf{X})$ by a fully-connected Bayesian network. Therefore, according to (2), in fact CC is the best approximation of the risk minimizer in subset 0-1 loss among the four CC-based methods. Moreover, for the k-dependence classifier chains, K-L divergence between $p_B(\mathbf{Y}|\mathbf{X})$ and $p(\mathbf{Y}|\mathbf{X})$ always decreases as the value of k increases. According to (2) again, we reach our conclusion. □

Here we note that all of CC, BCC and BR can be viewed as the special cases of kCC by setting the value of k as $q-1$, 1 and 0, respectively. CC works better than Bayesian Classifier Chains (BCC) and Binary Relevance (BR) in approximating ability of the underlying distribution $p(\mathbf{Y}|\mathbf{X})$. Therefore, by (2), CC outperforms BCC and BR in subset 0-1 loss in

an asymptotic case. According to Corollary 2, given the same chain order, as an asymptotic analysis, we have the following order in performance:

$$\text{PCC} \succ \text{CC} \succ k\text{CC} \succ \text{BCC} \succ \text{BR}, \quad 1 < k < q - 1. \tag{23}$$

Here A $\succ$ B presents that A *asymptotically outperforms* B in *subset 0-1 loss*. It is worth noting that, PCC is considered in (23) since it is the exact optimizer in subset 0-1 loss. Fig. 2 shows the probabilistic graphical models of CC-based methods over four labels in a chain order $Y_1 \to Y_2 \to Y_3 \to Y_4$.

4.2. k-dependence Bayesian network for CC

As discussed above, using all preceding labels as $\mathbf{Pa}_j = \mathbf{S}_j$ is optimal for approximating $p(\mathbf{Y}|\mathbf{X})$ in a given chain order. However it is not always true in real-world problems with a finite number of instances. It is often possible to build a better model by eliminating irrelevant and redundant parent labels. Successful elimination of such useless features simplifies the learning model, and gives a better performance. In this study, therefore, we propose to use the k-dependence Bayesian network [32] where at most k preceding labels are adopted as the parents of a label.

As shown in the last section, given a chain order, the optimization problem (19) can solved independently for each label. Hence, it suffices to solve the problem

$$\arg\max_B \mathcal{L}(B|\mathcal{D}) = \arg\max_B I(Y_j; \mathbf{Pa}_j|\mathbf{X}), \;\; j = 1, ..., q, \tag{24}$$

where $\mathbf{Pa}_j \in \mathbf{S}_j$ and $|\mathbf{Pa}_j| \leq k$. However, the calculation of conditional mutual information costs very highly for the high-dimensional feature vectors. Thus, we find an approximation for $I(Y_j; \mathbf{Pa}_j|\mathbf{X})$, relying on some appropriate assumptions.

Theorem 2.

$$\arg\max_B I(Y_j; \boldsymbol{Pa}_j) = \arg\max_B I(Y_j; \boldsymbol{Pa}_j|\boldsymbol{X}),$$

if the label Y_j is independent of the feature vector $\mathbf{X}$ conditioned on the parent labels $\boldsymbol{Pa}_j$.

Proof. According the fact $I(A; B|C) - I(A; B) = I(A; C|B) - I(A; C)$, we have

$$I(Y_j; \mathbf{Pa}_j|\mathbf{X}) = I(Y_j; \mathbf{Pa}_j) + I(Y_j; \mathbf{X}|\mathbf{Pa}_j) - I(Y_j; \mathbf{X}). \tag{25}$$

If the conditional independence assumption holds, the second term of (25) vanishes. In addition, the third term is independent of Bayesian network B. Thus we have the consistence. □

To simplify the computation, a forward parent label selection algorithm is developed here for the optimization problem $B^* = \arg\max_B I(Y_j; \mathbf{Pa}_j)$. Since

$$I(Y_j; \mathbf{Pa}_j \cup Y_l) = I(Y_j; \mathbf{Pa}_j) + I(Y_j; Y_l|\mathbf{Pa}_j), \tag{26}$$

starting from $\mathbf{Pa}_j = \emptyset$, we can update $\mathbf{Pa}_j$ by adding $Y_l \in \overline{\mathbf{Pa}}_j$ such that

$$Y_l = \arg\max_{Y_l \in \overline{\mathbf{Pa}}_j} I(Y_l; Y_j|\mathbf{Pa}_j), \tag{27}$$

until $|\mathbf{Pa}_j|$ reaches a predefined number. $I(Y_l; Y_j|\mathbf{Pa}_j)$ is further developed as,

$$\begin{aligned} I(Y_l; Y_j|\mathbf{Pa}_j) &= I(Y_l; Y_j) + I(Y_j; \mathbf{Pa}_j|Y_l) - I(Y_j; \mathbf{Pa}_j) \\ &= I(Y_l; Y_j) + H(\mathbf{Pa}_j|Y_l) - H(\mathbf{Pa}_j|Y_j, Y_l) - I(Y_j; \mathbf{Pa}_j) \end{aligned} \tag{28}$$

Parent Independence Assumption. *For target label Y_j, one preceding label $Y_k \in \boldsymbol{Pa}_j$, and $Y_l \in \overline{\boldsymbol{Pa}}_j$, we assume:*

$$p(\boldsymbol{Pa}_j|Y_l) = \prod_{Y_k \in \boldsymbol{Pa}_j} p(Y_k|Y_l), \tag{29}$$

$$p(\boldsymbol{Pa}_j|Y_j, Y_l) = \prod_{Y_k \in \boldsymbol{Pa}_j} p(Y_k|Y_j, Y_l). \tag{30}$$

These assumptions require that parent labels are conditional independent given one unselected parent Y_l (29) or given Y_l with the target label Y_j (30).

Under this assumption, we have the following:

$$\begin{aligned} I(Y_l; Y_j|\mathbf{Pa}_j) &= I(Y_l; Y_j) + \sum_{Y_k \in \mathbf{Pa}_j} H(Y_k|Y_l) - \sum_{Y_k \in \mathbf{Pa}_j} H(Y_k|Y_j, Y_l) - I(Y_j; \mathbf{Pa}_j), \\ &= I(Y_l; Y_j) + \sum_{Y_k \in \mathbf{Pa}_j} I(Y_k; Y_j|Y_l) - I(Y_j; \mathbf{Pa}_j). \end{aligned} \tag{31}$$

Substituting (31) into (27) and removing the last term independent of Y_l, we have

$$Y_l = \arg\max_{Y_l \in \overline{\mathbf{Pa}}_j} \left(I(Y_l; Y_j) + \sum_{Y_k \in \mathbf{Pa}_j} I(Y_j; Y_k|Y_l) \right). \tag{32}$$

In (32), the first term captures the *relevance* between the unselected parent label Y_l and the target label Y_j, while the second term models the mutual information between the selected label Y_k and the target label Y_j, conditioned on the unselected label Y_l. Since we have $I(Y_j; Y_k|Y_l) = I(Y_j; Y_k) + I(Y_l; Y_k|Y_j) - I(Y_l; Y_k)$, the second term actually captures both *conditional redundancy* $I(Y_l; Y_k|Y_j)$ and *redundancy* $I(Y_l; Y_k)$. It means that the parent label which has the best trade-off between relevancy and redundancy would be selected. It is worth noting that in [27], the authors reach the similar conclusion for information theoretic feature selection. The difference is that here our objective is to perform parent label selection to find the optimal k-dependence Bayesian network given a chain order.

We can generalize (32) as

$$Y_l = \arg\max_{Y_l \in \overline{\mathbf{Pa}}_j} \left(I(Y_l; Y_j) + \alpha \sum_{Y_k \in \mathbf{Pa}_j} I(Y_j; Y_k|Y_l) \right), \ j = 1, ..., q, \tag{33}$$

where $\alpha \geq 0$ denotes a parameter controlling the weights assigned to the two terms. In practice, we prefer to set $\alpha = \frac{1}{|\mathbf{Pa}_j|}$ to prevent from sweeping the first term as the number of parent labels grows. Based on (33), we propose the k-dependence Classifier Chains (k**CC**) method.

5. Multi-label feature selection

Based on the Optimization Strategy, the following equation can be developed based on the built Bayesian network B,

$$\arg\max_{\theta} \mathcal{L}(\theta|\mathcal{D}) = \arg\min_{\theta} I_{p_B}(\mathbf{X}_{\overline{\theta}}; \mathbf{Y}|\mathbf{X}_{\theta}). \tag{34}$$

Note that there is no monotonicity in the conditional mutual information, so that $\theta = (0, 0, ..., 0)^\intercal$ or $\theta = (1, 1, ..., 1)^\intercal$ is not always the solution. According to the chain rule of mutual information and $\mathbf{X} = \{\mathbf{X}_{\theta}, \mathbf{X}_{\overline{\theta}}\}$, we have

$$I_{p_B}(\mathbf{X}; \mathbf{Y}) = I_{p_B}(\mathbf{X}_{\theta}; \mathbf{Y}) + I_{p_B}(\mathbf{X}_{\overline{\theta}}; \mathbf{Y}|\mathbf{X}_{\theta}). \tag{35}$$

From the above formula, we see that minimization of $I_{p_B}(\mathbf{X}_{\overline{\theta}}; \mathbf{Y}|\mathbf{X}_{\theta})$ is equivalent to maximization of $I_{p_B}(\mathbf{X}_{\theta}; \mathbf{Y})$, thus (34) is transformed into:

$$\arg\max_{\theta} \mathcal{L}(\theta|\mathcal{D}) = \arg\max_{\theta} I_{p_B}(\mathbf{X}_{\theta}; \mathbf{Y}). \tag{36}$$

Note that the similar optimization problem has been proposed by intuition in the field of information theoretic feature selection for the single-label variable Y in a variety of papers [27, 33, 34]. Here we theoretically derive the optimization problem from conditional likelihood maximization for the multi-label case, and take label correlations into account by the learned Bayesian network B.

However, directly solving (36) is a non-trivial thing due to the difficulty on the calculation of mutual information between high-dimensional $\mathbf{X}_\theta$ and $\mathbf{Y}$. Hence, similar with Section 4.2, the greedy sequential optimization is applied for (36). Since

$$I_{p_B}(\mathbf{X}_\theta \cup X_m; \mathbf{Y}) = I_{p_B}(\mathbf{X}_\theta; \mathbf{Y}) + I_{p_B}(X_m; \mathbf{Y}|\mathbf{X}_\theta), \tag{37}$$

Starting from $\mathbf{X}_\theta = \emptyset$, we can update $\mathbf{X}_\theta$ by adding $X_m \in \mathbf{X}_{\overline{\theta}}$ such that

$$X_m = \arg\max_{X_m \in \mathbf{X}_{\overline{\theta}}} I_{p_B}(X_m; \mathbf{Y}|\mathbf{X}_\theta). \tag{38}$$

until $|\mathbf{X}_\theta|$ reaches a predefined number. By taking the Bayesian network B into consideration, we have

$$\begin{aligned} I_{p_B}(X_m; \mathbf{Y}|\mathbf{X}_\theta) &= \sum_{j=1}^{q} I(X_m; Y_j|\mathbf{Pa}_j, \mathbf{X}_\theta), \\ &= \sum_{j=1}^{q} \left(I(X_m; Y_j) + I(\mathbf{Pa}_j, \mathbf{X}_\theta; Y_j|X_m) - I(\mathbf{Pa}_j, \mathbf{X}_\theta; Y_j) \right). \end{aligned} \tag{39}$$

Parent and Feature Independence Assumption. *For each feature $X_n \in \mathbf{X}_\theta$, given one unselected feature $X_m \in \mathbf{X}_{\overline{\theta}}$ and the target labels $\mathbf{Y}$, we assume the following:*

$$\begin{aligned} p(\boldsymbol{Pa}_j, \mathbf{X}_{\theta_j}|Y_j) &= \prod_{Y_k \in \boldsymbol{Pa}_j} p(Y_k|Y_j) \prod_{X_n \in \mathbf{X}_\theta} p(X_n|Y_j), \\ p(\boldsymbol{Pa}_j, \mathbf{X}_\theta|Y_j, X_m) &= \prod_{Y_k \in \boldsymbol{Pa}_j} p(Y_k|Y_j, X_m) \prod_{X_n \in \mathbf{X}_\theta} p(X_n|Y_j, X_m), \end{aligned}$$

It requires that the parents $\boldsymbol{Pa}_j$ and selected features $\mathbf{X}_\theta$ are conditional independent of Y_j with or without the unselected feature X_m.

Based on the above assumption and (39), (38) becomes:

$$X_m = \arg\max_{X_m \in \mathbf{X}_{\overline{\theta}}} \sum_{j=1}^{q} \left(I(X_m; Y_j) + \sum_{Y_k \in \mathbf{Pa}_j} I(Y_j; Y_k|X_m) + \sum_{X_n \in \mathbf{X}_\theta} I(Y_j; X_n|X_m) \right). \tag{40}$$

The first term captures the *Relevance* between the unselected feature X_m and the target label Y_j, and the second term captures *Label Correlations* modeled in k-dependence Bayesian network. For the third term of (40), we express it as

$$I(Y_j; X_n|X_m) = I(Y_j; X_n) + I(X_m; X_n|Y_j) - I(X_m; X_n). \tag{41}$$

Since $I(Y_j; X_n)$ is independent of X_m, $I(Y_j; X_n|X_m)$ can be represented by last two terms. Here we call $I(Y_j; X_n|X_m)$ as *Redundancy Difference*, since it in fact reflects the difference of *Conditional Redundancy* $I(X_m; X_n|Y_j)$ and *Redundancy* $I(X_m; X_n)$. Hence, according to (40), one feature would be selected in that it provides the best trade-off among relevance, label correlations and redundancy.

We can further generalize (40) as,

$$X_m = \underset{X_m \in \mathbf{X}_{\overline{\theta}}}{\arg\max} \sum_{j=1}^{q} \left(I(X_m; Y_j) + \beta \sum_{Y_k \in \mathbf{Pa}_j} I(Y_j; Y_k|X_m) + \gamma \sum_{X_n \in \mathbf{X}_\theta} I(Y_j; X_n|X_m) \right). \tag{42}$$

where $\beta, \gamma \geq 0$ are two parameters controlling the weights on correlation and redundancy difference terms, respectively. In practice, we typically set $\beta = \frac{1}{|\mathbf{Pa}|}$ and $\gamma = \frac{1}{|\mathbf{X}_\theta|}$ to normalize the terms, preventing from sweeping either term as the number of $\mathbf{Pa}_j$ or $\mathbf{X}_{\theta_j}$ grows. This means we hold a relatively weak assumptions on conditional label independence (the 2nd term) and feature independence (the 3rd term).

In the derivation of (42), we assume the relevant features are identical for all the labels. However, it is more general and reasonable to assume that the selected feature subset is label-specific. For the target label j, its relevant features are denoted as $\mathbf{X}_{\theta_j}$, thus $\mathbf{X} = \{\mathbf{X}_{\theta_j}, \mathbf{X}_{\overline{\theta}_j}\}$, $j = 1, ..., q$. Based on the similar derivation and assumption of (42), we can obtain the following feature selection criterion for label j, $j = 1, ..., q$,

$$X_m = \underset{X_m \in \mathbf{X}_{\overline{\theta}_j}}{\arg\max} \left(I(X_m; Y_j) + \beta \sum_{Y_k \in \mathbf{Pa}_j} I(Y_j; Y_k|X_m) + \gamma \sum_{X_n \in \mathbf{X}_{\theta_j}} I(Y_j; X_n|X_m) \right). \tag{43}$$

In this case, we set $\beta = \frac{1}{|\mathbf{Pa}_j|}$ and $\gamma = \frac{1}{|\mathbf{X}_{\theta_j}|}$. Note that the criterion (43) is optimized label-independently, since $X_m \in \mathbf{X}_{\theta_j}$ and $\mathbf{X}_{\theta_j}$ is label-specific. In this paper, we employ (42) and (43) for *global* and *local* Multi-Label Feature Selection (**MLFS**), respectively.

6. Summary of theoretical findings

In Table 1, we summarize the techniques above for comparing model selection and feature selection algorithms. By combining kCC (33) and local MLFS (43), we propose the optimized

Table 1: The generality of the proposed framework for MLC.

	Parameter Setting		Algorithm	Authors
Model Selection	$\lvert\mathbf{Pa}_j\rvert = 0$		Binary Relevance (BR)	Boutell et al. (2004) [2]
	$\lvert\mathbf{Pa}_j\rvert \leq 1$		Bayesian CC (BCC)	Zaragoza et al.(2011) [7]
	$\lvert\mathbf{Pa}_j\rvert = j-1$		Classifier Chains (CC)	Read et al. (2011) [5]
	$\lvert\mathbf{Pa}_j\rvert \leq k$		kCC (Proposed)	Sun and Kudo (2017)
Feature Selection	$q=1$	$\beta=\gamma=0$	MI Maximization (MIM)	Lewis (1992) [35]
		$\beta=0,\ \gamma=\frac{1}{\lvert\mathbf{X}_\theta\rvert}$	Joint MI (JMI)	Yang and Moody (1999)[34]
			Max-Rel Min-Red†	Peng et al. (2005) [33]
	$q>1$	$\beta=\gamma=0$	Multi-Label MIM	Scchidis et al (2012) [15]
		$\beta=0,\ \gamma=\frac{1}{\lvert\mathbf{X}_\theta\rvert}$	Multi-Label JMI	Scchidis et al. (2012) [15]
		$\beta=\frac{1}{\lvert\mathbf{Pa}_j\rvert},\ \gamma=\frac{1}{\lvert\mathbf{X}_\theta\rvert}$	MLFS (Proposed)	Sun and Kudo (2017)

† Max-Rel Min-Red ignores the conditional redundancy term in (42).

Classifier Chains (**oCC**) method, whose procedure is outlined in Algorithm 1.

7. Implementation issues

7.1. Mutual information estimation

In the optimizations of both model and feature selection, calculation of mutual information is extensively used. For the discrete/categorical feature variables $\mathbf{X}$ ($\mathbf{Y}$ is originally binary), the calculation of mutual information is simple and straightforward. Given a sample of N i.i.d. observations $\{x^i, y^i\}_{i=1}^N$, based on the law of large numbers, we have the following approximation:

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \approx \frac{1}{N} \sum_{i=1}^{N} \log \frac{\hat{p}(x^i, y^i)}{\hat{p}(x^i)\hat{p}(y^i)}, \tag{44}$$

where $\hat{p}$ denotes the empirical probability distribution. In contrast, when the feature variable X is continuous, it becomes quite difficult to compute mutual information $I(X;Y)$, since it is typically impossible to obtain $\hat{p}$. One of the solutions is to use kernel density estimation, in which case, $\hat{p}$ is approximated by the following:

$$\hat{p}(x) = \frac{1}{N} \sum_{i=1}^{N} K_h(x - x^i), \tag{45}$$

Algorithm 1 The algorithm of oCC (kCC + MLFS)

Input: $\mathcal{D} = \{\mathbf{x}^i, \mathbf{y}^i\}_{i=1}^N$: training set with q labels, **chain**: label order, $\hat{\mathbf{x}}$: test instance, k: maximal size of parents, M: size of selected features, $f_1, ..., f_q$: q predictive models

Output: $\hat{\mathbf{y}}$: predicted label set of $\hat{\mathbf{x}}$

Training:

$\mathbf{S} \leftarrow \emptyset$, $\{\mathbf{Pa}_j\}_{j=1}^q \leftarrow \emptyset$;
for $j \in$ **chain do**
  for $l = 1, ..., k$ **do**
    select Y_l from $\mathbf{S}$ according to (33);
    $\mathbf{Pa}_j \leftarrow \mathbf{Pa}_j \cup Y_l$;
  for $m = 1, ..., M$ **do**
    select X_m from $\mathbf{X}$ according to (43);
    $\mathbf{X}_{\theta_j} \leftarrow \mathbf{X}_{\theta_j} \cup X_m$;
  build classifier f_j on $\mathcal{D}_j$, where $\mathcal{D}_j = \{\mathbf{x}^i_{\theta_j} \cup \mathbf{pa}^i_j, y^i_j\}_{i=1}^N$;
  $\mathbf{S} \leftarrow \mathbf{S} \cup Y_j$;

Testing:

for $j \in$ **chain do**
  $\hat{y}_j \leftarrow f_j(\hat{\mathbf{x}}_{\theta_j} \cup \hat{\mathbf{pa}}_j)$;
$\hat{\mathbf{y}} \leftarrow (\hat{y}_1, \hat{y}_2, ..., \hat{y}_q)$.

where $K_h(\cdot)$ is a non-negative kernel function with bandwidth h ($h > 0$). Typically, the Gaussian kernel can be used as $K_h(\cdot)$. However, it is computationally expensive to compute (45) and usually difficult to select a good value of the bandwidth h.

Another solution for computing $I(X;Y)$ with continuous feature X is to apply data discretization as preprocessing [27]. In this study, we adopt this method and discretize the continuous variable X based on its mean μ_X and standard deviation σ_X. For example, we divide a continuous value of feature variable X into one of three categories $\{-1, 0, 1\}$ according to $\mu_X \pm \sigma_X$. The experimental results demonstrate the efficiency of such approach for approximating $I(X;Y)$. Note that data discretization is performed only for calculating mutual information in feature selection, which means that original continuous/discrete features will be utilized to train classification models after the selection of useful features.

Algorithm 2 The algorithm of chain order selection

Input: $\mathcal{D} = \{\mathbf{x}^i, \mathbf{y}^i\}_{i=1}^N$: training data with q labels;

Output: $\boldsymbol{\pi}$: selected chain order of q labels;

1: $\mathbf{S}_\pi \leftarrow \emptyset$, $\overline{\mathbf{S}}_\pi \leftarrow \{1, 2, ..., q\}$;

2: Initialize $\boldsymbol{\pi}(1) \leftarrow \arg\max_{\boldsymbol{\pi}(1)} \sum_{j=2}^{q} \sum_{m=1}^{d} I(Y_{\pi(1)}; Y_{\pi(j)}|X_m)$;

3: $\mathbf{S}_\pi \leftarrow \boldsymbol{\pi}(1)$, $\overline{\mathbf{S}}_\pi \leftarrow \mathbf{S}_\pi / \boldsymbol{\pi}(1)$;

4: **for** $j = 2, ..., q$ **do**

5: Select $\boldsymbol{\pi}(j) \in \overline{\mathbf{S}}_\pi$ according to (47);

6: $\mathbf{S}_\pi \leftarrow \mathbf{S}_\pi \cup \boldsymbol{\pi}(j)$, $\overline{\mathbf{S}}_\pi \leftarrow \mathbf{S}_\pi / \boldsymbol{\pi}(j)$;

7.2. Chain order selection

The chain order is crucial to the performance of CC-based methods [5, 6]. To improve the performance, [18] and [19] propose to select the correct chain order by using the Beam Search and Monte Carlo algorithm, respectively. In this study, we develop a greedy algorithm for efficient chain order selection according to Theorem 1. Different from Theorem 1, where model selection is conducted to find the optimal k-dependence Bayesian network ($|\mathbf{Pa}_j| \leq k$, $\forall j$) based on the canonical order, here we aim to perform model selection by selecting the optimal chain order $\boldsymbol{\pi}$ for a fully-connected Bayesian network B_π ($|\mathbf{Pa}_j| = j - 1$, $\forall j$).

Suppose that a permutation $\boldsymbol{\pi} : \{1, 2, ..., q\} \mapsto \{1, 2, ..., q\}$ determines the optimal chain order of labels, so that the jth label is placed at the $\boldsymbol{\pi}(j)$th position of the chain. Thus, following Theorem 1, we have

$$\boldsymbol{\pi}^* = \arg\min_{B_\pi} D_{KL}(p(\mathbf{Y}|\mathbf{X})||p_{B_\pi}(\mathbf{Y}|\mathbf{X})) = \arg\max_{\boldsymbol{\pi}} \sum_{j=1}^{q} I(Y_{\pi(j)}; \mathbf{Pa}_j|\mathbf{X}), \tag{46}$$

where $\mathbf{Pa}_j = \{Y_{\pi(k)}\}_{k=1}^{j-1}$. Similar to the derivation in Section 4.2, a greedy forward search algorithm can be developed to solve the optimization problem in (46) with appropriate independence assumption. In the jth iteration, we select the best place $\boldsymbol{\pi}(j)$ for the jth label according to the following formula:

$$\boldsymbol{\pi}^*(j) = \arg\max_{\boldsymbol{\pi}(j)} \sum_{k=1}^{j-1} \left(\frac{1}{d} \sum_{m=1}^{d} I(Y_{\pi(j)}; Y_{\pi(k)}|X_m) + \frac{1}{k-1} \sum_{l=1}^{k-1} I(Y_{\pi(k)}; Y_{\pi(l)}|Y_{\pi(j)}) \right). \tag{47}$$

Algorithm 2 outlines the procedure of chain order selection. In Sec. 8.5, we shall show the efficiency of Algorithm 2.

Table 2: The statistics of experimental multi-label datasets. In below, N, d and q are the data size in instances, features and labels, respectively. Cardinality, Density and Distinct denotes the label cardinality, label density and number of distinct label combinations, respectively.

Dataset	N	d	q	Cardinality	Density	Distinct	Feature type	Domain
emotions	593	72	6	1.869	0.311	27	numeric	music
scene	2407	294	6	1.074	0.179	15	numeric	image
yeast	2417	103	14	4.237	0.303	198	numeric	biology
birds	645	260	19	1.014	0.053	133	numeric	audio
genbase	662	1186	27	1.252	0.046	32	nominal	biology
medical	978	1449	45	1.245	0.028	94	nominal	text
enron	1702	1001	53	3.378	0.064	753	nominal	text
languagelog	1460	1004	75	1.180	0.016	286	nominal	text
rcv1s1	6000	944	101	2.880	0.029	837	numeric	text
bibtex	7395	1836	159	2.402	0.015	1654	nominal	text
corel16k1	13766	500	153	2.859	0.019	1791	nominal	image
corel5k	5000	499	374	3.522	0.009	1453	nominal	image
delicious	16105	500	983	19.020	0.019	3937	nominal	text

8. Experiments

8.1. Datasets and evaluation metrics

We conducted experiments on thirteen benchmark datasets in Mulan [36] and Meka [37], in order to evaluate the performance of the proposed method. The statistics of the datasets are summarized in Table 2, where "Cardinality" and "Density" denote the label cardinality and label density, respectively. In addition, "Distinct" denotes the number of distinct label combinations, and "Feature type" presents the property of features.

Given a test dataset $\mathcal{T} = \{\mathbf{x}^i, \mathbf{y}^i\}_{i=1}^{N_T}$, we use four evaluation metrics to report the experimental results. Here $\mathbb{1}$ denotes the indicator function.

- **Exact-Match** $\coloneqq \frac{1}{N_T}\sum_{i=1}^{N_T} \mathbb{1}_{\hat{\mathbf{y}}^i=\mathbf{y}^i}$,
- **Hamming-Score** $\coloneqq \frac{1}{N_T}\sum_{i=1}^{N_T} \frac{1}{L}\sum_{\ell=1}^{L} \mathbb{1}_{\hat{y}_\ell^i=y_\ell^i}$,
- **Macro-F1** $\coloneqq \frac{1}{L}\sum_{\ell=1}^{L} \frac{2\sum_{i=1}^{N_T} \hat{y}_\ell^i \cdot y_\ell^i}{\sum_{i=1}^{N_T} \hat{y}_\ell^i + \sum_{i=1}^{N_T} y_\ell^i}$,
- **Micro-F1** $\coloneqq \frac{2\sum_{\ell=1}^{L}\sum_{i=1}^{N_T} \hat{y}_\ell^i \cdot y_\ell^i}{\sum_{\ell=1}^{L}\sum_{i=1}^{N_T} \hat{y}_\ell^i + \sum_{\ell=1}^{L}\sum_{i=1}^{N_T} y_\ell^i}$.

The above metrics can be cast into two categories, instance-based metrics (Exact-Match and Hamming-Score) and label-based metrics (Macro-F1 and Micro-F1) [38]. Among the metrics, Exact-Match is the most stringent measure, since it does not evaluate partial match of labels. In spite of that, it is a good metric to measure how well label correlations are modeled. Hamming-Score emphasizes on the prediction accuracy on label-instance pairs and is able to evaluate the performance on each single label. Macro-F1 and Micro-F1 take the partial match of label sets into account, providing good supplements for instance-based metrics. In addition, Macro-F1 is more sensitive to the performance of rare categories (the labels in minority), while Micro-F1 is affected more by the major categories (the labels in majority). Hence, joint use of Macro-F1 and Micro-F1 should be a good supplement for the instance-based evaluation metrics to evaluate the performances of MLC methods.

8.2. Configuration

In the experiments we compare the following seven MLC methods:

BR: Binary Relevance (BR) [2]. It transforms a multi-label problem with q labels into q binary classification problems according to the one-vs-all strategy.

ECC: Ensemble of CCs (ECC) [5]. A number of CCs are established by randomly selecting the chain orders, and the final prediction is made by majority votes.

EBCC: Ensemble of BCCs (EBCC) [7]. A number of BCCs are established by randomly selecting theirs roots, and its prediction is made by the same way ECC does.

MDDM: Multi-label Dimension reduction via Dependence Maximization (MDDM) [12]. It is a *global* FS-DR method. By maximizing the feature-label dependence, the projection is built to project the features into the low-dimensional subspace.

LLSF: Learning Label-Specific Features (LLSF) [16]. It is a *local* FS-DR method. Label-specific features are selected by optimizing the least squares problem with constraints of label correlation and feature sparsity.

oCC[1]: Optimized Classifier Chains (oCC). The proposed MLC method by combining model selection (kCC) and local feature selection (MLFS).

EoCC[1]: Ensemble of oCCs (EoCC). The same ensemble strategy of ECC is applied.

BR was used as the baseline MLC method. As the ensemble CC-based methods, ECC and EBCC were introduced due to their superior performances over some MLC decomposition methods as shown in [5, 7]. MDDM was chosen as a representative of global FS-DR methods, which outperformed several global FS-DR methods, like PCA, LPP [39], MLSI [22], CCA [23], as reported in [12]. As a local FS-DR method, LLSF was chosen due to its performance advantage in comparison with another local FS-DR method, LIFT [17], as reported in [16].

In the experiments, *5-fold cross validation* was performed to evaluate the classification performance. For fair comparison, a linear SVM implemented in Liblinear [40] was used as the baseline binary classifier for all the comparing methods. In parameter setting, we tune the important parameters on controlling the feature sparsity for the FS-DR methods, such as MDDM, LLSF and oCC/EoCC. Specifically, the dimensionality of feature space in MDDM (thr) and oCC/EoCC (M) is selected from $\{0.1, 0.3, 0.5, 0.7, 0.9\}$ of total number d of features, while the value of β in LLSF is tuned in range of $\{0.001, 0.01, 0.1, 1, 10\}$. As suggested in [12] and [16], we set the regularization parameter $\mu = 0.5$ in MDDM, and the parameters $\alpha = 0.5$ and $\gamma = 0.01$ in LLSF. To balance the performance of oCC in Exact-Match and Macro/Micro-F1, we set the parameters as $k = \lceil 0.8 \times q \rceil$. For the ensemble methods, we use an ensemble of 10 CCs/BCCs/oCCs for building ECC/EBCC/EoCC. In order to scale up the ensemble methods, random sampling is applied to randomly select 75% of instances and 50% of features for building each single model of the ensemble, as recommended in [5]. All the comparing methods were implemented in Matlab, and experiments were performed in a computer configured with an Intel Quad-Core i7-4770 CPU at 3.4GHz with 4GB RAM.

8.3. On k-dependence classifier chains

To evaluate the performance of model selection, an experiment was performed by kCC on four datasets, where we increased the number of parents k from 0 to $q - 1$ by step 1. For convenience, the values of each metric were normalized by dividing its maximum. Fig. 3 shows the experimental results in four metrics averaged by 5-fold cross validation. Consistent with the theoretical analysis, the performance of kCC in Exact-Match upgrades as the value of k increases except on the medical dataset. Moreover, as the value of k approaches

[1]We provide the codes of oCC and EoCC at: `https://github.com/futuresun912/oCC.git`

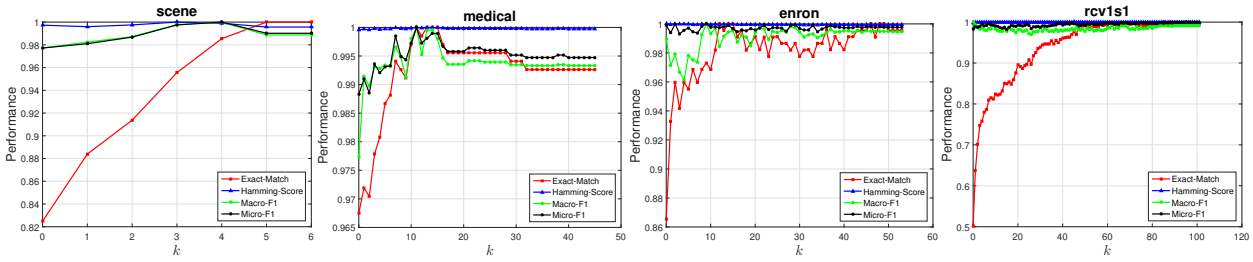


Figure 3: Performances of kCC on four different datasets in four evaluation metrics, whose values in each metric are normalized by dividing its maximum. The number k of parents is increased from 0 to $q-1$ by step 1. The indices of the maximum in Exact-Match over the four datasets are 5, 11, 12 and 88, respectively.

$0.8 \times q$, the performance becomes stable. In terms of Macro/Micro-F1, the performances share the similar tendency with Exact-Match, but the changes on values are relatively small. In Hamming-Score, its performance seems irrelevant to the change of k. Therefore, it is suggested to set the value of k as $\lceil 0.8 \times q \rceil$ if the objective is to optimize Exact-Match.

8.4. On multi-label feature selection

To evaluate the performance of multi-label feature selection, another experiment was performed by five information theoretic feature selection algorithms. The algorithms are generated according to (42) and (43) as Multi-Label Mutual Information Maximization (MLMIM) [15] ($\beta = \gamma = 0$ of (42)), Multi-Label Joint Mutual Information (MLJMI) [15] and Multi-Label Max-Relevance Min-Redundancy (MLMRMR) [33] ($\beta = 0,\ \gamma = 1/|\mathbf{X}_\theta|$ of (42), but MLMRMR ignores the conditional redundancy term.), $\text{MLFS}_{\text{local}}$ ($\beta = 1/|\mathbf{Pa}_j|,\ \gamma = 1/|\mathbf{X}_{\theta_j}|$ of (43)), and $\text{MLFS}_{\text{global}}$ ($\beta = 1/|\mathbf{Pa}_j|,\ \gamma = 1/|\mathbf{X}_\theta|$ of (42)). Fig. 4 shows the experimental results on four different datasets in Exact-Match and Macro-F1. We increased the number M of selected features from 10% to 100% by step 10% of $\min\{d, 200\}$. The local algorithm, $\text{MLFS}_{\text{local}}$, outperformed the other ones on average, and its performance was more stable to the value change of M compared with other algorithms. Among the global algorithms, $\text{MLFS}_{\text{global}}$ worked better than MLMIM, MLMRMR and MLJMI. Note that the performance of $\text{MLFS}_{\text{local}}$ and $\text{MLFS}_{\text{global}}$ achieved significant advantage with a smaller value of M, probably because of the success on modeling label correlations in (42). Similar pattern can be observed in other metrics. In summary, performing local feature selection is more effective

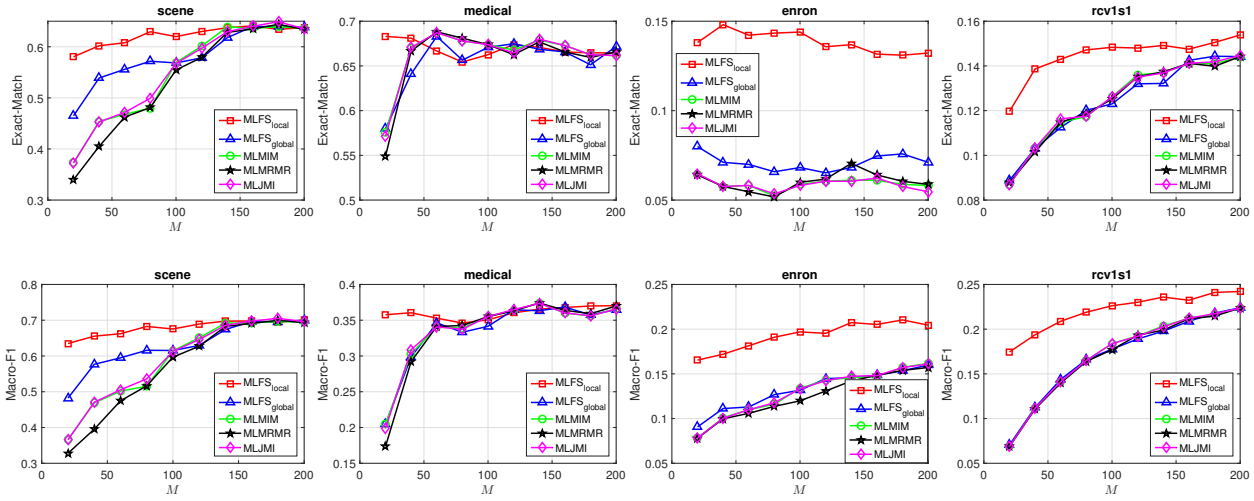


Figure 4: Comparing five information theoretic feature selection algorithms on four datasets in Exact-Match (the top row) and Macro-F1 (the bottom row). The number M of selected features is chosen from 10% to 100% by step 10% of $\min\{d, 200\}$.

than the global way, and it is important to take label correlations into account. Thus, we shall only employ $\text{MLFS}_{\text{local}}$ in the following experiments.

To evaluate the potential of oCC, we further conduct experiments by varying its two parameters: the number k of parents and the number M of selected features. The experimental results on the four datasets in Exact-Match and Macro-F1 are shown in Fig. 5. We select the values of k and M according to the percentages from 5% to 100% by step 5%. In terms of k, we can see that the performance becomes stable when a small percent of parents are selected. In terms of M, oCC achieved best performances when a fraction of features are selected. The results verify that the existence of useless parents and features.

8.5. On chain order selection

We performed another experiment to evaluate the performance of chain order selection developed in Algorithm 2. In this experiment, we incorporated Algorithm 2 with oCC (Algorithm 1), and set the two parameters of oCC as $k = q$ and $M = d$. To simplify the computation of Algorithm 2, we reduced the dimensionality of features to 1 by Principle Component Analysis (PCA) and ignored the second term of (47). We first compared oCC (with canonical label order $Y_1 \to Y_2 \to \cdots \to Y_{q-1} \to Y_q$), oCC_{cos} (with chain order selection) and EoCC_5 (ensemble of five oCCs with randomly selected orders). As shown in Fig. 6,

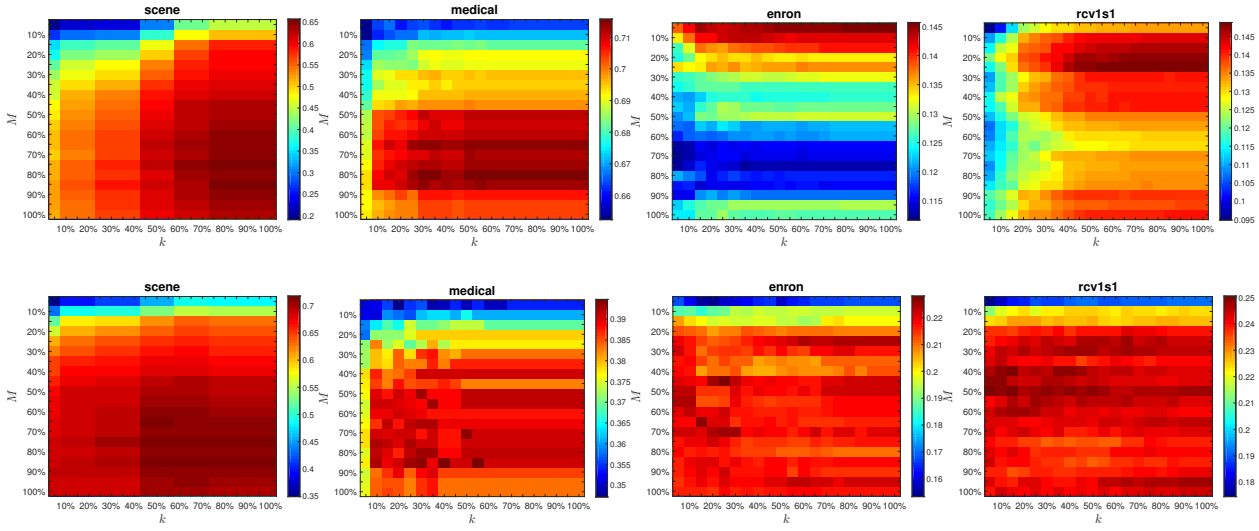


Figure 5: Comparison of oCC by varing the size k of parents and the number M of selected features on four different datasets in Exact-Match (the top row) and Macro-F1 (the bottom row). The values of k and M are varied by the percentages from 5% to 100% by step 5%.

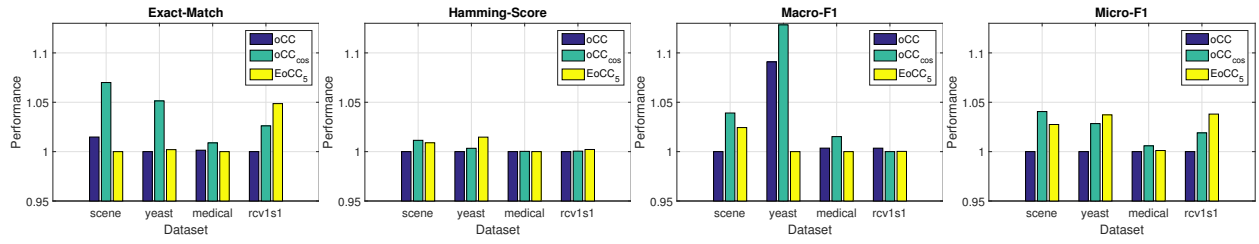


Figure 6: Comparison of oCC (with the canonical label order), oCC_{cos} and EoCC_5 on four datasets in four metrics. The values in each metric are normalized by dividing its minimum.

comparing to the canonical order, order selection works for improving the performance of oCC to some extent, although the extent is comparable to that of ensemble of different orders in many cases.

Next, oCC_{cos} was compared with EoCC_5 and EoCC_{10}, whose subscript denotes the ensemble number of oCCs with randomly selected chain orders. The number of training instances is varied from 10% to 100% by step 10% of the total number of instances. Fig. 7 shows the experimental results on four datasets in Exact-Match and Macro-F1. As shown in Fig. 7, although EoCC_5 and EoCC_{10} outperform oCC_{cos} on rcv1s1 in Exact-Match, oCC_{cos} works significantly better than EoCC_5 on scene and yeast, and is competitive with both EoCC_5 and EoCC_{10} on medical. In terms of processing time, EoCC cost several times more time

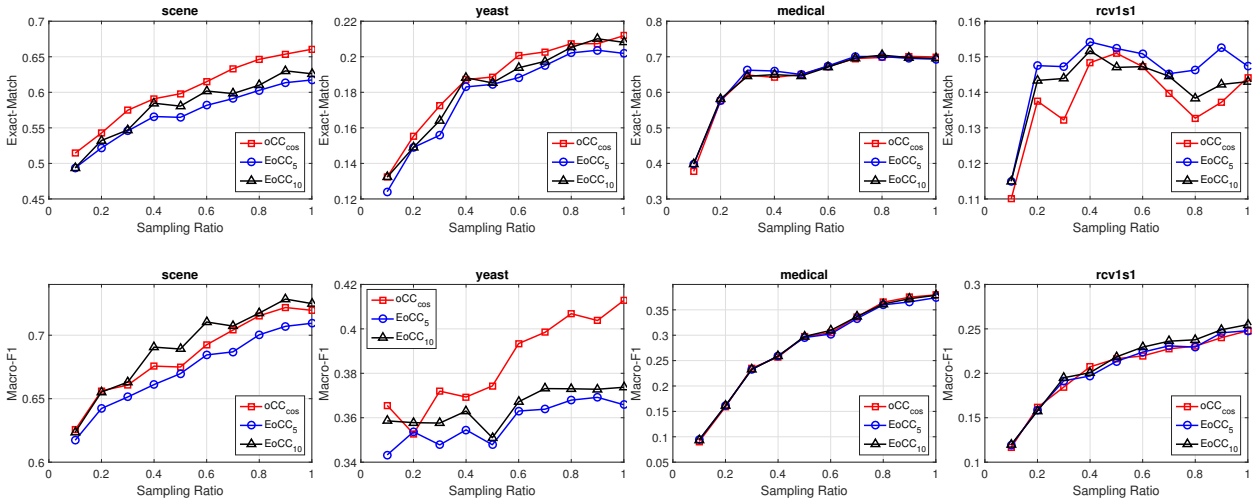


Figure 7: Comparison of oCC_{cos}, EoCC_5 and EoCC_{10} on four datasets in Exact-Match (the top row) and Macro-F1 (the bottom row). The sampling ratio on training instances is increased from 10% to 100% by step 10%.

than the compared oCC_{cos} in all cases. Therefore, in the following of this paper, we include this order selection procedure (Algorithm 2) in oCC otherwise stated elsewhere.

8.6. Comparison with the state-of-the-art

Tables 3 to 6 report the experimental results of seven comparing MLC methods over the thirteen benchmark multi-label datasets. For each evaluation metric, the larger the value, the better the performance. Among seven comparing methods, the best performance is highlighted in boldface. The average rank of each method over the datasets is reported in the last row of each Table.

The proposed oCC outperformed the other methods in terms of Exact-Match, Macro-F1 and Micro-F1 on average. It demonstrates the necessity of removing useless parents and features in classifier chains, indicating the effectiveness of the proposed framework on handling the MLC problems. On the other hand, benefiting from the ensemble strategy and oCC, EoCC ranked 1st in Hamming-Score, and worked the best among the three ensemble methods in four metrics. For the other ensemble methods, ECC achieved competitive results in all metrics except Macro-F1, and performed consistently better than EBCC, verifying the conclusion we reached at (23). Such observation is consistent with our theoretical analysis in Section 4.1, which shows BCC would worked worse than CC in modeling label correlations

Table 3: Experimental results (mean±std rank) of seven methods on thirteen multi-label datasets in terms of **Exact-Match**.

Dataset	BR	ECC	EBCC	MDDM	LLSF	oCC	EoCC
emotions	.233±.019 5.5	.270±.024 2.5	.238±.001 4	.206±.002 7	.233±.017 5.5	.270±.023 2.5	**.290±.011 1**
scene	.516±.013 6	.650±.021 3	.572±.013 4	.511±.010 7	.524±.010 5	**.669±.011 1**	.655±.007 2
yeast	.148±.009 6	.201±.014 3	.179±.011 4	.147±.008 7	.152±.010 5	**.210±.010 1**	.204±.009 2
birds	.474±.019 5.5	.482±.022 4	.489±.001 2	.464±.006 8	.474±.032 5.5	**.496±.013 1**	.487±.015 3
genbase	.982±.005 2.5	.973±.001 6	.973±.001 6	.977±.015 4	.982±.001 2.5	**.985±.005 1**	.973±.004 6
medical	.670±.015 4	.671±.018 3	.660±.001 6	.601±.007 7	.663±.011 5	**.694±.012 1**	.683±.014 2
enron	.113±.008 6	**.144±.006 1**	.130±.001 4	.112±.004 7	.114±.006 5	.140±.010 3	.143±.007 2
languagelog	.158±.002 6	.162±.001 2.5	**.163±.001 1**	.158±.004 6	.158±.001 6	.161±.005 4	.162±.003 2.5
rcv1s1	.073±.004 6	.138±.004 2	.105±.001 4	.064±.002 7	.074±.003 5	**.148±.004 1**	.137±.005 3
bibtex	.149±.003 5.5	.172±.001 2.5	.172±.001 2.5	.133±.003 7	.149±.001 5.5	.163±.004 4	**.176±.003 1**
corel16k1	.008±.001 5	.015±.002 2.5	.010±.001 4	.007±.001 6.5	.007±.001 6.5	**.023±.001 1**	.015±.002 2.5
corel5k	.006±.001 6	.013±.001 2	.009±.001 4	.005±.002 7	.007±.001 5	**.020±.003 1**	.012±.002 3
delicious	.002±.001 6	**.005±.001 1**	.003±.001 4	.002±.001 6	.002±.001 6	.004±.002 2.5	.004±.001 2.5
Rank	5.231 5.5	2.731 3	3.808 4	6.692 7	5.231 5.5	**1.846 1**	2.462 2

Table 4: Experimental results (mean±std rank) of seven methods on thirteen multi-label datasets in terms of **Hamming-Score**.

Dataset	BR	ECC	EBCC	MDDM	LLSF	oCC	EoCC
emotions	.785±.006 2.5	.778±.007 4.5	.778±.001 4.5	.776±.006 6	.785±.010 2.5	.767±.006 7	**.787±.006 1**
scene	.890±.003 7	.907±.005 2	.898±.001 4	.892±.002 5.5	.892±.002 5.5	.900±.004 3	**.909±.001 1**
yeast	.798±.004 2	.794±.005 5.5	.795±.001 4	.796±.005 3	**.799±.002 1**	.787±.005 7	.794±.005 5.5
birds	.948±.002 5.5	.950±.004 3.5	.950±.001 3.5	.946±.009 7	.948±.002 5.5	**.951±.002 1.5**	**.951±.002 1.5**
genbase	**.999±.000 4**	**.999±.001 4**	**.999±.001 4**	**.999±.001 4**	**.999±.001 4**	**.999±.000 4**	**.999±.000 4**
medical	**.990±.000 3.5**	**.990±.000 3.5**	**.990±.001 3.5**	.987±.001 7	**.990±.001 3.5**	**.990±.000 3.5**	**.990±.000 3.5**
enron	.942±.001 5	.950±.001 2	.940±.001 7	.948±.000 3	.941±.001 6	.946±.001 4	**.951±.001 1**
languagelog	.797±.000 6.5	.830±.001 2.5	.828±.001 4	.822±.000 5	.797±.000 6.5	**.836±.000 1**	.830±.000 2.5
rcv1s1	.966±.000 6.5	**.973±.001 1.5**	.972±.001 3	.971±.000 4	.966±.000 6.5	.967±.000 5	**.973±.000 1.5**
bibtex	.985±.000 5	**.988±.001 2**	**.988±.001 2**	.984±.000 7	.985±.001 5	.985±.000 5	**.988±.000 2**
corel16k1	.980±.000 5.5	**.981±.000 2.5**	**.981±.001 2.5**	**.981±.000 2.5**	.980±.000 5.5	.977±.000 7	**.981±.000 2.5**
corel5k	.988±.000 5	**.990±.001 2**	**.990±.001 2**	.987±.000 7	.988±.001 5	.988±.000 5	**.990±.000 2**
delicious	.981±.000 4.5	**.982±.000 2**	**.982±.001 2**	.980±.000 6	.981±.000 4.5	.979±.000 7	**.982±.000 2**
Rank	4.885 6	3.269 2	3.462 3	5.308 7	4.654 5	4.231 4	**2.192 1**

Table 5: Experimental results (mean±std rank) of seven methods on thirteen multi-label datasets in terms of **Macro-F1**.

Dataset	BR	ECC	EBCC	MDDM	LLSF	oCC	EoCC
emotions	.545±.015 5.5	.588±.006 2	.570±.001 4	.543±.009 7	.545±.015 5.5	.580±.012 3	**.613±.013 1**
scene	.684±.007 6	.742±.011 2	.712±.001 4	.673±.005 7	.688±.006 5	.724±.012 3	**.746±.004 1**
yeast	.355±.006 4.5	.359±.003 2.5	.353±.001 6	.355±.008 4.5	.351±.003 7	**.410±.007 1**	.359±.006 2.5
birds	.135±.018 3.5	.125±.006 7	.131±.001 5	.139±.002 2	.135±.014 3.5	**.177±.017 1**	.127±.013 6
genbase	**.769±.016 1.5**	.725±.001 6.5	.745±.001 5	.765±.012 3	**.769±.001 1.5**	.758±.016 4	725±.015 6.5
medical	.374±.007 4	.328±.013 6	.323±.001 7	.383±.007 2	.378±.008 3	**.387±.006 1**	.330±.007 5
enron	**.220±.010 1**	.192±.010 6.5	.197±.001 4.5	.218±.003 2.5	.218±.004 2.5	.194±.007 6.5	.197±.003 4.5
languagelog	.391±.003 2	.387±.008 6	.389±.001 4.5	.389±.006 4.5	.390±.005 3	.365±.004 7	**.398±.004 1**
rcv1s1	**.246±.007 1.5**	.211±.008 6	.219±.001 5	.234±.002 4	**.246±.005 1.5**	.235±.009 3	.209±.007 7
bibtex	**.329±.001 1.5**	.248±.001 7	.254±.001 5	.308±.003 4	**.329±.001 1.5**	.328±.003 3	.250±.002 6
corel16k1	.060±.002 2	.048±.003 4	.033±.001 7	.041±.001 6	.059±.003 3	**.064±.001 1**	.045±.001 5
corel5k	.046±.001 2.5	.031±.001 5.5	.031±.001 5.5	.043±.002 4	**.047±.001 1**	.046±.001 2.5	.030±.001 7
delicious	.123±.002 3	.102±.003 5	.098±.001 6	**.130±.001 1**	.122±.003 4	.124±.005 2	.096±.001 7
Rank	3.039 2	5.039 6	5.308 7	4.000 4	3.115 3	**2.923 1**	3.039 5

Table 6: Experimental results (mean±std rank) of seven methods on thirteen multi-label datasets in terms of **Micro-F1**.

Dataset	BR	ECC	EBCC	MDDM	LLSF	oCC	EoCC
emotions	.591±.016 5.5	.620±.005 2	.608±.001 3	.578±.010 7	.591±.016 5.5	.606±.011 4	**.647±.010 1**
scene	.677±.010 6	.734±.010 2	.704±.001 4	.668±.008 7	.682±.006 5	.715±.010 3	**.738±.003 1**
yeast	.633±.008 5	.642±.005 2	.637±.001 3	.631±.010 6.5	.631±.004 6.5	.636±.009 4	**.645±.009 1**
birds	.266±.026 6	.286±.011 2	.279±.001 3	.266±.001 6	.266±.011 6	**.300±.025 1**	.277±.024 4
genbase	**.993±.002 2**	.989±.001 5.5	.988±.001 7	.991±.007 4	**.993±.001 2**	**.993±.002 2**	.989±.002 5.5
medical	.809±.007 4	.804±.007 5	.799±.001 6	.764±.004 7	**.815±.011 1**	.811±.006 3	.812±.005 2
enron	.521±.005 6	.566±.008 3	.569±.001 2	.553±.003 4	.517±.003 7	.532±.006 5	**.573±.004 1**
languagelog	.520±.006 6.5	.573±.004 3	.570±.001 4	.553±.002 5	.520±.003 6.5	**.579±.006 1**	.577±.005 2
rcv1s1	.401±.003 4.5	.429±.004 2	.427±.001 3	.392±.004 7	.401±.003 4.5	.394±.003 6	**.432±.004 1**
bibtex	.429±.002 2.5	.416±.001 6	.419±.001 4	.412±.002 7	.429±.001 2.5	**.430±.002 1**	.418±.003 5
corel16k1	.106±.002 4	.137±.005 2.5	.088±.001 6	.078±.002 7	.105±.002 5	**.167±.007 1**	.137±.003 2.5
corel5k	.167±.003 4	.169±.001 2	.151±.001 6	.131±.003 7	.167±.001 4	**.191±.005 1**	.167±.003 4
delicious	**.257±.001 1.5**	.234±.005 4	.231±.001 6	.250±.002 3	**.257±.002 1.5**	.233±.004 5	.223±.003 7
Rank	4.423 6	3.115 3	4.385 5	6.077 7	4.346 4	**2.769 1**	2.885 2

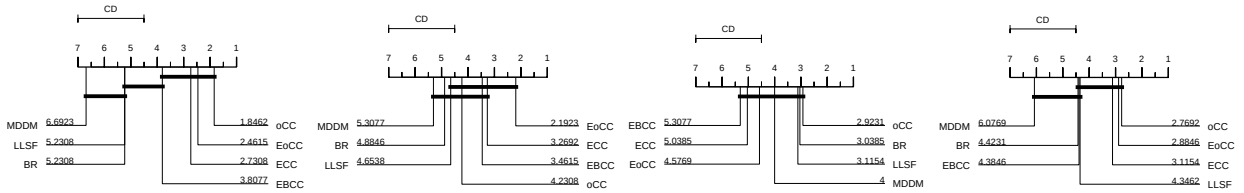
Table 7: Results of the Friedman Statistics F_F (7 methods, 13 datasets) and the Critical Value (0.05 significance level). The null hypothesis as the equal performance is rejected, if the values of F_F in terms of all metrics are higher than the Critical Value.

Friedman test	Exact-Match	Hamming-Score	Macro-F1	Micro-F1
F_F	25.017	3.967	3.273	5.052
Critical Value		2.230		

due to its limitation of $|\mathbf{Pa}| \leq 1$. For the FS-DR methods, the local method LLSF performed consistently better than the global method MDDM in all metrics. It is probably because that selection of global feature subset would loss some label-specific discriminative information. As the baseline method, BR is competitive with the best methods only in Macro-F1. The worse performance of BR probably results from its simple decomposition strategy which ignores label correlations.

8.7. Results on statistical test

To perform comparative analysis on experimental results in Tables 3 to 6 by statistical test, we utilized the evaluation methodology for MLC used in [41, 38], i.e., *Friedman test* with a post-hoc *Nemenyi test*. We fist conducted Friedman test [42] (7 methods, 13 datasets) with significance level 0.05, aiming to reject the null-hypothesis as equal performance among the comparing methods. The results are shown in Table 7. Since the values of the Friedman Statistic F_F in terms of all metrics are higher than the Critical Value, the null hypothesis as the equal performance was rejected. Therefore, a post-hoc test, Nemenyi test, is subsequently conducted to evaluate the performance between every two methods. According to [42], the performance of two methods is regarded as significantly different if their average ranks differ by at least the *critical difference* (CD). Figure 8 shows the CD diagrams for four evaluation metrics at 0.05 significance level. In each subfigure, the CD is given above the axis, where the averaged rank is marked. In Figure 8, algorithms which are not significantly different are connected by a thick line. In terms of Exact-Match, among 91 comparisons (7 methods $\times$ 13 datasets), oCC/EoCC outperformed other methods, and achieved statistically superior performance than the other methods except ECC/EBCC, consistent with the theoretical analysis in Section 4.1. Moreover, oCC/EoCC worked better than ECC on average, showing the effectiveness of the proposed framework on optimizing CC. In Hamming-Score, oCC



(a) Exact-Match (b) Hamming-Score (c) Macro-F1 (d) Micro-F1

Figure 8: CD diagrams (0.05 significance level) of seven comparing methods in four evaluation metrics. The performance of two methods is regarded as significantly different if their average ranks differ by at least the Critical Difference.

ranked fourth, following the three ensemble methods. It is because the objective of oCC is to approximate the optimizer in Exact-Match, which probably brings in the performance loss in Hamming-Score, as shown in [8]. In Macro-F1 and Micro-F1, oCC ranked first, while EoCC performed better than ECC and EBCC. In summary, oCC and EoCC achieved competitive performances in terms of four metrics. If our objective is to learn an optimizer in Exact-Match, oCC or EoCC should be the first choice.

9. Conclusion and future work

In this paper, we have reconsidered multi-label classification problems from the viewpoint of conditional likelihood maximization. Based on the proposed Optimization Strategy, a unified MLC framework was proposed with three optimization sub-problems: model selection, feature selection and parameter selection. We focused on addressing the former two problems by developing k-dependence Classifier Chains (kCC) and Multi-Label Feature Selection (MLFS), respectively. In addition, the generality of the proposed framework was demonstrated by mining the relationship with previous algorithms on CC-based methods and information theoretic feature selection.

Under a few acceptable assumptions on the label and feature space, a novel multi-label method, optimized Classifier Chains (oCC) was proposed by combining kCC and MLFS. Extensive experiments on benchmark multi-label datasets verified our assumption and demonstrated the effectiveness of the proposed framework. For the future work, it is interesting to develop the framework via likelihood maximization for other MLC decomposition methods.

Acknowledgments

This work was partially supported by JSPS KAKENHI Grant Numbers 15H02719.

[1] G. Tsoumakas, I. Katakis, Multi-label classification: An overview, International Journal of Data Warehousing and Mining 3 (2007) 1–13.

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

[3] J. Fürnkranz, E. Hüllermeier, E. Mencia, K. Brinker, Multilabel classification via calibrated label ranking, Machine Learning 73 (2) (2008) 133–153.

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

[5] J. Read, B. Pfahringer, G. Holmes, E. Frank, Classifier chains for multi-label classification, Machine Learning 85 (3) (2011) 333–359.

[6] K. Dembczynski, W. Cheng, E. Hüllermeier, Bayes optimal multilabel classification via probabilistic classifier chains, in: Proceedings of the 27th International Conference on Machine Learning, 2010, pp. 279–286.

[7] J. Zaragoza, L. Sucar, E. Morales, C. Bielza, P. L. naga, Bayesian chain classifiers for multidimensional classification, in: Proceedings of the 22nd International Joint Conference on Artificial Intelligence, 2011, pp. 2192–2197.

[8] K. Dembczynski, W. Waegeman, W. Cheng, E. Hllermeier, On label dependence and loss minimization in multi-label classification, Machine Learning 88 (1-2) (2012) 5–45.

[9] K. Dembczynski, A. Jachnik, W. Kotlowski, W. Waegeman, E. Hüllermeier, Optimizing the f-measure in multi-label classification: Plug-in rule approach versus structured loss minimization., ICML (3) 28 (2013) 1130–1138.

[10] H.-F. Yu, P. Jain, P. Kar, I. S. Dhillon, Large-scale multi-label learning with missing labels., in: ICML, 2014, pp. 593–601.

[11] K. Bhatia, H. Jain, P. Kar, M. Varma, P. Jain, Sparse local embeddings for extreme multi-label classification, in: Advances in NIPS 28, 2015, pp. 730–738.

[12] Y. Zhang, Z.-H. Zhou, Multilabel dimensionality reduction via dependence maximization, ACM Transactions on Knowledge Discovery from Data 4 (3) (2010) 14:1–14:21.

[13] Y. Chen, H. Lin, Feature-aware label space dimension reduction for multi-label classification, in: Advances in NIPS 25, 2012, pp. 1529–1537.

[14] Z. Lin, G. Ding, M. Hu, J. Wang, Multi-label classification via feature-aware implicit label space encoding., in: ICML, 2014, pp. 325–333.

[15] K. Sechidis, N. Nikolaou, G. Brown, Information theoretic feature selection in multi-label data through composite likelihood, Vol. 8621, 2014, pp. 143–152.

[16] J. Huang, G. Li, Q. Huang, X. Wu, Learning label specific features for multi-label classification, in: 2015 IEEE International Conference on Data Mining, 2015, 2015, pp. 181–190.

[17] M. Zhang, L. Wu, Lift: multi-label learning with label-specific features, IEEE Transactions on Pattern Analysis and Machine Intelligence 37 (1) (2015) 107–120.

[18] A. Kumar, S. Vembu, A. K. Menon, C. Elkan, Learning and inference in probabilistic classifier chains with beam search, in: Proceedings of the 2012 European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part I, ECML PKDD'12, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 665–680.

[19] J. Read, L. Martino, D. Luengo, Efficient monte carlo methods for multi-dimensional learning with classifier chains, Pattern Recognition 47 (3) (2014) 1535 – 1546, handwriting Recognition and other {PR} Applications.

[20] W. Liu, I. Tsang, On the optimality of classifier chain for multi-label classification, in: C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, R. Garnett (Eds.), Advances in Neural Information Processing Systems 28, Curran Associates, Inc., 2015, pp. 712–720.

[21] J. Read, L. Martino, P. M. Olmos, D. Luengo, Scalable multi-output label prediction: From classifier chains to classifier trellises, Pattern Recognition 48 (6) (2015) 2096–2109.

[22] K. Yu, S. Yu, V. Tresp, Multi-label informed latent semantic indexing, in: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '05, 2005, pp. 258–265.

[23] L. Sun, S. Ji, J. Ye, Canonical correlation analysis for multilabel classification: A least-squares formulation, extensions, and analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (1) (2011) 194–200.

[24] H. Wang, C. Ding, H. Huang, Multi-label linear discriminant analysis, in: Proceedings of the 11th European Conference on Computer Vision, Vol. 6316, 2010, pp. 126–139.

[25] G. Doquire, M. Verleysen, Mutual information-based feature selection for multilabel classification, Neurocomputing 122 (2013) 148 – 155, advances in cognitive and ubiquitous computingSelected papers from the Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2012).

[26] J. Lee, D.-W. Kim, Feature selection for multi-label classification using multivariate mutual information, Pattern Recognition Letters 34 (3) (2013) 349–357.

[27] G. Brown, A. Pocock, M.-J. Zhao, M. Luján, Conditional likelihood maximisation: A unifying framework for information theoretic feature selection, Journal of Machine Learning Research 13 (2012) 27–66.

[28] S. Kullback, R. A. Leibler, On information and sufficiency, The Annals of Mathematical Statistics 22 (1) (1951) 79–86.

[29] R. M. Fano, Transmission of information: statistical theory of communications., IEEE Transactions on Information Theory.

[30] L. Sun, M. Kudo, Polytree-augmented classifier chains for multi-label classification, in: Proceedings of the 24th International Joint Conference on Artificial Intelligence, 2015, pp. 3834–3840.

[31] K. Dembczyński, W. Waegeman, E. Hüllermeier, An analysis of chaining in multi-label classification, in: Proceedings of the 2012 European Conference on Artificial Intelligence, Vol. 242, IOS Press, 2012, pp. 294–299.

[32] N. Friedman, D. Geiger, M. Goldszmidt, Bayesian network classifiers, Machine Learning 29 (2-3) (1997) 131–163.

[33] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (8) (2005) 1226–1238.

[34] H. H. Yang, J. Moody, Data visualization and feature selection: New algorithms for nongaussian data, in: in Advances in Neural Information Processing Systems, MIT Press, 1999, pp. 687–693.

[35] D. D. Lewis, Feature selection and feature extraction for text categorization, in: Proceedings of the Workshop on Speech and Natural Language, Association for Computational Linguistics, Stroudsburg, PA, USA, 1992, pp. 212–217.

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

[37] J. Read, P. Reutemann, B. Pfahringer, G. Holmes, MEKA: A multi-label/multi-target extension to Weka, Journal of Machine Learning Research 17 (21) (2016) 1–5.
URL http://jmlr.org/papers/v17/12-164.html

[38] G. Madjarov, D. Kocev, D. Gjorgjevikj, S. Deroski, An extensive experimental comparison of methods for multi-label learning, Pattern Recognition 45 (9) (2012) 3084–3104.

[39] X. He, P. Niyogi, Locality preserving projections, in: In Advances in Neural Information Processing Systems 16, MIT Press, 2003.

[40] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, C.-J. Lin, LIBLINEAR: A library for large linear classification, Journal of Machine Learning Research 9 (2008) 1871–1874.

[41] W. Cheng, E. Hüllermeier, Combining instance-based learning and logistic regression for multilabel classification, Machine Learning 76 (2-3) (2009) 211–225.

[42] J. Demšar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.