Multi-Label Learning with Regularization Enriched Label-Specific Features

Ze-Sen Chen
Min-Ling Zhang

School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, China

Editors: Wee Sun Lee and Taiji Suzuki

Abstract

Multi-label learning learns from examples each associated with multiple class labels simultaneously, and the goal is to induce a predictive model which can assign a set of relevant labels for the unseen instance. Label-specific features serve as an effective strategy towards inducing multi-label predictive model, where the relevancy of each class label is determined by employing tailored features encoding inherent and distinct characteristics of the class label its own. In this paper, a regularization based approach named REEL is proposed for label-specific features generation, which works by enriching label-specific feature representation for each class label via synergizing informative label-specific features from other class labels with sparse regularization. Specifically, full-order label correlations are considered by REEL while the number of classifiers induced for multi-label prediction is linear to the number of class labels. Extensive experiments on fifteen benchmark multi-label data sets clearly show the favorable performance of REEL against other state-of-the-art multi-label learning approaches with label-specific features.

Keywords: Multi-label, Label-specific features, Sparse regularization

1. Introduction

Multi-label learning deals with objects with rich semantics where each example is represented by a single instance (feature vector) while associated with multiple class labels simultaneously (Zhang and Zhou, 2014; Gibaja and Ventura, 2015). In recent years, the need to learn from multi-label examples are widely witnessed in various applications such as text classification (Rubin et al., 2012), image annotation (Sun et al., 2014), social network analysis (Wang and Sukthankar, 2013), music emotion categorization (Trohidis et al., 2011; Wu et al., 2014), bioinformatics (Pan et al., 2019), etc.

To learn from multi-label examples, one common strategy is to build predictive model based on the single instance representation to discriminate all the class labels (Zhang and Zhou, 2014; Gibaja and Ventura, 2015). Nonetheless, the inherent and distinct characteristics of each class label is not fully considered by employing identical feature representation for model induction. For instance, in text categorization, features corresponding to word terms election, debate and voting are informative in discriminating political and non-political documents, while features corresponding to word terms Olympics, championship and referee are informative in discriminating sports and non-sports documents. Therefore, the strategy of label-specific features has been proposed to facilitating
multi-label learning which aims to determine the relevancy of each class label with tailored features of its own.

Existing attempts towards label-specific features work by generating tailored features for each class label independently (Zhang and Wu, 2015; Xu et al., 2016), considering pairwise label correlations (Huang et al., 2018; Weng et al., 2018), or selecting a subset of features from the original feature space (Huang et al., 2016; Sun et al., 2016; Zhang et al., 2018b). To generate label-specific features with strong discriminative abilities for inducing multi-label predictive model, it is desirable that the correlations among class labels can be fully exploited. On the other hand, the computational cost for generating label-specific features should be manageable especially in terms of the number of class labels in label space.

In this paper, a novel approach named REEL, i.e. REgularization Enriched Label-specific features, is proposed to learning from multi-label examples. Briefly, REEL employs a two-stage procedure for label-specific features generation. In the first stage, a number of base label-specific features are constructed by conducting clustering analysis on the positive and negative instances of each class label (Zhang and Wu, 2015; Xu et al., 2016). After that, REEL enriches the label-specific feature representation for each class label via synergizing informative base label-specific features from other class labels with sparse regularization. Thereafter, a set of binary classification models are induced based on the enriched label-specific features to determine the relevancy of each class label for unseen instance. Comprehensive experiments across a total of fifteen benchmark data sets validate the superior performance of REEL against state-of-the-art multi-label learning approaches based on label-specific features.

The rest of this paper is organized as follows. In Section 2, related works on label-specific features are briefly reviewed. In Section 3, technical details of the proposed approach based on label-specific features are presented. In Section 4, experimental results of comparative studies are reported. Finally, Section 5 concludes.

2. Related Work

The task of multi-label learning has been extensively studied in recent years, where significant number of approaches have been proposed to learning from examples with multiple class labels simultaneously (Zhang and Zhou, 2014; Gibaja and Ventura, 2015). Due to the combinatorial nature of the label set to be predicted, most works focus on exploiting correlations among class labels to help induce the multi-label predictive model. Generally speaking, the order of label correlations considered by the learning algorithm can be first-order where the multi-label learning problem is tackled by treating each class label independently (Boutell et al., 2004; Zhang et al., 2018c), second-order where the multi-label learning problem is tackled by modeling pairwise interactions between class labels (Fünkranz et al., 2008; Brinker et al., 2014), or high-order where the multi-label learning problem is tackled by considering high-order interactions among a subset of or all class labels (Read et al., 2011; Tsoumakas et al., 2011).

In addition to the exploitation of label correlations, the generalization performance of multi-label learning system can be improved by manipulating the feature space. The most straightforward feature manipulation strategy is to conduct dimensionality reduction (Sun et al., 2013) or feature selection (Pereira et al., 2018) over the original feature space. Other feature manipulation strategies include generating meta-level features by extracting refined discriminative information from the original features (Yang and Gopal, 2012; Canuto et al., 2016) and making use of multi-view
representation for multi-label data (Zhu et al., 2016; Zhan and Zhang, 2017; Zhang et al., 2018a). Nonetheless, these feature manipulation strategies share the same mechanism of using identical feature representation in the discrimination processes of all class labels.

In contrast to the above strategies for multi-label feature manipulation, the strategy of label-specific features works by utilizing tailored feature representation for each class label to better characterize its inherent and distinct properties. For label-specific features generation, one can conduct clustering analysis (Zhang and Wu, 2015) or attribute reduction (Xu et al., 2016) on the positive and negative instances of each class label independently. To consider label correlations in the generation procedure of label-specific features, it is not difficult to incorporate pairwise label correlations with similarity measure between class labels (Huang et al., 2018) or nearest neighbor rule (Weng et al., 2018). Rather than representing the label-specific features in the transformed feature space, it is also feasible to work with the original feature space by retaining a different subset of original features for each class label (Huang et al., 2016; Zhang et al., 2018b) or a group of class labels (Sun et al., 2016).

3. The REEL Approach

Let $\mathcal{X} = \mathbb{R}^d$ be the $d$-dimensional feature space and $\mathcal{Y} = \{l_1, l_2, \ldots, l_q\}$ be the label space consisting of $q$ possible class labels. Given the multi-label training set $\mathcal{D} = \{(\mathbf{x}_i, Y_i) \mid 1 \leq i \leq m\}$, where $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{id}]^\top$ is a $d$-dimensional feature vector and $Y_i \subseteq \mathcal{Y}$ is the set of relevant labels associated with $\mathbf{x}_i$. The task of multi-label learning is to induce a multi-label predictor $h : \mathcal{X} \mapsto 2^\mathcal{Y}$ which maps from the feature space to the powerset of label space.

Based on the strategy of label-specific features, REEL works by generating a feature mapping $\psi_k : \mathcal{X} \mapsto \mathcal{Z}_k$ for each class label $l_k$ ($1 \leq k \leq q$). Thereafter, a binary classification model $g_k : \mathcal{Z}_k \mapsto [0, 1]$ is induced based on the mapped feature space $\mathcal{Z}_k$. Accordingly, for the unseen instance $\mathbf{x} \in \mathcal{X}$, its label set can be predicted as $h(\mathbf{x}) = \{l_k \mid g_k(\psi_k(\mathbf{x})) \geq 0.5, 1 \leq k \leq q\}$. Specifically, REEL employs a two-stage procedure to fulfill the task of label-specific features generation.

For each class label $l_k \in \mathcal{Y}$, we can divide the instances into positive and negative set, following the clustering mechanism to analyze structural information of input space (Zhang and Wu, 2015; Xu et al., 2016), REEL adopts $k$-means algorithm to partition both set into $N_k$ clusters with:

$$N_k = \begin{cases} |l_k \in Y_k|, & \text{if } |l_k \in Y_k| < |l_k \notin Y_k| \\ |l_k \notin Y_k|, & \text{otherwise} \end{cases}$$  \hfill (1)

Here, the clustering centroids are denoted as $\{\mathbf{p}_{k1}^1, \mathbf{p}_{k1}^2, \ldots, \mathbf{p}_{N_k}^k\}$ and $\{\mathbf{n}_{k1}^1, \mathbf{n}_{k2}^2, \ldots, \mathbf{n}_{N_k}^k\}$ respectively.

Accordingly, the $2N_k$-dimensional base label-specific features $\phi_k : \mathcal{X} \mapsto \mathbb{R}^{2N_k}$ can be generated by querying the distance between the instance and clustering centroids:

$$\phi_k(\mathbf{x}) = [d(\mathbf{x}, \mathbf{p}_{k1}^1), \ldots, d(\mathbf{x}, \mathbf{p}_{N_k}^k), d(\mathbf{x}, \mathbf{n}_{k1}^1), \ldots, d(\mathbf{x}, \mathbf{n}_{N_k}^k)]^\top$$  \hfill (2)

Here, $d(\cdot, \cdot)$ returns the Euclidean distance between two feature vectors.

After having the base label-specific features in the first stage, REEL aims to enrich the label-specific features for each class label by exploiting its correlations with the other class labels. For each class label $l_k \in \mathcal{Y}$, let $\mathbf{y}_k = [y_{k1}, y_{k2}, \ldots, y_{km}]^\top$ denote the labeling vector for $l_k$ where
Table 1: The pseudo-code of REEL

Inputs:
\( \mathcal{D} \): multi-label training set \( \{(x_i, Y_i) \mid 1 \leq i \leq m\} \)
\( x_i \in \mathcal{X}, Y_i \subseteq \mathcal{Y}, \mathcal{X} = \mathbb{R}^d, \mathcal{Y} = \{l_1, l_2, \ldots, l_q\} \)
\( \lambda \): the regularization parameter (\( \lambda > 0 \))
\( \mathcal{L} \): the binary learning algorithm
\( x \): the unseen instance (\( x \in \mathcal{X} \))

Outputs:
\( Y \): predicted label set for \( x \)

Process:
1: for \( k = 1 \) to \( q \) do
2: Perform \( k \)-means algorithm on positive and negative set for \( l_k \) to obtain the clustering centroids \( \{p_{k1}, p_{k2}, \ldots, p_{NK_k}\} \) and \( \{n_{k1}, n_{k2}, \ldots, n_{NK_k}\} \) respectively, where \( N_k \) is set according to Eq.(1);
3: Form the base label-specific feature mapping \( \phi_k \) for \( l_k \) according to Eq.(2);
4: end for
5: for \( k = 1 \) to \( q \) do
6: Solve the \( L_1 \)-regularized optimization problem Eq.(4) (along with Eq.(3)) to obtain the weight vector \( \beta_k \);
7: Form the final label-specific feature mapping \( \psi_k \) according to Eq.(5) and Eq.(6);
8: end for
9: for \( k = 1 \) to \( q \) do
10: Generate the binary training set \( \mathcal{D}_k \) for \( l_k \) according to Eq.(7);
11: Induce the binary classifier \( g_k \) for \( l_k \) by invoking \( \mathcal{L} \) on \( \mathcal{D}_k \): \( g_k \leftarrow \mathcal{L}(\mathcal{D}_k) \);
12: end for
13: Obtain the prediction vector \( \eta \) for \( x \) by querying binary classifiers \( g_k \) (\( 1 \leq k \leq q \));
14: Update \( \eta \) to \( \hat{\eta} \) according to Eq.(8) and Eq.(9);
15: Return \( Y = \{l_k \mid \hat{\eta}_k \geq 0.5, 1 \leq k \leq q\} \).

y_{ki} = 1 \) if \( l_k \in Y_i \) and \( y_{ki} = 0 \) otherwise. Furthermore, let \( \varphi_k(x) \) denote the \( d_k \)-dimensional feature vector formed by concatenating all base label-specific features from all class labels other than \( l_k \):

\[
\varphi_k(x) = [\phi_1(x), \ldots, \phi_{k-1}(x), \phi_{k+1}(x), \ldots, \phi_q(x)]^\top
\]

Here, \( d_k = \sum_{k' \neq k} 2m_{k'} \). Let \( X_k = [\varphi_k(x_1), \varphi_k(x_2), \ldots, \varphi_k(x_m)]^\top \) be the \( m \times d_k \) matrix formed by applying \( \phi_k \) to all training examples, the correlation between \( l_k \) and the other class labels are modeled by solving the following optimization least-squares optimization problem with sparse regularization:

\[
\min_{\beta_k \in \mathbb{R}^{d_k}} \|y_k - X_k \cdot \beta_k\|_2^2 + \lambda \cdot \|\beta_k\|_1
\]

Here, \( \beta_k = [\beta_{k1}, \beta_{k2}, \ldots, \beta_{kd_k}]^\top \) is the weight vector to be optimized whose elements encode the importance of features in \( \varphi_k \) in predicting the relevancy of \( l_k \). Furthermore, \( \lambda > 0 \) corresponds to
The regularization parameter. After solving the above problem with off-the-shelf sparse regression techniques, REEL identifies base label-specific features from other class labels which can be used to enrich \( \phi_k \) by thresholding the learned weight vector \( \beta_k \). Specifically, let \( \mathcal{I}_k \) stores the indices of features in \( \varphi_k \) whose corresponding weight has magnitude greater than the threshold \( \gamma \):

\[
\mathcal{I}_k = \{ a \mid |\beta_{ka}| \geq \gamma, \ 1 \leq a \leq d_k \} \tag{5}
\]

Then, the final label-specific features \( \psi_k : \mathcal{X} \mapsto \mathcal{Z}_k \) for \( l_k \) are generated as:

\[
\psi_k(x) = [\phi_k(x), \Pi_{\mathcal{I}_k}(\varphi_k(x))] \tag{6}
\]

Here, \( \Pi_{\mathcal{I}}(u) \) represents the projection operation of retaining features in the index set \( \mathcal{I} \) for \( u \). Therefore, the dimensionality of \( \mathcal{Z}_k \) corresponds to \( 2N_k + |\mathcal{I}_k| \).

To train the predictive model, a binary training set \( \mathcal{D}_k \) for \( l_k \) can be transformed from the original multi-label training set \( \mathcal{D} \) as follows:

\[
\mathcal{D}_k = \{ (\psi_k(x_i), y_{ki}) \mid 1 \leq i \leq m \} \tag{7}
\]

Then, a binary classification model \( g_k : \mathcal{Z}_k \mapsto [0, 1] \) can be induced from \( \mathcal{D}_k \) by invoking some binary learning algorithm \( \mathcal{L} : g_k \leftarrow \mathcal{L}(\mathcal{D}_k) \). For unseen instance \( x \in \mathcal{X} \), let \( \eta = [\eta_1, \eta_2, \ldots, \eta_q]^\top = [g_1(\psi_1(x)), g_2(\psi_2(x)), \ldots, g_q(\psi_q(x))]^\top \) denote the prediction vector yielded by the induced binary classifiers. Instead of determining the relevancy of each class label for \( x \) based on \( \eta \), i.e. \( h(x) = \{ l_k \mid \eta_k \geq 0.5, \ 1 \leq k \leq q \} \), REEL improves \( \eta \) by further considering correlations among class labels.

Let \( \hat{y}_k = [\hat{y}_{k1}, \hat{y}_{k2}, \ldots, \hat{y}_{km}]^\top \) denote the signed labeling vector with \( \hat{y}_{ki} = +1 \) if \( l_k \in Y_i \) and \( \hat{y}_{ki} = -1 \) otherwise. Then, the correlation matrix \( \Theta = [\theta_{jk}]_{q \times q} \) is set according to the cosine similarity between the signed labeling vectors:

\[
\forall \ 1 \leq j, k \leq q : \ \theta_{jk} = \frac{\hat{y}_j^\top \cdot \hat{y}_k}{||\hat{y}_j||_2 \cdot ||\hat{y}_k||_2} \tag{8}
\]

Conceptually, \( \theta_{jk} \) takes the value within \([-1, 1]\) whose magnitude reflects the strength of correlation (either positive or negative) between \( l_j \) and \( l_k \). The correlation matrix \( \Theta \) is updated by thresholding \( \theta_{jk} \) into zero if its correlation strength is not strong, i.e. \( |\theta_{jk}| < \sigma \).

By utilizing the correlation matrix \( \Theta \), the prediction vector \( \eta \) is updated to \( \hat{\eta} \) with:

\[
\hat{\eta}_k = \begin{cases} 
\sum_{j=1}^{q} \theta_{jk} \cdot \eta_j, & \text{if } \left| \sum_{j=1}^{q} \theta_{jk} \cdot \eta_j \right| > \sigma \text{ or } \\
\eta_k, & \left| \sum_{j=1}^{q} \theta_{jk} \cdot \eta_j \right| < 1 - \sigma
\end{cases} \tag{9}
\]

Table 1 summarizes the complete procedure of REEL. In the first stage, the base label-specific features are generated by conducting clustering analysis on the positive and negative instances of each class label (Steps 1-4). In the second stage, the final label-specific features are generated

---

1. In this paper, the thresholding parameters \( \gamma \) (Eq.(5)) and \( \sigma \) are fixed to be 0.2 and 0.9 respectively. Furthermore, the ratio parameter \( \tau \) in Eq.(1) is fixed to be 0.1 (Zhang and Wu, 2015).
Table 2: Characteristics of the multi-label experimental data sets.

| Data set    | \( |S| \) | \( \text{dim}(S) \) | \( L(S) \) | \( F(S) \) | \( L\text{Card}(S) \) | \( L\text{Den}(S) \) | \( DL(S) \) | \( PDL(S) \) | Domain       |
|-------------|--------|----------------|--------|--------|----------------|----------------|--------|--------|-------------|
| Cal500      | 502    | 68             | 174    | numeric | 26.044        | 0.150          | 502    | 1.000  | audio       |
| emotions    | 593    | 72             | 6      | numeric | 1.868         | 0.311          | 27     | 0.046  | audio       |
| medical     | 978    | 1,449          | 45     | nominal | 1.245         | 0.028          | 94     | 0.096  | text        |
| enron       | 1,702  | 1,001          | 53     | nominal | 3.378         | 0.064          | 753    | 0.442  | text        |
| image       | 2,000  | 294            | 5      | numeric | 1.236         | 0.247          | 20     | 0.010  | image       |
| scene       | 2,407  | 294            | 5      | numeric | 1.074         | 0.179          | 15     | 0.006  | image       |
| yeast       | 2,417  | 103            | 14     | numeric | 4.237         | 0.303          | 198    | 0.082  | biology     |
| slashdot    | 3,782  | 1,079          | 22     | nominal | 1.181         | 0.054          | 156    | 0.041  | text        |
| rcv1-s1     | 6,000  | 500            | 101    | nominal | 2.880         | 0.029          | 1,028  | 0.171  | text        |
| rcv1-s2     | 6,000  | 500            | 101    | nominal | 2.634         | 0.026          | 954    | 0.159  | text        |
| rcv1-s3     | 6,000  | 500            | 101    | nominal | 2.614         | 0.026          | 939    | 0.156  | text        |
| rcv1-s4     | 6,000  | 500            | 101    | nominal | 2.484         | 0.025          | 816    | 0.136  | text        |
| bibtex      | 7395   | 1836           | 159    | nominal | 2.402         | 0.015          | 2856   | 0.386  | text        |
| corel16k (sample 1) | 13766 | 500           | 153    | nominal | 2.859         | 0.019          | 4803   | 0.349  | image       |
| corel16k (sample 2) | 13761 | 500           | 153    | nominal | 2.882         | 0.018          | 4868   | 0.354  | image       |

by enriching the base label-specific features of each class label via synergizing informative label-specific features from other class labels with sparse regularization (Steps 5-8). In this way, full-order label correlations have been considered via the synergy process of base label-specific features from all class labels. After that, a number of binary classifiers are generated based on the generated label-specific features (Steps 9-12) and then employed to make prediction on unseen instance (Steps 13-15).

4. Experiments

4.1. Experimental Setup

Table 2 summarizes detailed characteristics of the fifteen benchmark multi-label data sets employed for experimental studies. For each multi-label data set \( S \), we use \( |S| \), \( \text{dim}(S) \), \( L(S) \) and \( F(S) \) to represent the number of examples, number of features, number of class labels and feature type respectively. In addition, several multi-label statistics including label cardinality \( L\text{Card}(S) \), label density \( L\text{Den}(S) \), distinct label sets \( DL(S) \) and proportion of distinct label sets \( PDL(S) \) are also utilized to characterize properties of each data set. Detailed definitions on these multi-label statistics can be found in (Read et al., 2011; Zhan and Zhang, 2017).

As shown in Table 2, experimental data sets are roughly organized in ascending order of \( |S| \) with eight being regular-scale (first part, \( |S| < 5,000 \)) and seven being large-scale (second part, \( |S| \geq 5,000 \)). Specifically, the fifteen experimental data sets exhibit diversified multi-label properties which provide solid basis for thorough comparative studies.

To show the effectiveness of the proposed REEL approach, five benchmark multi-label learning approaches have been employed for comparative studies (with parameter configuration suggested in respective literatures):
MULTI-LABEL LEARNING WITH REGULARIZATION ENRICHED LABEL-SPECIFIC FEATURES

- **BR** (Boutell et al., 2004): A baseline approach which decomposes the multi-label learning problem into a number of independent binary learning problems, one per class label. BR employs the original feature representation for inducing all binary classifiers, which can be regarded as a degenerated version of label-specific features strategy. [Base learner: linear kernel SVM]

- **MDDM** (Zhang and Zhou, 2010): A multi-label dimensionality reduction approach which works by maximizing the dependence between original feature space and the associated class labels. \( thr = 99\% \)

- **LIFT** (Zhang and Wu, 2015): A first-order multi-label learning approach based on label-specific features strategy, which generates tailored features via conducting clustering analysis on each class label independently. [Base learner: linear kernel SVM, \( r = 0.1 \)]

- **LLSF** (Huang et al., 2016): A second-order multi-label learning approach based on label-specific features strategy, which generates tailored features by retaining a different subset of original features for each class label with feature-sharing between a pair of closely-related class labels. \( \alpha = 0.5, \beta = 0.1, \gamma = 0.01 \)

- **MLSF** (Sun et al., 2016): A high-order multi-label learning approach based on label-specific features strategy, which generates tailored features by retaining a different subset of original features for a group of class labels. \( K = \lceil q/10 \rceil, \epsilon = 0.01, \alpha = 0.8, \gamma = 0.01, \rho = 1 \)

For each comparing approach, parameter setup is stated above, which is suggested in respective literature. For REEL, as shown in Table 1, the regularization parameter \( \lambda \) is set to be 1 and Libsvm (Chang and Lin, 2011) is employed to instantiate the binary learning algorithm \( L \).

To evaluate the performance of each comparing approach, six widely-used multi-label evaluation metrics are used including hamming loss, ranking loss, one-error, coverage, average precision and micro-averaging AUC (Zhang and Zhou, 2014).\(^2\) For the first four metrics, the smaller the metric value the better the performance. For the other two metrics, the larger the metric value the better the performance. Ten-fold cross-validation is performed on each benchmark data set, where the mean metric value as well as standard deviation are recorded for performance evaluation.

### 4.2. Experimental Results

Tables 3 to 8 report the detailed experimental results of each comparing approach in terms of each evaluation metric respectively. Furthermore, the best performance among the comparing approaches is also shown in boldface. In this paper, Friedman test (Demšar, 2006) is employed here for statistical performance comparisons of multiple algorithms over a number of data sets. It is shown that at significance level \( \alpha = 0.05 \), the null hypothesis of equal performance among the comparing approaches is rejected in terms of each evaluation metric. Consequently, Bonferroni-Dunn test (Demšar, 2006) is employed as the post-hoc test to show the relative performance among comparing approaches by treating REEL as the control approach.

Figure 1 illustrates the CD diagrams where the average rank of each approach is marked along the axis with lower ranks to the right. Any approach whose average rank is within one critical

\(^2\) All evaluation metrics take value in \([0,1]\), where coverage is normalized by the number of class labels.
Table 3: The performance of each comparing approach (mean ± std. deviation) on each data set in terms of *hamming loss*. On each data set, the best performance among the comparing algorithms is shown in bold face.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>REEL</th>
<th>BR</th>
<th>MDDM</th>
<th>LIFT</th>
<th>LLSF</th>
<th>MLSF</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAL500</td>
<td>0.137±0.005</td>
<td>0.215±0.016</td>
<td>0.140±0.004</td>
<td>0.138±0.004</td>
<td>0.166±0.078</td>
<td>0.138±0.004</td>
</tr>
<tr>
<td>emotions</td>
<td>0.178±0.019</td>
<td>0.318±0.0635</td>
<td>0.394±0.028</td>
<td>0.179±0.015</td>
<td>0.184±0.005</td>
<td>0.407±0.021</td>
</tr>
<tr>
<td>medical</td>
<td>0.012±0.001</td>
<td>0.019±0.0001</td>
<td>0.017±0.002</td>
<td>0.022±0.008</td>
<td>0.025±0.002</td>
<td>0.013±0.002</td>
</tr>
<tr>
<td>enron</td>
<td>0.048±0.002</td>
<td>0.060±0.003</td>
<td>0.052±0.001</td>
<td>0.075±0.002</td>
<td>0.089±0.003</td>
<td>0.052±0.002</td>
</tr>
<tr>
<td>image</td>
<td>0.154±0.009</td>
<td>0.179±0.011</td>
<td>0.191±0.008</td>
<td>0.156±0.009</td>
<td>0.184±0.013</td>
<td>0.223±0.014</td>
</tr>
<tr>
<td>scene</td>
<td>0.076±0.009</td>
<td>0.110±0.007</td>
<td>0.437±0.019</td>
<td>0.080±0.009</td>
<td>0.144±0.012</td>
<td>0.126±0.008</td>
</tr>
<tr>
<td>yeast</td>
<td>0.191±0.004</td>
<td>0.200±0.008</td>
<td>0.199±0.005</td>
<td>0.197±0.005</td>
<td>0.311±0.009</td>
<td>0.209±0.006</td>
</tr>
<tr>
<td>slashdot</td>
<td>0.039±0.002</td>
<td>0.049±0.002</td>
<td>0.038±0.002</td>
<td>0.037±0.008</td>
<td>0.042±0.002</td>
<td>0.044±0.001</td>
</tr>
<tr>
<td>rcv1-s1</td>
<td>0.026±0.001</td>
<td>0.034±0.001</td>
<td>0.027±0.001</td>
<td>0.028±0.001</td>
<td>0.030±0.001</td>
<td>0.028±0.001</td>
</tr>
<tr>
<td>rcv1-s2</td>
<td>0.023±0.001</td>
<td>0.031±0.001</td>
<td>0.024±0.001</td>
<td>0.023±0.001</td>
<td>0.025±0.001</td>
<td>0.024±0.001</td>
</tr>
<tr>
<td>rcv1-s3</td>
<td>0.023±0.001</td>
<td>0.031±0.001</td>
<td>0.024±0.001</td>
<td>0.023±0.001</td>
<td>0.025±0.001</td>
<td>0.024±0.001</td>
</tr>
<tr>
<td>rcv1-s4</td>
<td>0.019±0.001</td>
<td>0.023±0.001</td>
<td>0.020±0.001</td>
<td>0.020±0.001</td>
<td>0.022±0.001</td>
<td>0.019±0.001</td>
</tr>
<tr>
<td>bibtex</td>
<td>0.078±0.006</td>
<td>0.084±0.001</td>
<td>0.104±0.006</td>
<td>0.082±0.007</td>
<td>0.082±0.007</td>
<td>0.350±0.023</td>
</tr>
<tr>
<td>core16k-s1</td>
<td>0.164±0.004</td>
<td>0.220±0.001</td>
<td>0.185±0.007</td>
<td>0.162±0.003</td>
<td>0.170±0.006</td>
<td>0.177±0.003</td>
</tr>
<tr>
<td>core16k-s2</td>
<td>0.172±0.002</td>
<td>0.224±0.001</td>
<td>0.184±0.011</td>
<td>0.172±0.003</td>
<td>0.166±0.007</td>
<td>0.169±0.005</td>
</tr>
</tbody>
</table>

Table 4: The performance of each comparing approach (mean ± std. deviation) on each data set in terms of *ranking loss*. On each data set, the best performance among the comparing algorithms is shown in bold face.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>REEL</th>
<th>BR</th>
<th>MDDM</th>
<th>LIFT</th>
<th>LLSF</th>
<th>MLSF</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAL500</td>
<td>0.137±0.005</td>
<td>0.215±0.016</td>
<td>0.140±0.004</td>
<td>0.138±0.004</td>
<td>0.166±0.078</td>
<td>0.138±0.004</td>
</tr>
<tr>
<td>emotions</td>
<td>0.138±0.021</td>
<td>0.113±0.047</td>
<td>0.479±0.044</td>
<td>0.144±0.023</td>
<td>0.258±0.031</td>
<td>0.483±0.042</td>
</tr>
<tr>
<td>medical</td>
<td>0.036±0.014</td>
<td>0.041±0.001</td>
<td>0.027±0.010</td>
<td>0.053±0.016</td>
<td>0.046±0.028</td>
<td>0.050±0.035</td>
</tr>
<tr>
<td>enron</td>
<td>0.084±0.005</td>
<td>0.016±0.001</td>
<td>0.097±0.006</td>
<td>0.219±0.004</td>
<td>0.126±0.005</td>
<td>0.089±0.003</td>
</tr>
<tr>
<td>image</td>
<td>0.154±0.009</td>
<td>0.179±0.011</td>
<td>0.191±0.008</td>
<td>0.156±0.009</td>
<td>0.184±0.013</td>
<td>0.223±0.014</td>
</tr>
<tr>
<td>scene</td>
<td>0.062±0.007</td>
<td>0.030±0.029</td>
<td>0.437±0.019</td>
<td>0.061±0.008</td>
<td>0.097±0.016</td>
<td>0.125±0.011</td>
</tr>
<tr>
<td>yeast</td>
<td>0.164±0.008</td>
<td>0.058±0.006</td>
<td>0.175±0.007</td>
<td>0.165±0.006</td>
<td>0.357±0.014</td>
<td>0.209±0.009</td>
</tr>
<tr>
<td>slashdot</td>
<td>0.403±0.035</td>
<td>0.508±0.024</td>
<td>0.357±0.023</td>
<td>0.363±0.015</td>
<td>0.380±0.022</td>
<td>0.476±0.022</td>
</tr>
<tr>
<td>rcv1-s1</td>
<td>0.050±0.003</td>
<td>0.071±0.001</td>
<td>0.089±0.005</td>
<td>0.053±0.003</td>
<td>0.060±0.03</td>
<td>0.074±0.004</td>
</tr>
<tr>
<td>rcv1-s2</td>
<td>0.053±0.003</td>
<td>0.060±0.001</td>
<td>0.089±0.004</td>
<td>0.058±0.002</td>
<td>0.062±0.005</td>
<td>0.080±0.004</td>
</tr>
<tr>
<td>rcv1-s3</td>
<td>0.053±0.002</td>
<td>0.065±0.001</td>
<td>0.097±0.004</td>
<td>0.058±0.003</td>
<td>0.059±0.003</td>
<td>0.083±0.006</td>
</tr>
<tr>
<td>rcv1-s4</td>
<td>0.037±0.002</td>
<td>0.057±0.001</td>
<td>0.067±0.005</td>
<td>0.040±0.003</td>
<td>0.050±0.005</td>
<td>0.057±0.006</td>
</tr>
<tr>
<td>bibtex</td>
<td>0.078±0.006</td>
<td>0.084±0.001</td>
<td>0.104±0.006</td>
<td>0.082±0.007</td>
<td>0.082±0.007</td>
<td>0.350±0.023</td>
</tr>
<tr>
<td>core16k-s1</td>
<td>0.164±0.004</td>
<td>0.220±0.001</td>
<td>0.185±0.007</td>
<td>0.162±0.003</td>
<td>0.170±0.006</td>
<td>0.177±0.003</td>
</tr>
<tr>
<td>core16k-s2</td>
<td>0.172±0.002</td>
<td>0.224±0.001</td>
<td>0.184±0.011</td>
<td>0.172±0.003</td>
<td>0.166±0.007</td>
<td>0.69±0.005</td>
</tr>
</tbody>
</table>

Based on the reported experimental results, the following major observations can be made:

- The performance of each comparing approach is significantly different against REEL.
- On each data set, the best performance among the comparing algorithms is shown in bold face.
- The measured difference (CD) with REEL is interconnected to each other with a thick line. Otherwise, it is regarded to have significantly different performance against REEL.
Table 5: The performance of each comparing approach (mean ± std. deviation) on each data set in terms of one-error. On each data set, the best performance among the comparing algorithms is shown in bold face.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>REEL</th>
<th>BR</th>
<th>MDDM</th>
<th>LIFT</th>
<th>LLSF</th>
<th>MLSF</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAL500</td>
<td>0.119±0.043</td>
<td>0.515±0.207</td>
<td>0.209±0.020</td>
<td>0.129±0.057</td>
<td>0.127±0.009</td>
<td>0.124±0.044</td>
</tr>
<tr>
<td>emotions</td>
<td>0.237±0.051</td>
<td>0.544±0.100</td>
<td>0.659±0.068</td>
<td>0.238±0.073</td>
<td>0.424±0.063</td>
<td>0.676±0.041</td>
</tr>
<tr>
<td>medical</td>
<td>0.163±0.024</td>
<td>0.181±0.035</td>
<td>0.230±0.041</td>
<td>0.347±0.059</td>
<td>0.213±0.028</td>
<td>0.184±0.031</td>
</tr>
<tr>
<td>enron</td>
<td>0.249±0.025</td>
<td>0.418±0.034</td>
<td>0.290±0.054</td>
<td>0.235±0.029</td>
<td>0.255±0.016</td>
<td>0.299±0.028</td>
</tr>
<tr>
<td>image</td>
<td>0.262±0.025</td>
<td>0.410±0.032</td>
<td>0.345±0.022</td>
<td>0.258±0.029</td>
<td>0.322±0.023</td>
<td>0.458±0.032</td>
</tr>
<tr>
<td>scene</td>
<td>0.190±0.026</td>
<td>0.337±0.033</td>
<td>0.437±0.019</td>
<td>0.191±0.026</td>
<td>0.263±0.031</td>
<td>0.343±0.030</td>
</tr>
<tr>
<td>yeast</td>
<td>0.216±0.023</td>
<td>0.336±0.038</td>
<td>0.232±0.030</td>
<td>0.224±0.016</td>
<td>0.373±0.026</td>
<td>0.253±0.029</td>
</tr>
<tr>
<td>slashdot</td>
<td>0.403±0.035</td>
<td>0.508±0.024</td>
<td>0.357±0.023</td>
<td>0.363±0.015</td>
<td>0.380±0.022</td>
<td>0.476±0.022</td>
</tr>
<tr>
<td>rcv1-s1</td>
<td>0.421±0.019</td>
<td>0.640±0.018</td>
<td>0.474±0.035</td>
<td>0.420±0.013</td>
<td>0.421±0.019</td>
<td>0.500±0.012</td>
</tr>
<tr>
<td>rcv1-s2</td>
<td>0.429±0.017</td>
<td>0.590±0.014</td>
<td>0.483±0.027</td>
<td>0.433±0.019</td>
<td>0.488±0.012</td>
<td>0.509±0.011</td>
</tr>
<tr>
<td>rcv1-s3</td>
<td>0.442±0.002</td>
<td>0.592±0.013</td>
<td>0.487±0.026</td>
<td>0.450±0.003</td>
<td>0.413±0.011</td>
<td>0.450±0.003</td>
</tr>
<tr>
<td>rcv1-s4</td>
<td>0.387±0.013</td>
<td>0.458±0.017</td>
<td>0.443±0.053</td>
<td>0.385±0.010</td>
<td>0.341±0.017</td>
<td>0.362±0.008</td>
</tr>
<tr>
<td>bitext</td>
<td>0.391±0.015</td>
<td>0.547±0.022</td>
<td>0.460±0.081</td>
<td>0.413±0.010</td>
<td>0.347±0.021</td>
<td>0.350±0.023</td>
</tr>
<tr>
<td>core16k-s1</td>
<td>0.699±0.019</td>
<td>0.878±0.009</td>
<td>0.728±0.024</td>
<td>0.680±0.023</td>
<td>0.638±0.009</td>
<td>0.781±0.024</td>
</tr>
<tr>
<td>core16k-s2</td>
<td>0.655±0.007</td>
<td>0.880±0.007</td>
<td>0.736±0.018</td>
<td>0.667±0.005</td>
<td>0.638±0.012</td>
<td>0.766±0.020</td>
</tr>
</tbody>
</table>

Table 6: The performance of each comparing approach (mean ± std. deviation) on each data set in terms of coverage. On each data set, the best performance among the comparing algorithms is shown in bold face.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>REEL</th>
<th>BR</th>
<th>MDDM</th>
<th>LIFT</th>
<th>LLSF</th>
<th>MLSF</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAL500</td>
<td>0.758±0.016</td>
<td>0.976±0.007</td>
<td>0.754±0.012</td>
<td>0.761±0.015</td>
<td>0.742±0.028</td>
<td>0.795±0.021</td>
</tr>
<tr>
<td>emotions</td>
<td>0.279±0.026</td>
<td>0.485±0.059</td>
<td>0.523±0.033</td>
<td>0.283±0.021</td>
<td>0.376±0.037</td>
<td>0.529±0.040</td>
</tr>
<tr>
<td>medical</td>
<td>0.052±0.015</td>
<td>0.120±0.025</td>
<td>0.055±0.015</td>
<td>0.072±0.019</td>
<td>0.088±0.018</td>
<td>0.069±0.037</td>
</tr>
<tr>
<td>enron</td>
<td>0.237±0.009</td>
<td>0.585±0.020</td>
<td>0.261±0.011</td>
<td>0.219±0.009</td>
<td>0.265±0.008</td>
<td>0.239±0.007</td>
</tr>
<tr>
<td>image</td>
<td>0.167±0.011</td>
<td>0.270±0.015</td>
<td>0.199±0.010</td>
<td>0.168±0.013</td>
<td>0.195±0.013</td>
<td>0.242±0.017</td>
</tr>
<tr>
<td>scene</td>
<td>0.066±0.007</td>
<td>0.163±0.016</td>
<td>0.437±0.019</td>
<td>0.065±0.008</td>
<td>0.096±0.015</td>
<td>0.119±0.010</td>
</tr>
<tr>
<td>yeast</td>
<td>0.454±0.008</td>
<td>0.637±0.017</td>
<td>0.462±0.007</td>
<td>0.454±0.007</td>
<td>0.630±0.011</td>
<td>0.516±0.008</td>
</tr>
<tr>
<td>slashdot</td>
<td>0.103±0.009</td>
<td>0.224±0.012</td>
<td>0.087±0.006</td>
<td>0.093±0.007</td>
<td>0.134±0.006</td>
<td>0.123±0.005</td>
</tr>
<tr>
<td>rcv1-s1</td>
<td>0.124±0.008</td>
<td>0.449±0.013</td>
<td>0.200±0.008</td>
<td>0.126±0.008</td>
<td>0.139±0.007</td>
<td>0.169±0.008</td>
</tr>
<tr>
<td>rcv1-s2</td>
<td>0.127±0.007</td>
<td>0.378±0.009</td>
<td>0.192±0.008</td>
<td>0.136±0.005</td>
<td>0.139±0.010</td>
<td>0.173±0.009</td>
</tr>
<tr>
<td>rcv1-s3</td>
<td>0.124±0.005</td>
<td>0.380±0.010</td>
<td>0.209±0.008</td>
<td>0.119±0.008</td>
<td>0.133±0.005</td>
<td>0.183±0.039</td>
</tr>
<tr>
<td>rcv1-s4</td>
<td>0.090±0.003</td>
<td>0.330±0.013</td>
<td>0.145±0.008</td>
<td>0.093±0.004</td>
<td>0.112±0.009</td>
<td>0.130±0.019</td>
</tr>
<tr>
<td>bitext</td>
<td>0.145±0.011</td>
<td>0.439±0.008</td>
<td>0.184±0.009</td>
<td>0.142±0.013</td>
<td>0.151±0.007</td>
<td>0.350±0.023</td>
</tr>
<tr>
<td>core16k-s1</td>
<td>0.323±0.007</td>
<td>0.670±0.008</td>
<td>0.364±0.015</td>
<td>0.328±0.004</td>
<td>0.312±0.009</td>
<td>0.344±0.005</td>
</tr>
<tr>
<td>core16k-s2</td>
<td>0.320±0.015</td>
<td>0.668±0.006</td>
<td>0.364±0.021</td>
<td>0.312±0.013</td>
<td>0.308±0.011</td>
<td>0.334±0.008</td>
</tr>
</tbody>
</table>

- It is impressive that REEL achieves lowest average rank in terms of all evaluation metrics. No comparing approach has significantly outperformed REEL based on the Friedman statistical test.
Table 7: The performance of each comparing approach (mean ± std. deviation) on each data set in terms of average precision. On each data set, the best performance among the comparing algorithms is shown in bold.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>REEL</th>
<th>BR</th>
<th>MDDM</th>
<th>LIFT</th>
<th>LLSF</th>
<th>MLSF</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAL500</td>
<td>0.503±0.012</td>
<td>0.226±0.038</td>
<td>0.491±0.018</td>
<td>0.499±0.012</td>
<td>0.497±0.078</td>
<td>0.481±0.009</td>
</tr>
<tr>
<td>emotions</td>
<td>0.826±0.051</td>
<td>0.629±0.048</td>
<td>0.538±0.038</td>
<td>0.818±0.034</td>
<td>0.703±0.040</td>
<td>0.532±0.024</td>
</tr>
<tr>
<td>medical</td>
<td>0.872±0.015</td>
<td>0.809±0.035</td>
<td>0.837±0.031</td>
<td>0.742±0.038</td>
<td>0.789±0.022</td>
<td>0.849±0.027</td>
</tr>
<tr>
<td>enron</td>
<td>0.672±0.019</td>
<td>0.457±0.020</td>
<td>0.626±0.023</td>
<td>0.697±0.010</td>
<td>0.665±0.016</td>
<td>0.642±0.013</td>
</tr>
<tr>
<td>image</td>
<td>0.828±0.013</td>
<td>0.722±0.012</td>
<td>0.778±0.012</td>
<td>0.826±0.016</td>
<td>0.790±0.015</td>
<td>0.709±0.019</td>
</tr>
<tr>
<td>scene</td>
<td>0.887±0.014</td>
<td>0.768±0.018</td>
<td>0.437±0.019</td>
<td>0.883±0.014</td>
<td>0.839±0.020</td>
<td>0.791±0.018</td>
</tr>
<tr>
<td>yeast</td>
<td>0.770±0.015</td>
<td>0.675±0.016</td>
<td>0.754±0.015</td>
<td>0.766±0.015</td>
<td>0.602±0.015</td>
<td>0.719±0.014</td>
</tr>
<tr>
<td>slashdot</td>
<td>0.695±0.024</td>
<td>0.592±0.021</td>
<td>0.735±0.015</td>
<td>0.711±0.013</td>
<td>0.707±0.012</td>
<td>0.637±0.016</td>
</tr>
<tr>
<td>rcv1-s1</td>
<td>0.591±0.009</td>
<td>0.382±0.009</td>
<td>0.496±0.015</td>
<td>0.585±0.010</td>
<td>0.607±0.008</td>
<td>0.511±0.008</td>
</tr>
<tr>
<td>rcv1-s2</td>
<td>0.606±0.008</td>
<td>0.433±0.012</td>
<td>0.516±0.018</td>
<td>0.600±0.009</td>
<td>0.627±0.012</td>
<td>0.520±0.036</td>
</tr>
<tr>
<td>rcv1-s3</td>
<td>0.601±0.010</td>
<td>0.437±0.015</td>
<td>0.506±0.016</td>
<td>0.596±0.012</td>
<td>0.625±0.008</td>
<td>0.561±0.015</td>
</tr>
<tr>
<td>rcv1-s4</td>
<td>0.667±0.009</td>
<td>0.489±0.017</td>
<td>0.572±0.029</td>
<td>0.658±0.008</td>
<td>0.698±0.012</td>
<td>0.665±0.011</td>
</tr>
<tr>
<td>bibtex</td>
<td>0.550±0.013</td>
<td>0.386±0.029</td>
<td>0.483±0.031</td>
<td>0.545±0.011</td>
<td>0.603±0.013</td>
<td>0.350±0.023</td>
</tr>
<tr>
<td>corel16k-s1</td>
<td>0.301±0.009</td>
<td>0.104±0.003</td>
<td>0.274±0.007</td>
<td>0.305±0.010</td>
<td>0.156±0.029</td>
<td>0.255±0.010</td>
</tr>
<tr>
<td>corel16k-s2</td>
<td>0.312±0.005</td>
<td>0.096±0.003</td>
<td>0.264±0.009</td>
<td>0.318±0.008</td>
<td>0.341±0.008</td>
<td>0.255±0.006</td>
</tr>
</tbody>
</table>

Table 8: The performance of each comparing approach (mean ± std. deviation) on each data set in terms of macro-averaging AUC. On each data set, the best performance among the comparing algorithms is shown in bold.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>REEL</th>
<th>BR</th>
<th>MDDM</th>
<th>LIFT</th>
<th>LLSF</th>
<th>MLSF</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAL500</td>
<td>0.554±0.012</td>
<td>0.501±0.001</td>
<td>0.517±0.018</td>
<td>0.546±0.010</td>
<td>0.566±0.016</td>
<td>0.519±0.012</td>
</tr>
<tr>
<td>emotions</td>
<td>0.858±0.023</td>
<td>0.594±0.033</td>
<td>0.488±0.048</td>
<td>0.858±0.016</td>
<td>0.553±0.032</td>
<td>0.486±0.038</td>
</tr>
<tr>
<td>medical</td>
<td>0.894±0.052</td>
<td>0.808±0.034</td>
<td>0.875±0.025</td>
<td>0.863±0.042</td>
<td>0.825±0.033</td>
<td>0.885±0.039</td>
</tr>
<tr>
<td>enron</td>
<td>0.683±0.020</td>
<td>0.595±0.015</td>
<td>0.684±0.036</td>
<td>0.731±0.022</td>
<td>0.675±0.023</td>
<td>0.674±0.020</td>
</tr>
<tr>
<td>image</td>
<td>0.860±0.012</td>
<td>0.717±0.012</td>
<td>0.815±0.015</td>
<td>0.860±0.014</td>
<td>0.788±0.023</td>
<td>0.778±0.025</td>
</tr>
<tr>
<td>scene</td>
<td>0.948±0.006</td>
<td>0.801±0.017</td>
<td>0.437±0.019</td>
<td>0.945±0.007</td>
<td>0.922±0.011</td>
<td>0.888±0.009</td>
</tr>
<tr>
<td>yeast</td>
<td>0.686±0.015</td>
<td>0.572±0.008</td>
<td>0.614±0.015</td>
<td>0.693±0.019</td>
<td>0.566±0.016</td>
<td>0.616±0.013</td>
</tr>
<tr>
<td>slashdot</td>
<td>0.853±0.012</td>
<td>0.682±0.013</td>
<td>0.881±0.015</td>
<td>0.861±0.017</td>
<td>0.850±0.011</td>
<td>0.819±0.015</td>
</tr>
<tr>
<td>rcv1-s1</td>
<td>0.912±0.004</td>
<td>0.900±0.001</td>
<td>0.899±0.006</td>
<td>0.911±0.009</td>
<td>0.903±0.004</td>
<td>0.672±0.009</td>
</tr>
<tr>
<td>rcv1-s2</td>
<td>0.900±0.001</td>
<td>0.617±0.011</td>
<td>0.825±0.016</td>
<td>0.892±0.007</td>
<td>0.857±0.006</td>
<td>0.835±0.008</td>
</tr>
<tr>
<td>rcv1-s3</td>
<td>0.899±0.006</td>
<td>0.616±0.009</td>
<td>0.779±0.025</td>
<td>0.886±0.008</td>
<td>0.852±0.008</td>
<td>0.842±0.009</td>
</tr>
<tr>
<td>rcv1-s4</td>
<td>0.911±0.009</td>
<td>0.621±0.007</td>
<td>0.810±0.021</td>
<td>0.905±0.049</td>
<td>0.833±0.012</td>
<td>0.746±0.028</td>
</tr>
<tr>
<td>bibtex</td>
<td>0.903±0.004</td>
<td>0.648±0.007</td>
<td>0.887±0.005</td>
<td>0.908±0.003</td>
<td>0.898±0.009</td>
<td>0.350±0.023</td>
</tr>
<tr>
<td>corel16k-s1</td>
<td>0.672±0.009</td>
<td>0.520±0.003</td>
<td>0.622±0.011</td>
<td>0.678±0.010</td>
<td>0.678±0.009</td>
<td>0.661±0.008</td>
</tr>
<tr>
<td>corel16k-s2</td>
<td>0.719±0.009</td>
<td>0.522±0.004</td>
<td>0.628±0.011</td>
<td>0.720±0.008</td>
<td>0.709±0.007</td>
<td>0.676±0.008</td>
</tr>
</tbody>
</table>

- Compared to BR which corresponds to the degenerated version of label-specific features strategy without considering tailored features for each class label, REEL achieves superior performance in terms of all evaluation metrics except ranking loss.
Figure 1: Comparison of REEL (control approach) against other approaches with the Bonferroni-Dunn test. Approaches not connected with REEL in the CD diagram are considered to have significantly different performance from the control approach (CD=1.7597 at 0.05 significance level).

- Compared to MDDM which performs feature manipulation via dimensionality reduction, REEL achieves superior performance in terms of all evaluation metrics.

- Compared to other learning approaches LIFT, LLSF and MLSF which also employs the strategy of label-specific features for model induction, REEL achieves superior performance in 63.3% cases out of 30 statistical comparisons (5 comparing approaches × 6 evaluation metrics).

As shown in Table 1, two parameters $\lambda$ and $\mathcal{L}$ need to be specified to instantiate REEL. In this paper, SVM is utilized to serve as the binary learning algorithm $\mathcal{L}$. For the other regularization parameter $\lambda$, Figure 2 shows how the performance of REEL changes as the value of $\lambda$ increases in terms of several evaluation metrics. Generally, the performance of REEL is relatively stable with $\lambda$ taking values within $[0.1, 1]$. In this paper, $\lambda$ is set to be 1 as shown in Subsection 4.1.

5. Conclusion

In this paper, the problem of generating label-specific features for multi-label learning is studied. By employing a two-stage generation procedure, the base label-specific features for each class label are enriched by concatenating informative label-specific features from other class labels via sparse regularization. The computational complexity of resulting multi-label classification model is linear to the number of class labels in label space. Comprehensive comparative studies against state-
Figure 2: The performance of REEL changes as the value of parameter $\lambda$ increases.

of-the-art approaches validate the effectiveness of the proposed label-specific features generation techniques for multi-label learning.

Acknowledgements

The authors wish to thank the anonymous reviewers for their helpful comments and suggestions. This work was supported by the National Key R&D Program of China (2018YFB1004300), the National Science Foundation of China (61573104), and partially supported by the Collaborative Innovation Center of Novel Software Technology and Industrialization.

References


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


